STUDIA UNIV. BABEŞ–BOLYAI, INFORMATICA, Volume **XLIX**, Number 1, 2004

EVOLUTIONARY CLUSTERING USING ADAPTIVE PROTOTYPES

D. DUMITRESCU AND R. GORUNESCU

Abstract. Genetic chromodynamics has proven to be a very efficient metaheuristics for detecting multiple optima points.

It is our goal to show that it is extremely suitable in the field of clustering as well.

Keywords: clustering, evolutionary computation, genetic algorithms, genetic chromodynamics, multiple optima points, stepping stone, local interaction, weighted similarity measures.

1. Introduction

A genetic chromodynamics-based clustering method is proposed.

Genetic chromodynamics (GC) is not a particular evolutionary technique but merely a metaheuristics for maintaining population diversity and for detecting multiple optima. Its strategy is to form and maintain stable subpopulations that co-evolve and lead, at convergence, each to an optimum. It uses a variable sized solution population, a stepping stone search mechanism in connection with a local interaction principle, and a special operator for merging very similar individuals. Therefore the clustering mechanism we get by using these principles is very simple: each convergence point represents a cluster, no number of clusters are given in advance and we will obtain in the end both the optimal number of groups and the optimal classification.

2. Genetic Chromodynamics. Basic Ideas

Here we summarize briefly the main underlying principles of genetic chromodynamics metaheuristics[2]:

- the population size is variable;
- the sub-population structure is not predefined;
- there is a stepping stone mechanism;

Received by the editors: September 12, 2003.

2000 *Mathematics Subject Classification.* 62H30, 68T20.

1998 *CR Categories and Descriptors.* I.5.3 [Pattern Recognition]: Clustering – *Clustering algorithms*; I.2.8 [**Artificial Intelligence**]: Problem Solving, Control Methods, and Search – *Heuristic methods*.

- the local interaction principle holds;
- recombination and mutation are exclusive operators when applied to the same individual;
- a new operator called merging is introduced;
- the algorithm stops when, after an a priori fixed number of iterations, no significant change occurs in the population.

The algorithm starts with a large initial population whose size may be reduced at every generation. The sub-populations have an arbitrary structure and they become better separated with each iteration. The stepping stone search mechanism provides the possibility that each individual takes part in the process of forming the new generation. Thus, by making use of it, each individual is considered for mating. Its mate will be determined by applying a local selection scheme. This local interaction principle is a natural thing to do, since from a biological point of view it is more likely that individuals from the same sub-population mate rather than from different ones. If a second chromosome is found within that range, then they will recombine. Else mutation will be applied to the current individual. It is also possible to use a global search mechanism for the mate instead, in some situations, but this is of no interest with respect to the problem at hand. As an observation, selection is carried out both globally - it is the case of the first chromosome — and locally — as seen for the second parent, if there is one. If two chromosomes are very similar, they are merged into a single one, by usually taking their mean.

3. Genetic Chromodynamics Clustering

The clustering method we propose uses GC principles. A standard genetic chromodynamics model is used, together with some data-related features. The data points allow both numerical and nominal attributes. We have tried to develop an algorithm that would work perfectly with any kind of data. And it almost does. The problem is that for a special type of data, e.g. geometrical instances, the evaluation function will not work in most of the cases, because it should be more or less problem dependent.

3.1. **Representation. Initial population.**

Each instance from our database represents a chromosome, every attribute of our record being thus a gene. For example if we are dealing with a database for describing the features of different candidates present for some secretary positions, with the structure

(typing skills, number of foreign languages known, years of experience),

then all the chromosomes will have the exact same structure. Let us denote by c the current chromosome. Therefore its value could be for example:

$$c = ((\textit{computer - 0.25, typewriter - 0.14}),\ 2,\ 4).$$

An important feature of our approach is that each chromosome allows numerical as well as nominal attributes. The problem we were faced with was that no nominal variables can be used in our mathematical computations. Thus we have represented our nominal data as fuzzy.

