HAL
archives-ouvertes.fr

Anomaly Detection with the Voronoi Diagram Evolutionary Algorithm

Luis Martí, Arsene Fansi-Tchango, Laurent Navarro, Marc Schoenauer

▶ **To cite this version:**

Luis Martí, Arsene Fansi-Tchango, Laurent Navarro, Marc Schoenauer. Anomaly Detection with the Voronoi Diagram Evolutionary Algorithm. J. Handl; E. Hart; P.R. Lewis; M. López-Ibáñez; G. Ochoa; B. Paechter. Parallel Problem Solving from Nature – PPSN XIV, Sep 2016, Edinburgh, United Kingdom. Springer Verlag, 9921, pp.697-706, 2016, LNCS. <10.1007/978-3-319-45823-6_65>. <hal-01387621>

HAL Id: hal-01387621
https://hal.inria.fr/hal-01387621

Submitted on 26 Oct 2016

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Anomaly Detection with the Voronoi Diagram Evolutionary Algorithm

Luis Martí[1,2], Arsene Fansi-Tchango[3], Laurent Navarro[3], and Marc Schoenauer[1]

[1] TAO team, CNRS/INRIA/LRI, Université Paris-Saclay, Paris, France.
[2] Universidade Federal Flumnense, Niterói (RJ) Brazil.
[3] Thalés Research, Paris, France.

Abstract. This paper presents the Voronoi diagram-based evolutionary algorithm (VorEAl). VorEAl partitions input space in abnormal/normal subsets using Voronoi diagrams. Diagrams are evolved using a multi-objective bio-inspired approach in order to conjointly optimize classification metrics while also being able to represent areas of the data space that are not present in the training dataset. As part of the paper VorEAl is experimentally validated and contrasted with similar approaches.

1 Introduction

Anomalous Internet traffic detection is a major question of computer network security. Intrusion detection systems (IDSs) [9] have proposed with the intention of tackling this issue. They are meant to protect a network by providing a line of defense that is able to detect and react to network attacks. Two main approaches are used when building an IDS: i) misuse-based and ii) anomaly-based detection. While the former focuses on detecting attacks that follow a known pattern or signature, the latter is interested in building a model representing the system's normal behavior while assuming all deviated activities to be anomalous or intrusions. Because of that fact anomaly detection has received increasing attention in the recent past.

Anomaly detection has been addressed with different approaches (see [2] for a survey). Among nature-inspired approaches artificial immune systems (AISs) [7] have received an special attention.

This paper proposes the Voronoi diagram-based evolutionary algorithm (VorEAl). VorEAl is inspired on AISs and the representations that had been proposed for evolutionary shape design consolidating previous progresses made in this direction [8]. Its main distinctive feature is that it evolves Voronoi diagram-based representations for normal/abnormal regions of the search space. Such representation offers a flexible and compact alternative to some common representations used in AIS such as hyper-spheres and hyper-rectangles. VorEAl applies a multi-objective approach that takes into account the detection accuracy and other especially devised volume-based methods that promotes the emergence of solutions that also adequately represent areas of the input space

where no normal data has been received and, therefore, should represent anomalies. As in any multi-objective approach, the algorithm produces a set of trade-off solutions. VorEAl applies a committee approach that is based on the best (in term of *a priori* given set of preferences) subset of those solution.

The paper is organized as follows. Section 2 presents the context of AIS and some existing approaches to anomaly detection. Section 3 introduces the Voronoi representation for abnormal and normal input subsets, together with the variation operators and objective functions used to evolve it and VorEAl as a whole. Section 4 introduces our methodology for the experimental validation of VorEAl, also presenting the results of the study and comparing them with other approaches from the literature. Finally, Section 5 discusses the results and sketches some further research directions.

2 Foundations

There has been a consistent interest by the community on proposing nature-inspired approaches to anomaly detection. In this context, AISs have attracted attention as they embody an analogy to the biological immune system. They are particularly appealing for anomaly detection problems as they capture the ability of the biological system of telling apart normal body cells from pathogens. That is, from a computational perspective, they create a model that is able to discriminate between normal (self) and abnormal (non-self) objects. This feature make AISs specially suited to be applied in the context of anomaly-based IDSs.

In order to extend AISs' performance it is necessary to apply algorithms that combine a powerful representation capacity as well as the possibility of adequately adapting that capacity to meet the problem characteristics.

