STUDIA UNIV. BABEŞ–BOLYAI, INFORMATICA, Volume **LVI**, Number 2, 2011

COLLABORATIVE SEARCH OPERATORS FOR EVOLUTIONARY APPROACHES TO DENSITY CLASSIFICATION IN CELLULAR AUTOMATA

ANCA GOG, CAMELIA CHIRA

Abstract. The density classification problem is a prototypical distributed computational task for Cellular Automata widely studied for the analysis of complex systems. This paper focuses on evolutionary models designed to approach this problem, particularly on the importance of search operators in the context of evolutionary algorithms. Different collaborative recombination operators are described and engaged in an evolutionary search framework for the density classification task in cellular automata. The significance of considering genetic material from parents, global best/worst solutions and the individual's best ancestors in the recombination process is discussed.

1. Introduction

Cellular Automata (CA) are discrete dynamical systems having the ability to generate highly complex behaviour starting from a simple initial configuration and set of update rules [9, 15, 1]. Evolving CA rules for the computationally emergent task of density classification is a challenging problem extensively studied due to its simple description and potential to generate a variety of complex behaviours [8, 14]. The task refers to determining the initial density most present in the initial cellular state of a one-dimensional cellular automaton within a number of update steps. This is not a trivial task because finding the density of the initial configuration is a global task while the CA evolution relies only on local interactions between cells with limited information and communication.

This paper focuses on evolutionary approaches to the computationally emergent task of density classification. Genetic algorithms have already been

---

Received by the editors: April 10, 2011.

2010 *Mathematics Subject Classification.* 68-02.

1998 *CR Categories and Descriptors.* I.2.8 [**Artificial Intelligence**]: Problem Solving, Control Methods, and Search – *Heuristic methods*.

Key words and phrases. evolutionary algorithms, density classification task, cellular automata, collaborative search.

succesfully applied for this problem in many studies [3, 9, 10, 12, 8, 6]. An important search operator in genetic algorithms is the recombination of two individuals which should be able to produce new interesting rules. Recent studies [4, 5, 2] highlight the benefits of considering information received from other selected specific individuals (besides the genetic material of the parents) in the recombination process for permutation-based encoding problems. In this paper, we intend to study their performance in an evolutionary model for the density classification problem in CA which requires a binary representation.

2. Collaborative Search Operators for Evolutionary Models

When using an evolutionary framework for solving complex problems, the choice of search operators is of great importance as the success of the approach depends on it. For detecting rules in cellular automata it is also necessary to design suitable search operators able to increase both the exploration and the exploitation of the search space, thus leading to valuable results.

Several collaborative recombination schemes for permutation-based encoding have been proposed [4, 5, 2] with encouraging results. The collaborative feature refers to the fact that when crossover between two individuals is performed, genetic material from the parents best ancestors and/or from the best/worst solution obtained so far is also involved.

The best results have been obtained with *Best Order Crossover (BOX)* [2], which clearly outperforms the most popular recombination operators for permutation-based encoding. This crossover operator uses genetic material belonging to the *GlobalBest* individual together with genetic information from the two parents that are subject to recombination. Several cutting points are randomly chosen and each resulting sequence in the offspring inherits information from one of three sources.

It is well known that genetic operators are highly dependent on the chromosomes encoding and on the problem to solve. Nevertheless, due to the encouraging results obtained by collaborative recombination for permutation-based encoding, the binary codification has also been considered and approached in this paper. Several collaborative recombination schemes have been tested against the difficult problem of rule detection in cellular automata. The best performing operator is presented in what follows.

2.1. **Two-Point Line Crossover.**

For each individual in the population we keep track of its best ancestor (*LineBest*), representing the best individual that has contributed to its creation by mutation or recombination. When performing recombination, the *LineBest* of each parent is also considered besides genetic information from the two parents. Two cutting points are randomly chosen, thus resulting three sequences of genes. For the first offspring, each of the three sequences is taken from one of the two parents or from the *LineBest* of the second parent. The source of the sequence is randomly chosen but in such a way that each of the three chromosomes will contribute to the first offspring. The second offspring is obtained in a similar way, using the *LineBest* of the first parent instead of the *LineBest* of the second parent and different cutting points.

3. Evolutionary Model for the Density Classification Problem

Experiments focus on the most frequently studied version of the density classification problem: the one-dimensional binary-state CA of size $N = 149$ based on the radius of 3. This means that each cell is connected to 3 neighbors from both sides giving a neighborhood size of 7. The radius of the CA gives a rule size of $2^{2r+1} = 128$. The number of all possible rules is $2^{128} \simeq 10^{36}$ which makes an exhaustive evaluation of all this rules unfeasible.