The initial population is made of the data points.

3.2. **Fitness function.** First of all, we have to define the similarity measure between two chromosomes, since our function is built upon its expression. We have used a weighted similarity measure, since each attribute of our data has a different degree of importance in the field they are extracted from.

$$distance(a,b) = \sum_{k=1}^{n} compare(a_k, b_k),$$

where a and b are the two chromosomes and n represents the number of attributes.

At this point there are two cases. First of all, if we are dealing with numerical attributes, the difference between the two attributes was considered the square weighted Euclidian similarity measure, that is:

$$compare(a_k, b_k) = (a_k - b_k)^2 weight_k,$$

where $weight_k$ is a positive number specifying the importance of attribute k.

In the other case, of the nominal attributes, the formula was considered the max-min distance specific to fuzzy data:

$$compare(a_k, b_k) = \max(\min_{i=1}^{n_k})(a_k^i, 1 - b_k^i) weight_k,$$

where n_k is the number of values for the k-th attribute of the chromosome.

Now the expression of the fitness value is as follows:

$$eval(pop_i) = \sum_{i=1}^{popNo} 1/e^{distance(pop_i, pop_j)},$$

where pop denotes the vector representing the population of chromosomes, $popNo$ meaning the current population size and pop_i meaning the current chromosome.

3.3. **Selection operator.** The mate for the current chromosome, c, will be selected within its pre-determined mating region. This region is defined, mathematically speaking, as the closed ball $V(c, r)$, where we specify the radius, r. Proportional selection will be used from this point on.

3.4. **Variation operators.** Standard recombination and mutation operators are used for guiding the search process. Merging is an additional variation operator.

id	outlook	temperature	humidity	windy
x_1	(sunny-0.78,overcast-0.45,rainy-0.20)	(hot-0.90,mild-0.50,cool-0.10)	(high-0.78,normal-0.12)	(true-0.13,false-0.90)
x_2	(sunny-0.80,overcast-0.34,rainy-0.10)	(hot-0.80,mild-0.40,cool-0.20)	(high-0.80,normal-0.20)	(true-0.89,false-0.23)
x_3	(sunny-0.30,overcast-0.85,rainy-0.34)	(hot-0.90,mild-0.30,cool-0.10)	(high-0.90,normal-0.30)	(true-0.16,false-0.77)
x_4	(sunny-0.10,overcast-0.50,rainy-0.90)	(hot-0.40,mild-0.80,cool-0.50)	(high-0.70,normal-0.10)	(true-0.22,false-0.86)
x_5	(sunny-0.13,overcast-0.50,rainy-0.70)	(hot-0.10,mild-0.50,cool-0.80)	(high-0.30,normal-0.80)	(true-0.15,false-0.88)
x_6	(sunny-0.20,overcast-0.40,rainy-0.87)	(hot-0.20,mild-0.40,cool-0.90)	(high-0.30,normal-0.79)	(true-0.77,false-0.30)
x_7	(sunny-0.50,overcast-0.80,rainy-0.30)	(hot-0.30,mild-0.20,cool-0.92)	(high-0.40,normal-0.98)	(true-0.89,false-0.20)
x_8	(sunny-0.90,overcast-0.70,rainy-0.10)	(hot-0.60,mild-0.80,cool-0.20)	(high-0.84,normal-0.22)	(true-0.14,false-0.88)
x_9	(sunny-0.78,overcast-0.34,rainy-0.20)	(hot-0.20,mild-0.60,cool-0.96)	(high-0.13,normal-0.95)	(true-0.10,false-0.98)
x_{10}	(sunny-0.10,overcast-0.50,rainy-0.70)	(hot-0.10,mild-0.90,cool-0.50)	(high-0.24,normal-0.87)	(true-0.34,false-0.68)
x_{11}	(sunny-0.80,overcast-0.30,rainy-0.10)	(hot-0.20,mild-0.87,cool-0.40)	(high-0.32,normal-0.89)	(true-0.56,false-0.45)
x_{12}	(sunny-0.40,overcast-0.90,rainy-0.30)	(hot-0.12,mild-0.90,cool-0.60)	(high-0.82,normal-0.30)	(true-0.85,false-0.30)
x_{13}	(sunny-0.20,overcast-0.90,rainy-0.50)	(hot-0.90,mild-0.50,cool-0.20)	(high-0.40,normal-0.80)	(true-0.65,false-0.22)
x_{14}	(sunny-0.10,overcast-0.30,rainy-0.90)	(hot-0.40,mild-0.78,cool-0.11)	(high-0.98,normal-0.14)	(true-0.94,false-0.12)
	0.2	0.3	0.1	0.4