Voronoi diagrams are geometrical constructs that were known by ancient Greeks. Any set of points, known as *Voronoi sites*, in a given n-dimensional Euclidean space $\mathcal{E}$ defines a *Voronoi diagram*, i.e., a partition of that space into *Voronoi cells*: the cell corresponding to a given site S is the set of points whose closest site is S. The boundaries between Voronoi cells are the medians of the $[S_i S_j]$ segments, for neighbor Voronoi sites S_i and S_j. Though originally defined in two or three dimensions, there exist several algorithmic procedures to efficiently compute Voronoi diagrams in any dimension.

Voronoi diagrams offer a compact representation for shapes (surfaces in 2D and volumes in 3D, for instance), by attaching to each Voronoi cell (or, equivalently, to the corresponding Voronoi site), a Boolean label. The resulting Voronoi diagram is a partition of the space into 2 subsets: the "true" cells are the shape/volume, and the "false" cells are the outside of the shape/volume. The *genotype* is here a (variable length) list of labeled Voronoi sites, and the phenotype is the corresponding partition in the space into two subsets. More generally, any piece-wise constant function on the underlying space can be represented by a similar representation by using real-valued labels. Such representation has been successfully used in the context of Evolutionary Optimum Design [5,10]. In particular, it has been demonstrated that the local complexity of the representation

```
function mutate_voronoi(I, p_s, p_f, p_t, p_+, p_−, η)
  ▷ I, individual to be mutated.
  ▷ p_s ∈ [0,1], prob. of mutating a site.
  ▷ p_f ∈ [0,1], prob. of mutating a site feature (coordinate).
  ▷ p_t ∈ [0,1], prob. of changing the label of a site.
  ▷ p_+ ∈ [0,1], prob. of adding a new site.
  ▷ p_− ∈ [0,1], prob. of removing a site.
  ▷ η ∈ (0,∞], learning rate.
  for all S ∈ I do
    if U[0,1) < p_s then
      for all x ∈ S do
        if U[0,1) < p_f then
          x ← mutate_log_normal(x, η)
      if U[0,1) < p_t then
        S.ℓ ← switch_label(S.ℓ).
  if U[0,1) < p_+ then
    I ← I ∪ {random_site}.
  if U[0,1) < p_− then
    i ← U[1,|I|); I ← I \ {I(i)}.
  return I, mutated individual.
```

Fig. 1: Mutation of a Voronoi diagram.

can also be adjusted by evolution: in regions of the space where the shape has a complex boundary, several Voronoi sites will be used, whereas only a few of them will be necessary elsewhere.

In the context of classification, the target phenotypes are partitions of the parameter space into positive and negative examples (in the case of 2 classes), and can hence also be represented by Voronoi diagrams with Boolean labels —or with labels taken from a finite alphabet in the case of more than 2 classes.

3 The Voronoi diagrams-based evolutionary algorithm

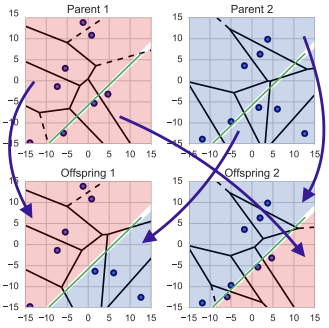
We now discuss the building blocks of VorEAl. In particular, we present variation operators, the possible strategies used for evaluating the individuals and how these elements are assembled together to form the algorithm.

3.1 Variation operators

The genotypes of Voronoi representations is a variable length list of Voronoi sites $(S_1, \ldots, S_p)$, with $p \in [P_{\min}, P_{\max}]$, where each site is defined by its n coordinates in $\mathcal{E}$. Each site S has an associated label $S.\ell$ that determines how a point that falls within the corresponding cell is classified.

