Evolutionary Method of Population Classification According to Level of Social Resilience

Coulibaly Kpinna Tiekoura
Research Laboratory in Computer Science and Telecommunications (LARIT)
National Polytechnic Institute Houphoüet Boigny (INPHB)
Abidjan, Ivory Coast

Babri Michel
National Polytechnic Institute
Abidjan, Ivory Coast

Souleymane Oumtanaga
National Polytechnic Institute
Abidjan, Ivory Coast

Brou Konan Marcellin
National Polytechnic Institute
Yamoussoukro,
Ivory Coast

***Abstract*—Following the many natural disasters and global socio-economic upheavals of the 21st century, the concept of resilience is increasingly the subject of much research aimed at finding appropriate responses to these traumas. However, most existing work on resilience is limited to a broad cross-disciplinary panel of non-operational theoretical approaches. Thus, the study of the processes of social resilience is confronted with difficulties of modeling and a lack of appropriate analysis tools. However, the existing stratification methods are too general to take into account the specificities of the resilience and are difficult to use for non-specialists in modeling. In addition, most traditional methods of partition research have limitations including their inability to effectively exploit the research space. In this paper, we propose a classification algorithm based on the technique of genetic algorithms and adapted to the context of social resilience. Our objective function, after penalization by two criteria, allows to explore widely the space of research for solutions while favoring classes quite homogeneous and well separated between them.**

Keywords—genetic algorithm; Unsupervised classification; social resilience; Partitioning method

I. Introduction

Resilience is a polysemic concept that is studied in several fields including sociology, ecology, economics, computer science and psychology. Resulting from the physics of materials where it designates the ability of a system to resume its initial equilibrium after a deformation, resilience is the segment of many researches these days. However, analysis of the literature in this area reveals a lack of operational approaches. This paper is a contribution to the process of operationalizing the concept of social resilience that is defined by the French ethologist Boris Cyrulnik as the ability of a person, a social group or an environment to overcome suffering or trauma [1].

One of the fundamental principles of clustering is to ensure the partitioning of a set of objects so that the elements of the same group are as similar as possible and that the various groups are distinct among themselves. There are several families of classification methods, the most widely used of which are hierarchical classification methods and partitioning methods. These methods, however, present a certain number of not inconsiderable drawbacks. In effect, hierarchical or agglomerative methods are limited to small sets of sizes due to the fact that they store in memory a dissimilarity matrix whose size is quadratic as a function of the number of vertices. As for partition-based classification methods, in addition to generating sub-optimal results dependent on the initial partition, they exploit only a small part of the solution search space. This calls for the need to develop other methods offering more possibilities for exploring this research space. The genetic algorithms developed by John Holland [2] respond to this concern. Indeed, these algorithms, inspired by the principles of the neo-Darwinian natural evolution are known for their effectiveness in exploring quite large and complex research spaces. They generally allow to generate good solutions following the application of a cycle of operations (selection, crossing, mutation). One of the interests of our proposal is its ability to identify different dominant characteristic groups in a given population. It thus adapts well to the context of social resilience [3] especially in the study of social stratification within a population victim of a traumatic shock. In other words, an application of this situation could be the identification of the social groupings of a population according to the degrees of resilience of the different individuals facing a traumatic shock.

In this paper, after a presentation of the genetic algorithms and some work done, we present our proposition followed by a conclusion.

II. Genetic Algorithms

A. Principle

Genetic algorithms are part of the stochastic optimization algorithms [4][5]. They represent a modeling of natural evolution to solve a research problem. Their goal is to evolve a set of solutions towards an optimal solution. To do so, the algorithm randomly generates a population of individuals

(chromosomes) and proceeds by successive iteration to generate new individuals by applying different selection, crossover and mutation operators until reaching a stop criterion. An evaluation function makes it possible to evaluate beforehand each chromosome candidate for the selection. As a result of the evaluation, a sub-population which is victorious of chromosomes is retained for reproduction. The crossover and mutation operations are carried out respectively according to a crossing probability (Pc) and a mutation probability (Pm).

