A Genetic Algorithm for the Shortest Common Superstring Problem

Luis C. González, Heidi J. Romero, and Carlos A. Brizuela

Computer Science Department, CICESE Research Center
Km 107 Carr. Tijuana-Ensenada, Ensenada, B.C., México
+52–646–1750500
{gurrola, hjromero, cbrizuel}@cicese.mx

Many real world problems can be modeled as the shortest common superstring problem. This problem has several important applications in areas such as DNA sequencing and data compression.

The shortest common superstring problem (SCS) can be formulated as follows. Given a set of strings $\mathbf{S} = \{s_1, s_2, ..., s_n\}$ the goal is to find the shortest string N^* such that each $s_i \in \mathbf{S}$ is a string of N^*. Finding the SCS of a set of strings is known to be NP-hard [2].

The shortest common superstring problem can be modeled as the asymmetric TSP (ATSP). The optimum tour cost for the ATSP problem represents a lower bound for the optimum of the SCS N^* of $\mathbf{S}$. Unfortunately, the ATSP is also an NP-hard problem. Fortunately, a lower bound for this problem is the well known Held-Karp bound [4]. We will use this bound to study the solution quality produced by our algorithm. We propose an algorithm which is based on a recently proposed algorithm [1] for the sequencing by hybridization (SBH) problem. Each individual is represented by a permutation of indices of substrings in $\mathbf{S}$. Specifically, the adjacency-based coding is used. The fitness of each individual is given by the maximum number k of overlapping characters the individual represents, taking into account all substrings in the chromosome. For each individual, its fitness value is normalized considering the maximum possible overlap. The individuals are selected according to the stochastic remainder method without replacement [3]. The population for the next generation is constructed based on the already selected individuals, which are randomly paired to undergo crossover with $p_c = 1$, always maintaining the best individual found (elitism).

In order to test our algorithm we have decided to deal with the SBH, motivated by the availability of many benchmarks for this problem coming from the real world. All sets $\mathbf{S}$ used in the experiment have been derived from DNA sequences coding human proteins (taken from GenBank, National Institute of Health, USA).

Table 1 presents the approximation ratio between the solutions generated by the GA (SCS GA) and their corresponding Held-Karp (HK) bounds. Column 1 presents the number of instances. Column 2 presents the size of the instances. Column 3 presents average approximation ratios to the HK bound, and column 4 presents their standard deviation. Column 5 presents approximation ratios of the best results to the HK bound and column 6 presents their standard deviation.

K. Deb et al. (Eds.): GECCO 2004, LNCS 3103, pp. 1305–1306, 2004.

It would also be interesting to see how similar the solutions are to the original DNA sequences, from where the sets of substrings were derived. For this purpose, we compared the solutions generated by the GA with the original sequences using a pairwise alignment algorithm. Column 7 presents how similar the solutions generated by the GA and the original sequences are; if both sequences are identical, then the similarity score is 100%.

Table 1. Comparison results of the SCS GA and the Held-Karp lower bound over 30 runs

No. of Instances	**|S|**	**SCS GA (%)**	**S.D.**	**Best (%)**	**S.D.**	**Similarity score (%)**
10	100	5.48	2.7	1.20	1.97	98.95%
10	200	3.83	1.34	0.81	1.08	96.48%
10	300	3.38	1.07	1.22	1.28	97.16%
10	400	2.58	0.84	0.82	0.59	96.46%
10	500	2.6	0.86	1.21	0.58	87.85%
10	600	2.33	0.86	1.06	0.63	83.39%
10	700	2.49	0.99	1.50	0.8	70.58%
10	800	2.35	0.81	1.44	0.64	70.72%
10	900	2.02	0.78	1.11	0.49	69.50%
10	1000	2.05	0.67	1.26	0.5	57.18%
10	2000	2.12	0.73	1.28	0.5	31.22%
10	3000	3.22	1.15	2.18	0.9	29.20%

The best solution lengths (except for the case of $|\mathbf{S}| = 3000$) exceeds the HK bound in no more than 1.5%. Furthermore, if we consider that this algorithm was designed for the SCS problem, the similarity percentage score for the first six sets of instances, where $100 \leq |\mathbf{S}| \leq 600$, can be considered as motivating, and this point is of special interest given that for real hybridization experiments $100 \leq |\mathbf{S}| \leq 500$.

Relative errors less than 6% from the optimum tell us about the suitability of this algorithm for real world applications.

References

1. C. Brizuela, L. González, and H. Romero. An Improved Genetic Algorithm for the Sequencing by Hybridization Problem. In *(to appear) Proceedings of the 2nd European Workshop on Evolutionary Bioinformatics, EvoBIO 2004*.
2. M. Garey and D. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W.H. Freeman, 1979.
3. D.E. Goldberg. *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison-Wesley, 1989.
4. M. Held and R. Karp. The Traveling Salesman Problem and Minimum Spanning Trees. *Operations Research*, 18:1138–1162, 1970.