The Impact of Pseudorandom Number Quality on *P-RnaPredict*, a Parallel Genetic Algorithm for RNA Secondary Structure Prediction

Kay C. Wiese
School of Computing Science
Simon Fraser University
Surrey, B.C., Canada
wiese@cs.sfu.ca

Andrew Hendriks, Alain Deschênes, and Belgacem Ben Youssef
InfoNet Media Center
Simon Fraser University
Surrey, B.C., Canada
ahendrik@sfu.ca, aadesche@sfu.ca, bbenyous@sfu.ca

ABSTRACT

This paper presents a parallel version of *RnaPredict*, a genetic algorithm (GA) for RNA secondary structure prediction. The research presented here builds on previous work and examines the impact of three different pseudorandom number generators (PRNGs) on the GA's performance. The three generators tested are the C standard library PRNG RAND, a parallelized multiplicative congruential generator (MCG), and a parallelized Mersenne Twister (MT). A fully parallel version of *RnaPredict* using the Message Passing Interface (MPI) was implemented. The PRNG comparison tests were performed with known structures that are 118, 122, 543, and 556 nucleotides in length. The effects of the PRNGs are investigated and the predicted structures are compared to known structures.

Categories and Subject Descriptors

J.3 [**Life and Medical Sciences**]: Biology and genetics; D.1.3 [**Programming Techniques**]: Parallel programming; G.3 [**Probability and Statistics**]: Random number generation

General Terms

Algorithms, Performance

Keywords

Bioinformatics, RNA Secondary Structure Prediction, Parallel Evolutionary Algorithms, Random Number Generators

1. INTRODUCTION

The basic function of a biomolecule is determined by its 3-dimensional shape, otherwise known as the tertiary structure. However, existing empirical methods to determine this shape are too costly and lengthy to be practical. RNA is of interest as a biomolecule because it is central in several stages of protein synthesis. Also, its secondary structure dominates its tertiary structure. In our model, RNA secondary structure develops as a consequence of bonds which form between specific pairs of nucleotides known as the canonical base pairs. Searching a sequence of nucleotides for all possible base pairs is rapid and straightforward; the challenge comes from attempting to predict which specific canonical base pairs will form bonds in the real structure. Various algorithms have been used for RNA structure prediction such as dynamic programming, comparative methods, and stochastic methods such as genetic algorithms (GAs).

The fitness metric employed to guide a given GA through the search space of possible base pairs is energy minimization. As an RNA molecule will fold into a structure with near minimal free energy (ΔG), the GA attempts to find the structure resulting in the lowest energy possible.

Coarse-grained distributed GAs [1] offer a number of advantages beyond the benefits of parallelization. These include the prevention of premature convergence by maintaining diversity, an increase of the selection pressure within the entire population, and also a reduction of the time to convergence.

This work builds on [5] and presents *P-RnaPredict*, a fully parallelized distributed GA to predict RNA secondary structure; it is based on the MPI standard to run on a 128 node Beowulf cluster. In our parallel GA, random numbers are used to make coarse-grained decisions in population initialization, selection, crossover, mutation, and migration. During development of *P-RnaPredict*, three major PRNG issues arose. The first was the dramatic increase in random number consumption as the RNA sequences increased in length. The second was that the standard development library PRNG functions are not designed for parallel usage. Third, we average our results over 30 randomly seeded runs. This implicitly assumes that the random numbers generated for each run are independent of each other. This compelled our investigation into the impact of PRNGs on GAs in general, and *P-RnaPredict* in particular.

Although there appears to be very little in the literature regarding parallel GAs and PRNGs, a series of empirical studies [6] were done on how serial GA performance is impacted by PRNGs. These studies indicated that PRNG quality had no statistically significant effect on GA performance. However, GA performance could vary depending on

GECCO'05, June 25–29, 2005, Washington, DC, USA.
ACM 1-59593-010-8/05/0006.

the PRNG and test function chosen, and in isolated cases poor PRNGs could result in slightly better GA performance.

An "ablation" study in 2002 by Cantù-Paz [2] found that the PRNG used to initialize the random population is critical, whilst the other GA components were relatively unaffected. This is especially significant in a multi-population GA like ours, as any overlap in the PRNG period during population initialization could result in duplicate individuals. This in turn could result in diminished GA performance.

Based on these observations, two parallel PRNGs were selected for evaluation in the parallel GA implementation. The first was the parallel MT, named "Dynamic Creation" (DC). The second was a parallelized version of a Multiplicative Congruential Generator (MCG). The MCG's parameters were m=2^{31}-1, c=0, and a=6208991 as suggested by [3]. This MCG was parallelized by the leap-frog method [4], and was deliberately chosen to have a lower quality and shorter period than the DC. We also employed the original serial GA as a control, which used the standard C library PRNG RAND.

Selection of test parameters were based on previously published experimental results [7], and were as follows: The crossover probability (P_c) was set to 0.7, while the mutation probability (P_m) varied as either 0.25 or 0.8. The selection technique employed was standard roulette wheel selection (STDS), with 1-Elitism. The chosen thermodynamic model was INN-HB and crossover was CX. The global population of 700 was split into two separate sets of deme sizes and deme counts: (50, 14), and (70, 10) respectively. The migration interval was fixed at 20 generations, and the migration rate was fixed at 10 percent. Finally, the topology was fully connected, and the migration policy was set to "best replace worst." Each parameter set was repeated with 30 random seeds and the results averaged.