B. Genetic operators

- *The selection operator:* It allows parents to be chosen for reproduction according to an evaluation function called fitness. Generally, for a population of n individuals, n / 2 is selected for reproduction through the crossing step. We distinguish several techniques of selection in the literature of which the most used are the technique of roulette or the "roulette-wheel", the technique of the tournament, the technique of the rank (ranking) and the universal stochastic selection [6][7][8].

- *The crossover operator:* This operator makes it possible to cross the $n/2$ pre-selected parents to generate new children who have characteristics of their parents. It thus complements the population of $n/2$ individuals to n individuals. The crossing is done according to a probability P_c which increases with the number of cross points. Three main crossover operators are distinguished: crossing at one point, crossing at n-points ($n \geqslant 2$) and uniform crossing.

- *The mutation operator:* This operation consists in modifying, randomly, the value of an allele following a mutation probability P_m, which is generally very low. A too high mutation probability could lead to a suboptimal solution.

III. State of the Art

M. Merzougui et al. [9] propose an improvement of the unsupervised classification algorithm Isodata through its main parameters. Indeed, because the results of "Isodata" are intrinsically linked to a threshold from which a class is divided and another threshold from which two classes are merged, the authors use the genetic algorithms to determine these two optimal thresholds. This has improved the quality of this algorithm. However, other parameters are empirically fixed, such as the bounds of the chromosome membership interval of the initial population. This helps to always influence the results of the algorithm despite some performance.

Stephane Legrand [10] proposes a genetic program to discover subsets of homogeneous and distinct data in a file called "Zoo". Thus, it represents an individual in the form of a tree of logical formulas. Each logical formula consists of a variable number of predicates. It evaluates the individuals from an evaluation function based on a measure of homogeneity (H) and a measure of the separability (S) of the data subsets and equal to: $fitness = H + \mu S$. It applies a coefficient μ to the measurement of separability in order to vary the relative weight of the two measurements. It considers the homogeneity H as the weighted average of the homogeneity of the various subsets and the separability S as the weighted average of the distances between the centroids of the subsets. The convergence of the algorithm is not proved. Moreover, the arbitrary choice of the coefficient μ greatly influences the quality of the results.

Maulik et al. [11] propose a clustering method based on a genetic algorithm in which each element is assigned to the nearest centroid so as to form clusters. Each time, the centroids are recalculated as the average of the elements of the same group and the inverse of the intra-group inertia is then calculated to reduce to a maximization problem. The authors use a representation of the individuals in the form of k tuples and encode the coordinates of the k centroids by real numbers. Initially, they initialize an initial population of P chromosomes randomly. Moreover, the selection technique used is an elitist proportional castor, which allows to retain the best candidate of the previous generation. Unlike the previous algorithm, it converge towards the global optimum. However, it does not solve the question of non-consistent classes (having one element) and separability between classes.

Greene [12] proposes a method that generates hierarchies of partitions. It begins with a top-down method by which the initial population is subdivided into several subpopulations. Evaluation consists in optimizing a function dependent on intra-group and inter-group inertia and on the size of the constituted groups. To limit the influence of initial conditions including the order of insertion of objects in the tree, the author proposes to generate the best possible tree by applying a genetic algorithm. An initial population of trees is generated by choosing a random order of insertion of the objects. The different selection, crossing and mutation operators are applied. The selection is made by the elitist proportional roller technique where the two best solutions are retained after evaluating the quality of each tree. For crossing, it chooses the best branches of the first level of each tree. The algorithm takes into account any objects that are repeated in two classes or missing in the partition. In the first case, the object is maintained in the best class and in the second case, it is simply reinserted. This algorithm unfortunately does not provide information on the optimality of the generated solution.

IV. Our Proposal

A. Motivations

In order to study the processes of social resilience, researchers often use classification methods that are often poorly adapted to this domain because they do not respect certain specificities linked to the concept, particularly its unobservable, temporal and dynamic aspect.