TABLE 1. The weather data set

mating region	**mutation step**	**merging radius**
0.5	0.07	0.4

Table 2. Algorithm parameter values

3.4.1. *Merging.* Each chromosome from the current population will be taken into consideration, observing whether there are other chromosomes similar to it behind a certain threshold called merging radius. If that should be the case, the best one from the group will be kept, and the others will be deleted from the population.

3.5. **Stop condition.** The algorithm stops when, after a number of iterations, considered equal in value to the number of the objects in the data set, no new offsprings are accepted in the population.

The last population provides the optimal clustering. Its members correspond to the centers of the resulting clusters and they also hold the information regarding the distribution of the initial data points to these centers.

3.6. **Other parameter settings and experimental results.** Consider a fictional data set that describes the weather conditions for playing some unspecified game [7] given in Table 1.

We consider the values for the other parameters involved given in Table 2.

The corresponding classes are:

$$A_1 = \{x_2, x_{14}, x_{12}, x_4, x_3, x_1, x_8\},$$

$$A_2 = \{x_9, x_5\},$$

$$A_3 = \{x_{10}, x_7, x_6\},$$

$$A_4 = \{x_{11}\}$$

and

$$A_5 = \{x_{13}\}.$$

The run of the corresponding program provides in 9 out of 10 cases, the following grouping:

(i) instances x_6, x_7 in a cluster,
(ii) instances x_5, x_9 in a cluster,
(iii) instances x_2, x_{14} in a cluster, and
(iv) instances x_3, x_4, x_8, x_1 in a cluster.

3.7. **Conclusions and future work.** The initial population size is probably rather small considering only the instances in the data set. A population consisting of the data which is tried to be clustered, maybe with a slight modification in their values, on the one hand, and a double number of randomly generated data points, on the other hand, would probably lead to better results.

References

[1] Dumitrescu, D., Genetic Chromodynamics, Studia Universitatis Babes Bolyai, Ser. Informatica, 2000, 39–50

[2] Dumitrescu, D., A New Type of Evolutionary Metaheuristics, 2003. (to appear)

[3] Dumitrescu, D., Genetic Algorithms and Evolution Strategies, Blue Publishing House, Cluj-Napoca 2000

[4] Dumitrescu, D., Lazzerini, B., Jain, L., C., Dumitrescu, A., Evolutionary Computation, CRC Press, Boca Raton, Florida, 2000

[5] Dumitrescu D., Gorunescu R., Adaptive Prototypes in Evolutionary Clustering, Research Notes in Artificial Intelligence and Digital Communications, 103, 2003, 48–55

[6] Michalewicz, Z., Genetic Algorithms + Data Structures + Evolution Programs, 2nd edition, Springer Verlag, 1992

[7] Witten, I., H., Frank, E., Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations, Morgan Kaufmann, 1999

Faculty of Mathematics and Computer Science, Department of Computer Science, Babes-Bolyai University, 3400 Cluj - Napoca Romania
E-mail address: ddumitr@cs.ubbcluj.ro

Faculty of Mathematics and Computer Science, Department of Computer Science, University of Craiova, 13 Al. I. Cuza 1100 Craiova Romania
E-mail address: ruxandragorunescu@yahoo.com