A very simple evolutionary framework has been setup for solving the density classification task, in order to better analyze the performances of different recombination operators. A potential solution of the problem is a one-dimensional array of bits of size $2^{2r+1} = 128$ (because we have considered the radius as having the value r=3) and represents a rule table for the cellular automaton. The initial population is randomly generated.

The potential solutions are evaluated by means of a real-valued fitness function $f : X \to [0, 1]$, where X denotes the search space of the problem. As stated before, $|X| = 2^{128}$. The fitness function represents the fraction of correct classification over 100 randomly generated initial configurations. A relative fitness is used, as the set of initial configurations is generated anew for each generation of the algorithm. This way, solutions with high fitness in one generation and which survive in the next generation will be evaluated again using another set of 100 initial configurations.

Every set of 100 initial configurations was generated so that their densities are uniformly distributed over $[0, 1]$. It is important to underline the difference between the fitness of a rule and the performance of a rule. While the fitness is evaluated by using 100 uniformly distributed initial configurations, the performance of a rule is computed as the fraction of correct classifications for 10^4 randomly generated initial configurations. The initial configurations are generated in such a way that each cell has the same probability $\frac{1}{2}$ of being 0 or 1. This means that the density of 1s will be around $\frac{1}{2}$ for most of the initial configurations and these are actually the most difficult cases to correctly classify. The CA is iterated until it reaches a fixed-point configuration of 1s or 0s but for no more than $M \simeq 2N$ time steps.

The individual resulted after each recombination will be mutated at exactly two randomly chosen positions. A weak mutation is considered, the probability of obtaining a different value for the chosen position being equal to the probability of obtaining the very same value.

The algorithm is applied for 100 generations with a population size of 100, roulette selection, different crossover schemes with the same probability of 0.8, weak mutation with probability 0.2 and elite size of 10%.

4. COMPUTATIONAL EXPERIMENTS

The algorithm described in the previous section has been applied with the following recombination operators: *one-point crossover* (a single cutting point on both parents chromosomes is randomly chosen and resulting sequences are swapped in order to create offspring); *two-point crossover* (two cutting points on both parents chromosomes are randomly chosen and everything between the two points is swapped between the parents chromosomes); *uniform crossover* (each gene from the offspring is taken from either parent with the same probability). A recent study on the efficiency of crossover operators in genetic algorithms with binary representation [13] revealed the good performance of *two-point crossover* and *uniform crossover*. However, we also consider *one-point crossover* as, for the investigated problem of rule detection, *one-point crossover* seems to have slightly better results compared to the other two. Proposed collaborative *two-point line crossover* has been compared with these three popular operators. Tables 1 presents the average and the maximum performances obtained after 10 runs of the algorithm with different crossover operators.

TABLE 1. Performances obtained after 10 runs of the algorithm

	One-point crossover	**Two-point crossover**	**Uniform crossover**	**Two-point line crossover**
Average	**0.64**	**0.63**	**0.50**	**0.67**
Best	**0.65**	**0.65**	**0.50**	**0.73**

Obtained results indicate a high performance of the proposed collaborative recombination. The difference between the best rule performance obtained when using the *two-point line crossover* (0.73) and the maximum value obtained by other operators (0.65) is not at all neglectable. This is true especially because we are using a standard evolutionary algorithm and the best known performances is 0.88 [11] and has been obtained with a method that uses a two-tier evolutionary environment. We should also recall another good performance obtained by [7] (0.86) using a coevolutionary approach that evolves both populations of rules and of initial configurations, thus increasing the computational complexity.

The small values obtained when using *uniform crossover* indicate the fact that mixing individual genes from the two parents instead of mixing (relatively) long sequences of genes leads to difficulties in the search process. This conclusion is also confirmed by some experiments we have performed with other collaborative crossover operators. We have tried different schemes where genes are randomly taken from parents, *LineBest* of parents, *GlobalBest* or *GlobalWorst* and similar results have been obtained (both performance and fitness values were not higher than 0.53). This led us to considering sequences of genes instead of individual genes. Several one-point collaborative crossover schemes have also been considered, with a sequence taken from one parent and the other sequence take from the *LineBest* of the other parent, or from the *GlobalBest/GlobalWorst* but obtained performances did not exceed 0.64, values similar to the ones obtained by simple *one-point crossover* and *two-point crossover*. Even smaller values have been obtained when more than two cutting points have been considered (performance did not exceed 0.60). Regarding the two-point collaborative crossover, values around 0.64 have been also obtained when genetic material from *GlobalBest* has been considered instead of or together with genetic material from *LineBest*. This might be due to the fact that using the best individual in all recombinations (even if the best individual constantly changes) affects the search process by not introducing diversity within the population.