function `crossover_voronoi`$(\mathcal{I}_1, \mathcal{I}_2)$
▷ $\mathcal{I}_1$ and $\mathcal{I}_2$, individuals to mate.
repeat
$\boldsymbol{P} \leftarrow$ `random_hyperplane`$(\mathcal{I}_1 \cup \mathcal{I}_2)$.
$\xi_1^{(1)}, \xi_2^{(1)} \leftarrow$ `split_individual`$(\mathcal{I}_1, \boldsymbol{P})$.
$\xi_1^{(2)}, \xi_2^{(2)} \leftarrow$ `split_individual`$(\mathcal{I}_2, \boldsymbol{P})$.
until $\xi_k^{(i)} \neq \varnothing, \forall i, k$
$\mathcal{O}_1 = \xi_1^{(1)} \cup \xi_2^{(2)}$; $\mathcal{O}_1.\ell = \mathcal{I}_1.\ell$;
$\mathcal{O}_2 = \xi_1^{(2)} \cup \xi_2^{(1)}$; $\mathcal{O}_2.\ell = \mathcal{I}_2.\ell$.
return $\mathcal{O}_1, \mathcal{O}_2$, offspring.

(a) Crossover of two Voronoi diagrams, applied with probability p_c.



(b) Example of crossover of two Voronoi genotype individuals in 2D.

Fig. 2: Crossover operator for Voronoi diagrams.

Mutation Operator Several mutation operators can be designed for such a variable-length representation.

- At the individual level, a Voronoi site can be added, at a randomly chosen position, with a random label; or a randomly chosen Voronoi site can be removed.
- At the site level, Voronoi sites can be moved around in the space – and the well-known self-adaptive Gaussian mutation has been chosen here, inspired by Evolution Strategies (see (1) below); or the label of a Voronoi site can be changed.

In the self-adaptive Gaussian mutation [11], each coordinate x of each Voronoi site also "carries" its own variance σ that is used for its Gaussian mutation. Coordinate x undergoes Gaussian mutation with variance σ while σ undergoes a log-normal mutation with learning rate η as follows:

$$x \leftarrow \sigma\mathcal{N}(x, 1) \text{ and } \sigma \leftarrow \sigma e^{\eta\mathcal{N}0,1} \tag{1}$$

The different mutation operators are applied according to different probabilities, following the procedure described in Figure 1.

Crossover Operator The crossover operator for Voronoi representation should not simply exchange some Voronoi sites between both parents, but should respect the locality of the representation. Voronoi sites that are close to each other should have more chance to stay together than Voronoi sites that are far apart. This is achieved by the geometric crossover that operates on two (randomly selected) parents by creating a random cutting hyperplane, and exchanges the Voronoi sites from both sides of the hyperplane. The Voronoi diagrams are of course reconstructed after the crossover. This procedure is described in detail in Figure 2a. A two-dimensional example is given in Figure 2b.

3.2 Objectives and fitness assignment

Anomaly detection can be posed as a particular case of classification problem where data items must be tagged either as "normal" or "anomalous". That relying on a dataset $\Psi = \left\{\boldsymbol{x}^{(i)}, y^{(i)}\right\}$ where, without loss of generality we can state that $\boldsymbol{x} \in \mathbb{R}^n$ and $y^{(i)} \in \{\text{normal}; \text{anomaly}\}$ obtain a classifier that correctly detects instances that correspond to each of the two categories. Because of this fact the existing metrics devised to assess the quality of a classification algorithm are also applicable in this context. For this particular problem, the most relevant metrics are accuracy, recall and specificity, although many more could also be of use. Accuracy seems the best choice in the general case, as one wants to correctly identify all examples. But when dealing with anomalies, the dataset is generally highly imbalanced, as normally there are fewer anomalous instances than 'normal' ones. If only the classification accuracy is used, the error contribution of the anomalies will be reduced and hence the model will be biased to not regard them.

Furthermore, as already mentioned, the anomaly detection problem requires that the classifier is not only able to correctly classify the "normal" and "anomalous" instances present in the training dataset but is also capable of detecting when a given input falls in an area that was not covered by data of the training set and, therefore, also can be interpreted as an anomaly.