Moreover, the most widely used classification methods present a certain number of notable inconveniences including their inadequacy to large data sets (for hierarchical algorithms) and the very limited exploitation of the solution search space (For partitioning algorithms). All these limitations can contribute to biased results. Thus, we propose to develop a partitioning method hybridized with the technique of genetic algorithms for the classification of data of social resilience. This method, in addition to taking into account the specificities of social resilience, has the ability to explore a large solution-

seeking space and can be applied to larger sets of data. In addition, it can be adapted to any field of study. In this paper, the algorithm is applied to a real data set, obtained from a survey of a sample of people in relation to the recent post-electoral crisis in Ivory Coast. The objective is to find the main sociological groupings caused by the trauma of this crisis within the population studied. In a broader case study, the results of our algorithm can be used by the actors to facilitate the making of certain decisions in favor of the resilience of the traumatized individuals.

B. Notation

n : The number of objects to be classified;

T : The time horizon for estimating the resilience of individuals;

Q : The total number of classes;

$\Omega^{t:T}$: Set of objects to be classified according to the information collected over the period from t to T;

$Pop(t)$: Population of individuals (chromosomes) at time t;

ξ_i^t : Estimation of the resilience of the individual at time t. $i \in [1...n]$

C_q : The q[th] class as $q \in [1...Q]$

P_{cr} : Crossover probability;

P_{mut} : Mutation probability;

n_q : Number of objects in class C_q ;

I_j^t : j[th] partition of the Set $\Omega^{t:T}$ at time t;

K : Population size (Number of partitions);

M : Maximum number of iteration (generation);

$Matr$: Dissimilarity matrix;

$fit_{init}^t()$: Initial objective function;

$fit_p^t()$: Objective function after penalization;

$fit_p^t(I^t)$: Evaluation value of the individual I^t ;

g_q : Center of gravity of the class C_q;

g : Center of gravity of the whole point cloud;

d : Euclidean distance;

δ^α : Percentage of classes whose numbers are less than 1 (minimum number);

δ^β : Percentage of classes with closely spaced classes;

δ : The overall penalty rates;

A : All classes whose size is less than or equal to 1;

B : All non-homogeneous classes;

$card(A)$: Cardinality of the set A.

C. Representation of Individuals

An individual is a class partition and is a potential solution to the problem. In the context of genetic algorithms, it is represented by a chromosome composed of genes. Each gene represents a class and consists of a sequence of binary digits (0, 1). In this paper, we use a presence / absence coding where the presence of an object in a class is marked by the number 1 and its absence by the number 0.

Example of coding of our chromosome:

Either a given set of 12 traumatized persons each represented by its social resilience value ξ_i^t :

$$\Omega^{t:T} = \left\{ \xi_1^t ; \xi_2^t ; \xi_3^t ; \xi_4^t ; \xi_5^t ; \xi_6^t ; \xi_7^t ; \xi_8^t ; \xi_9^t ; \xi_{10}^t ; \xi_{11}^t ; \xi_{12}^t \right\}$$

A random partitioning of this set made it possible to obtain the following two partitions:

$$I_1^t = \left\{ (\xi_1^t ; \xi_3^t ; \xi_4^t ; \xi_6^t ; \xi_8^t), (\xi_2^t ; \xi_5^t ; \xi_9^t ; \xi_{12}^t), (\xi_7^t ; \xi_{10}^t ; \xi_{11}^t) \right\}$$

$$I_2^t = \left\{ (\xi_1^t ; \xi_5^t ; \xi_9^t ; \xi_{11}^t), (\xi_2^t ; \xi_3^t ; \xi_4^t ; \xi_6^t ; \xi_8^t), (\xi_7^t ; \xi_{10}^t ; \xi_{12}^t) \right\}$$

The coding of these partitions gives the following chromosomes:

$$I_1^t = \{(101101010000)(010010001001)(000000100110)\}$$

$$I_2^t = \{(100010001010)(011101010000)(000000100101)\}$$

D. Our evaluation function

In order to obtain homogeneous classes, we propose an evaluation function which minimizes the ratio of intra class inertia by total inertia. It is as follows:

$$fit_{init}^t(I_i^t) = \frac{1/n \sum_{q=1}^{Q} \sum_{\xi_i^t \in C_q} d^2(\xi_i^t, g_q)}{1/n \sum_{i=1}^{n} d^2(\xi_i^t, g)}$$

$$fit_{init}^t(I_i^t) = \frac{\sum_{q=1}^{Q} \sum_{\xi_i^t \in C_q} d^2(\xi_i^t, g_q)}{\sum_{i=1}^{n} d^2(\xi_i^t, g)} \quad (1)$$