Four RNA sequences were taken as test data from the Comparative RNA Web Site; they were chosen to provide a good variety of sequence lengths and a variety of organisms. Each sequence chosen had a known structure available for comparison, determined by comparative methods. The four sequences used were a 556 nucleotide (nt) *Acanthamoeba griffini* sequence, a 543 nt *Hildenbrandia rubra* sequence, a 118 nt *Saccharomyces cerevisiae* sequence, and a 122 nt *Haloarcula marismortui* sequence. Only the *A.griffini* results are shown here.

2. RESULTS

Table 1 presents the *Acanthamoeba griffini* results; it indicates that the MCG PRNG performed best in two of the parameter sets based on average ΔG, with the DC and RAND PRNGs performing best in one parameter set each. Overall, the MCG PRNG reached the best average ΔG at -190.79 kcal/mol with the following parameters: a Deme Size of 70, a Deme Count of 10, and a P_m of 0.8. Averaged over 30 runs, the DC PRNG found the highest percentage of base pairs matching the known structure at 32.34%. The best overall structure was found with 64.88% matching base pairs with the following parameters: a MCG PRNG, a Deme Size of 70, a Deme Count of 10, and a P_m of 0.8.

Table 1: *P-RnaPredict* results using three different PRNGs on the *A.griffini* sequence

Deme Size	P_m	Deme Count	PRNG	Avg. ΔG	Avg. Base Pair %	Best Base Pair %
70	0.25	10	DC	-187.58	28.39	58.77
70	0.25	10	MCG	-187.29	30.35	56.48
70	0.25	10	RAND	-186.35	27.04	46.56
70	0.8	10	MCG	**-190.79**	29.79	**64.88**
70	0.8	10	DC	-189.35	29.26	60.30
70	0.8	10	RAND	-187.8	28.39	60.30
50	0.25	14	RAND	-186.51	26.89	52.67
50	0.25	14	DC	-184.74	**32.34**	58.01
50	0.25	14	MCG	-184.43	28.04	48.09
50	0.8	14	MCG	-188.85	31.67	54.96
50	0.8	14	DC	-188.29	26.92	48.85
50	0.8	14	RAND	-185.27	27.17	47.32

3. CONCLUSIONS

The results from the four sequences indicate that PRNG quality does not have a significant effect on GA performance, which is in keeping with the previous research on serial GAs and PRNGs. However, the serial version of RAND consistently underperformed and it cannot easily be parallelized. For a truly parallel implementation such as *P-RnaPredict*, other PRNGs such as MCG and DC need to be used. Overall, prediction accuracy is very good, particularly so for shorter sequences. Further improvements are expected from modelling non-canonical base pairs.

4. REFERENCES

[1] E. Cantù-Paz. *Efficient and Accurate Parallel Genetic Algorithms*. Kluwer Academic Publishers, 2000.

[2] E. Cantù-Paz. On random numbers and the performance of genetic algorithms. In W. B. Langdon, E. Cantú-Paz, K. Mathias, R. Roy, D. Davis, R. Poli, K. Balakrishnan, V. Honavar, G. Rudolph, J. Wegener, L. Bull, M. A. Potter, A. C. Schultz, J. F. Miller, E. Burke, and N. Jonoska, editors, *GECCO 2002: Proceedings of the Genetic and Evolutionary Computation Conference*, pages 311–318, Morgan Kaufmann Publishers, 2002.

[3] G. A. Fishman and I. Louis R Moore. An exhaustive analysis of multiplicative congruential random number generators with modulus 2^{31}-1. *SIAM J. Sci. Stat. Comput.*, 7(1):24–45, 1986.

[4] G. Fox, M. Johnson, G. Lyzenga, S. Otto, J. Salmon, and D. Walker. *Solving Problems On Concurrent Processors, vol. 1 - General Techniques And Regular Problems*. Prentice-Hall International, 1988.

[5] A. Hendriks, A. Deschênes, and K. C. Wiese. A parallel evolutionary algorithm for RNA secondary structure prediction using stacking-energies (INN and INN-HB). In *Proceedings of the 2004 IEEE Symposium on Computational Intelligence in Bioinformatics and Computational Biology (CIBCB'04)*, pages 223–230, IEEE Press, 2004.

[6] M. M. Meysenburg and J. A. Foster. Randomness and GA performance, revisited. In W. Banzhaf, J. Daida, A. E. Eiben, M. H. Garzon, V. Honavar, M. Jakiela, and R. E. Smith, Editors, *Proceedings of the Seventh International Conference on Genetic Algorithms*, pages 425–432, San Francisco, CA, Morgan Kaufmann, 1999.

[7] K. C. Wiese, A. Deschênes, and E. Glen. Permutation based RNA secondary structure prediction via a genetic algorithm. In R. Sarker, R. Reynolds, H. Abbass, K. C. Tan, B. McKay, D. Essam, and T. Gedeon, editors, *Proceedings of the 2003 Congress on Evolutionary Computation (CEC2003)*, pages 335–342, Canberra, IEEE Press, 2003.