5. Conclusions

A collaborative recombination operator for binary encoding has been proposed. Numerical results indicate a competitive performance compared with some of the most popular crossover operators for the difficult problem of rule detection in cellular automata. The main feature of the proposed operator is the use of genetic information not only from the parents, but from their *LineBest* as well. Several other collaborative recombination schemes have been studied but weak results have been obtained. It has been observed that mixing particular genes instead of genes sequences leads to difficulties in the search process. More extensive studies on the collaborative feature will follow.

6. Acknowledgments

This research is supported by Grant PN II TE 320, Emergence, auto-organization and evolution: New computational models in the study of complex systems, funded by CNCSIS, Romania.

References

[1] C. Chira, A. Gog, R. Lung, D. Iclanzan, *Complex Systems and Cellular Automata Models in the Study of Complexity*, Studia Informatica series, Vol. LV, No. 4 (2010), pp. 33-49.

[2] C. Chira, A. Gog, *Comparative Analysis of Recombination Operators in Genetic Algorithms for the Travelling Salesman Problem*, HAIS 2011, to appear.

[3] J.P. Crutchfield, M. Mitchell, *The evolution of emergent computation*, Proceedings of the National Academy of Sciences, USA 92 (23), (1995), pp.10742-10746.

[4] Gog, A., Dumitrescu, D., *A New Recombination Operator for Permutation Based Encoding*, Proceedings of the 2nd International Conference on Intelligent Computer Communication and Processing (ICCP 2006), (2006) pp. 11-16.

[5] Gog, A., Dumitrescu, D., Hirsbrunner, B., *BestWorst Recombination Scheme for Combinatorial Optimization*, Proceedings of the International Conference on Genetic and Evolutionary Methods (GEM 2007), Las Vegas, USA, (2007) pp. 115-119.

[6] A. Gog, C. Chira, *Cellular Automata Rule Detection Using Circular Asynchronous Evolutionary Search*, HAIS 2009, LNCS 5572 (2009), pp. 261-268.

[7] H. Juille, J.B. Pollack, *Coevolutionary learning and the design of complex systems*, Advances in Complex Systems, Vol. 2, No. 4 (2000), pp.371-394.

[8] M. Marques-Pita, M. Mitchell, L. Rocha, *The role of conceptual structure in designing cellular automata to perform collective computation*, Proceedings of the Conference on Unconventional Computation, UC 2008, Springer (Lecture Notes in Computer Science), 2008.

[9] M. Mitchell, J. P. Crutchfield, R. Das, *Evolving Cellular Automata with Genetic Algorithms: A Review of Recent Work*, Proceedings of the First International Conference on Evolutionary Computation and Its Applications, Russian Academy of Sciences (1996).

[10] M. Mitchell, M. D. Thomure, N. L. Williams, *The role of space in the Success of Coevolutionary Learning*, Proceedings of ALIFE X - The Tenth International Conference on the Simulation and Synthesis of Living Systems (2006).

[11] G.M.B. Oliveira, L.G.A. Martinsa, L.B. de Carvalho, E. Fynn, *Some Investigations About Synchronization and Density Classification Tasks in One-dimensional and Two-dimensional Cellular Automata Rule Spaces*, Electronic Notes in Theoretical Computer Science (ENTCS), Vol. 252, pp. 121-142, 2009.

[12] L.Pagie, M. Mitchell, *A comparison of evolutionary and coevolutionary search*, Int. J. Comput. Intell. Appl., Vol. 2, No. 1, (2002), pp. 53-69.

[13] S. Picek, M. Golub, *On the efficiency of crossover operators in genetic algorithms with binary representation.*, World Scientific and Engineering Academy and Society (WSEAS), Stevens Point, Wisconsin, USA (2010), 167-172.

[14] M.Tomassini, M. Venzi, *Evolution of Asynchronous Cellular Automata for the Density Task*, Parallel Problem Solving from Nature - PPSN VII, Lecture Notes in Computer Science, Springer Berlin / Heidelberg, Volume 2439 (2002), pp. 934-943.

[15] S. Wolfram, *A New Kind of Science*, Wolfram Media (2002).

BABEŞ-BOLYAI UNIVERSITY, DEPARTMENT OF COMPUTER SCIENCE, CLUJ-NAPOCA, ROMANIA

E-mail address: `{anca,cchira}@cs.ubbcluj.ro`