Since the classification often leads to empty classes or classes containing a single element, we propose to penalize the above evaluation function by a rate which is the percentage of classes whose numbers are less than or equal to one. Moreover, in order to obtain homogeneous classes well separated from each other, we propose to penalize also the objective function by the percentage of classes whose class centers are relatively close. We obtain the global penalization rate δ such as:

$$\delta = \sum_{j \in \{\alpha, \beta\}} \delta^j \quad (2)$$

$$\delta^{\alpha} = \frac{card(A)}{Q} \quad (3)$$

$$\delta^{\beta} = \frac{card(B)}{Q(Q-1)/2} \quad (4)$$

Therefore, the penalized objective function is calculated as follows:

$$fit_p^t(I_i^t) = fit_{init}^t(I_i^t) - \delta\, fit_{init}^t(I_i^t)$$

$$fit_p^t(I_i^t) = (1-\delta)\, fit_{init}^t(I_i^t)$$

$$fit_p^t(I_i^t) = (1-\delta)\frac{\sum_{q=1}^{Q} \sum_{\xi_i^t \in C_q} d^2(\xi_i^t, g_q)}{\sum_{i=1}^{n} d^2(\xi_i^t, g)} \quad (5)$$

E. The choice of parameters

- For the selection, we use the roulette method which is similar to a lottery wheel on which each individual is represented by a sector equivalent to his fitness value. At each turn of the wheel, each individual has a probability of being selected proportional to its fitness value :

$$prob(I_i) = fit_i^t(I_i) / \sum_{i=1}^{n} fit_i^t(I_i) \quad (6)$$

- For the crossing of the individuals, we use the crossing at a point of cut chosen randomly among the $l-1$ possible points (l representing the length of a chromosome). At this level, we choose a crossing probability as advocated by Goldberg [13]. In our case, $P_{cr} = 0,6$
- For the mutation, we opt for a mutation probability inversely proportional to the size of our population, i.e. $P_{mut} = 0,08$.
- As criterion for stopping our algorithm, we retain the maximum number of iterations (or generations) fixed.

F. Proposed Algorithm

Algorithm: Significant group identification algorithm (AlgoGene)

INPUT: dissimilarity matrix (*Matr*),
Maximum number of classes (Q),
Population size (K)
Maximum number of generations (J)

OUTPUT: Partition $I_j^t = \{C_1,...,C_q\}$, which *minimizes the most fitness function*

BEGIN

1. *//Random generation of the initial population*
   - 1.1. Choose random Q centers of gravity g_q
   - 1.2. Assign each observation to the nearest center: $I_0^0 = \{C_1,...,C_q\}$
   - 1.3. Calculate the new class centers

$$g_q \leftarrow \sum_{\xi_i^t \in C_q} \xi_i^t \Big/ n_q$$

   - 1.4. Repeat steps 1.2 and 1.3 until the initial fixed population size (K)
   - 1.5. Return the initial population to optimize

$$pop(0) \leftarrow \{I_0^0,...,I_k^0\}$$

2. *// Optimization of the initial population*
   - 2.1. Coding the initial population (pop (0)) to binary
   - 2.2. Evaluate the initial population

Repeat

   - 2.3. Select from the roulette wheel K/2 parents individuals (P (t) in P (t-1))
   - 2.4. Cross at a point the selected individuals with a probability P_{cr} .
   - 2.5. Making a mutation on the descendants obtained with a probability P_{mut} .
   - 2.6. Returning the new population

$$pop(t) \leftarrow pop(t-1) + descendants$$

   - 2.7. Evaluate the new population found

$$fit_p^t(I_j^t) \leftarrow (1-\delta)\frac{\sum_{q=1}^{Q} \sum_{\xi_i^t \in C_q} d^2(\xi_i^t, g_q)}{\sum_{i=1}^{n} d^2(\xi_i^t, g)}$$

Until: Number of generations > M

Return the best partition $I_j^t = \{C_1,...,C_q\}$

END

G. Results and interpretations

For the application of our algorithm, we use a real data set, obtained from a survey of a sample of one hundred (100) individuals (see Table 1). This survey relates to the trauma caused by the recent post-election crisis in these people.

TABLE I. EXTRACT FROM THE DATABASE USED

	COH1	CONSC1	HUM1	APS1	COH2
1	0.1197035	-0.1519177	-0.1274281	0.5022938	0.07017892
2	0.7304356	0.9966927	0.8014604	0.5022938	0.579762
3	0.3611916	0.2965908	0.3199828	0.7619627	0.579762
4	0.4889475	1.409642	-1.433323	1.329501	-1.113173
5	1.341168	0.4579468	0.5947884	0.4540931	0.579762
6	-0.1217846	0.2965908	0.3199828	1.069832	-0.4941326
7	-1.712493	-0.9778162	-0.9518449	-0.9406526	-1.513299
8	0.7304356	1.409642	1.351072	1.329501	0.8071893
9	0.3611916	-0.1163584	0.07924394	-0.9888533	0.07017892
10	1.341168	1.696794	1.351072	1.329501	1.5442
11	-3.544689	-0.9778162	-0.1614949	-0.1134453	0.2976062
12	-0.1217846	-0.403511	0.07924394	-0.8924519	0.2976062
13	0.4889475	0.2965908	1.351072	1.329501	-0.2119768
14	-0.4910286	1.122489	1.351072	1.021632	0.2976062
15	-2.564713	-0.1163584	-0.6770393	-0.9406526	-1.513299
16	-0.1217846	-0.9778162	0.3540495	0.7619627	-0.1572484
17	0.1197035	-0.9422569	-3.804373	-0.06524462	-1.95964
18	-0.1217846	-0.1163584	-0.1955617	0.1462235	0.01545047
19	-1.357273	0.2965908	-1.226651	-1.363589	-1.285871
20	-0.2495405	0.870896	-0.4363005	0.5022938	0.5250336
21	0.7304356	0.2965908	0.8355272	0.5022938	-0.03927799
22	-1.101761	-0.6906636	-0.1955617	-0.6327831	-1.231143
23	0.1197035	-0.403511	1.110333	1.329501	-0.1572484
24	0.7304356	1.409642	1.351072	1.329501	1.262044
25	-0.2495405	0.2965908	1.076266	1.021632	0.8071893
26	-0.9880286	-2.916766	-2.257739	-1.76786	-3.551631
27	-1.101761	-1.103613	-0.1955617	-0.3731142	-1.45857
28	-0.1217846	-0.9422569	-0.5044341	-0.4213149	-1.003716
29	0.1197035	-0.403511	-0.1955617	-0.6327831	-0.2119768
30	0.1197035	0.2965908	0.5607216	0.4540931	0.5250336
31	-0.4910286	-0.6906636	-0.5044341	-0.4213149	-0.5488611
32	0.7304356	0.5837434	-0.1274281	1.069832	0.8071893
33	1.341168	0.04499754	-2.325873	-0.7291845	-1.058444
34	0.3611916	0.9966927	0.8355272	0.4540931	0.5250336
35	0.4889475	0.1707942	-0.7111061	-0.1134453	-0.2667053
36	-2.564713	-2.629613	-2.325873	-2.999338	-1.850183
37	0.1197035	0.5837434	0.3199828	0.1944242	0.5250336
38	-0.6187845	-1.103613	-0.4363005	-0.6327831	-0.7215599
39	1.341168	1.283845	0.8014604	-0.4695156	0.8071893

The objective is to identify the significant groupings that can be obtained from this population in order to make decisions. After simulations, it appears that the best classification result is obtained for 3 classes with a Rand index of 0.89 after 150 iterations (generations) (see Table 2). According to this classification, 18 individuals are in the first class, 32 individuals are in the second class and the other 50 individuals are in the third class.

TABLE II. TABLE OF RAND INDICES OBTAINED FOR 2, 3, 4, 5 AND 6 CLASSES

	2	*3*	*4*	*5*	*6*
Rand Index	*0,629*	*0,891*	*0,87*	*0,859*	*0,53*

The following figures show the best groupings obtained respectively for 3, 4, 5 and 6 classes.

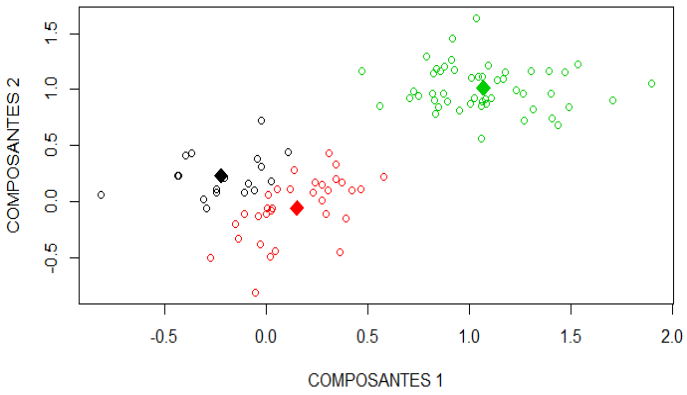

Fig. 1. Grouping into 3 classes

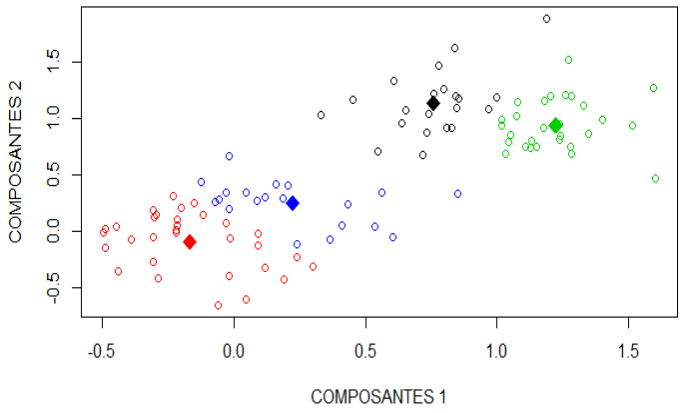

Fig. 2. Grouping into 4 classes

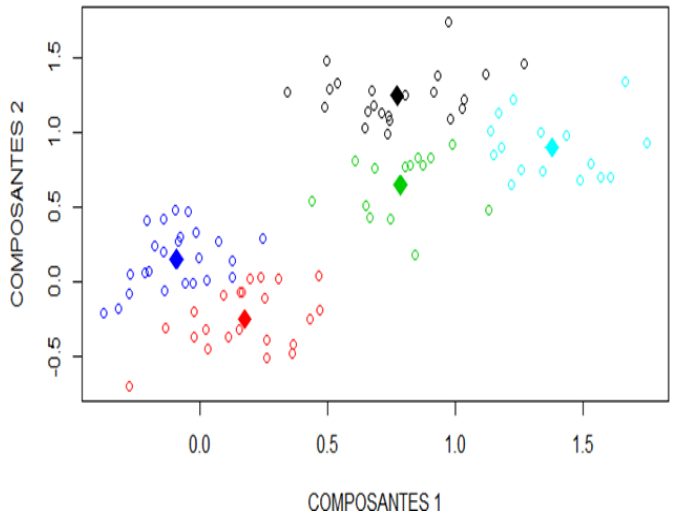

Fig. 3. Grouping into 5 classes

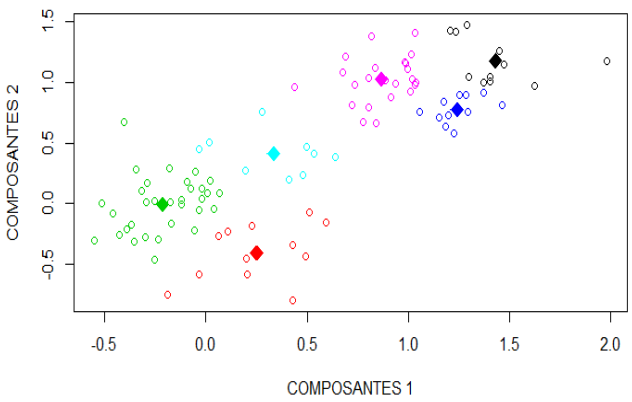

Fig. 4. Grouping into 6 classes

On the other hand, experimentation has shown that, from 150 iterations, the classes are more closely grouped and distinct.

V. Comparison of our Proposal with Other Works

These hybrid algorithms are very different which makes them very difficult to compare. However, in the table below, we present some points of comparison.

TABLE III. Comparative Table of Our Algorithm (Algogene) with Other Works

	Merzougui et al	**Stephane Legrand**	**Maulik et al**	**Greene**	**AlgoGene**
Model	Partition in k fixed classes	Partition in k fixed classes	Partition in k fixed classes	Partition Hierarchy (free k)	Partition in k fixed classes
Validity classes	Overlap from 6 classes	Overlap between classes	Always valid	Not valid (empty clases + duplicates)	Always valid
Convergence	Converge to global optimum	No indication on convergence	Converge to global optimum	Not standard, depending on initial conditions	Converge to global optimum
Separability	Not respected at a certain level of classes	Much related to the parameter μ	Not respected for certain classes	Respected	Respected
Representation	Real number encoding	Tree of logical formulas	Centroids (actual coordinates)	Tree	Binary coding absence / presence

VI. Conclusion

We proposed a hybrid-partitioning algorithm for the identification of significant groups as a function of the levels of resilience. It generates from a traditional method of partitioning partitions, which are then optimized using the technique of genetic algorithms to give the best partition possible: one that minimizes the most intra-class inertia and promotes classes while eliminating classes that have only one element.

The results of our simulations showed that the algorithm converges after 150 iterations by providing a solution corresponding to the expected objective. The Rand index (0.89) obtained without doubt translates the good performance of our algorithm. In future work, we intend to extend this algorithm to other areas of study other than social resilience to test its robustness.

References

[1] Boris Cyrulnik « Manifeste pour la résilience ». *Spirale* 2/2001, n°18, p. 77-82, 2001.

[2] J. H. Holland. Adaptation in Natural and Artificial Systems. *University of Michigan Press, Ann Arbor, MI*, USA, 1975.

[3] Boris Cyrulnik. « *Le murmure des fantômes* ». Odile Jacob, 2003.

[4] Duflo, Marie. *Algorithmes stochastiques*. 1996.

[5] Back, Thomas. Evolutionary algorithms in theory and practice: *evolution strategies, evolutionary programming, genetic algorithms*. Oxford university press, 1996.

[6] Sean Luke. Essentials of Metaheuristics. Lulu, *second edition*, 2013.

[7] T. Blickle & L. Thiele. A comparison of selection schemes used in genetic algorithms. *Evolutionary Computation*, 4(11): 311–347, 1995.

[8] D. E. Goldberg & K. Deb. A comparative analysis of selection schemes used in genetic algorithms. *In Foundations of Genetic Algorithms*, pp. 69–93. Morgan Kaufmann, 1991.

[9] M Merzougui, M. Nasri, Ahmad El Allaoui. Isodata et les algorithms génétiques pour une classification non supervisée. Présenté au Congrès Méditerranéen des Télécommunications (CMT'16), 12-13 mai 2016, At Téhouan, Maroc. Répéré à https://www.researchgate.net/publication/303276467_Isodata_et_les_algorithmes_genetiques_pour_une_classification_non_supervisee.

[10] Stephane Legrand, Résolution de problème de classification par algorithmes évolutionnaires grâce au logiciel DEAP, octobre 2014, repéré à https://stephanelegrand.files.wordpress.com/2014/10/classification_algo_evol.pdf

[11] U. Maulik and S. Bandyopadhyay. Genetic algorithm-based clustering technique. *Pattern Becognition*, 33: 1455-1465, 2000.

[12] William A. Greene. Unsupervised hierarchical clustering via a genetic algorithm. In *Proceedings of the 2003 Congress on Evolution*, pages 998-1005, 2003.

[13] D. E. Goldberg. Genetic Algorithms in Search, Optimization, and Machine Learning. Studies in *Computational Intelligence. Addison-Wesley Longman Publishing Co., Inc., 1st edition*, 1989. ISBN 0201157675.