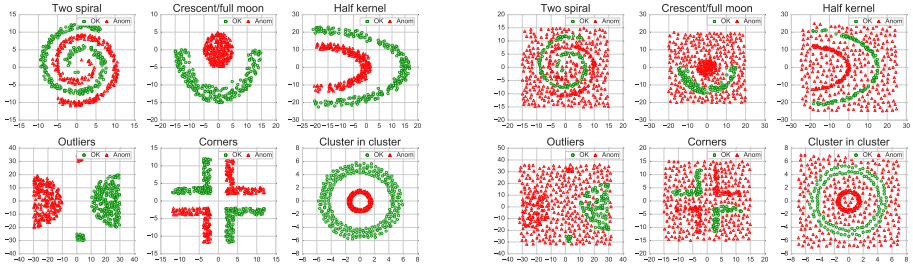
It is possible to prompt the Voronoi diagrams (individuals) to represent the known data in a form as compact as possible by expressing that as the relation between the volumes of the Voronoi cell and the convex hull of the training data that it contains. Let $\mathcal{I} = \{S_i, i = 1 \dots n_{\mathcal{I}}\}$ be a Voronoi diagram, and, for each cell C_i, let $v_i \in \mathbb{R}$ be its volume and $\mathcal{D}_i$ the set of data points it contains, i.e., $\mathcal{D}_i = \{\boldsymbol{x} \in \Psi; d(x, S_i) \leq d(x, S_j) \forall i \neq j\}$, d being the n-dimensional Euclidian distance. We can then define the individual compactness as the sum, for each cell, of the ratio of the volume of the convex hulls of $\mathcal{D}_i$ and the volume of the cell,

$$\mathrm{C}(\mathcal{I}) = \begin{cases} \sum_i \frac{\text{volume}(\text{convex\_hull}(\mathcal{D}_i))}{v_i} & \text{if } |\mathcal{D}_i| > n, \\ 0 & \text{in other case.} \end{cases} \tag{2}$$

It could be hypothesized that the previous formulation can be improved by adding a multiplicative term that counts the number of elements in $\mathcal{D}_i$, resulting in the multiplicative compactness

$$\mathrm{C}_{\text{mult}}(\mathcal{I}) = \begin{cases} \sum_i (|\mathcal{D}_i| - n) \frac{\text{volume}(\text{convex\_hull}(\mathcal{D}_i))}{v_i} & \text{if } |\mathcal{D}_i| > n, \\ 0 & \text{in other case.} \end{cases} \tag{3}$$

In both cases, maximizing the compactness will produce cells that contain the data in a form as tight as possible. Those compactness objectives can be complemented by one that promotes the existence of empty cells that represent areas of the input domain that are now present in the training data. Such objective would take care of sites with small $\mathcal{D}_i$'s and promote that they become empty as the evolution takes place. A form of representing this is by computing the



(a) Training datasets. (b) Test datasets.

Fig. 3: Training and testing datasets. Test set anomalies present in the test datasets are generated using the procedure described in Section 4.

total volume of cells with an anomaly label of an individual and rate it by the number of elements it contains,

$$\mathrm{EV}(\mathcal{I}) = \sum_{i,S_i.\ell=\text{anomaly}} \frac{v_i}{1+2\ln(|\mathcal{D}_i|+1)}\,. \tag{4}$$

Consequently, it is obvious that it is necessary to jointly address all of those objectives. Therefore, a multi-objective optimization approach will empower the algorithm with the capacity to address all the requirements of the task at the same time.

3.3 Algorithm description

VorEAl consolidates the previous components as an algorithm that constructs a classification model. The algorithm starts by creating an initial random population $\mathcal{P}_0$ of n_{pop} individuals. At a given iteration t, individuals in the population $\mathcal{P}_t$ are then mutated and mated using operators described above and thus producing an offspring population $\mathcal{P}_{\text{off}}$ that consists of n_{off} individuals. At this point, individuals that have not yet been evaluated are presented with the dataset and the values of the different objective functions are calculated. In this work, we compute the accuracy, recall and specificity, but it should be noted that others are available. From the union of $\mathcal{P}_t$ and $\mathcal{P}_{\text{off}}$, the best n_{pop} are selected using the non-dominated sorting selection of NSGA-II [3].

This process repeats until the stopping criterion of the algorithm is met. When that happens, the algorithm has a final population $\mathcal{P}_{\text{final}}$ from which the best individual(s) can be selected to represent the 'self' of the AIS. This a non-trivial task as it implies taking into account the different conflictive objectives. In this work, we select a committee of individuals $\mathcal{P}_{\text{committee}} \subseteq \mathcal{P}_{\text{final}}$ that contains the ρ-percent of $\mathcal{P}_{\text{final}}$ with the highest accuracy. Hence, the classifier returns the most voted decision among the members of $\mathcal{P}_{\text{committee}}$.

4 Experimental study

The previous discussion and proposal must be complemented by a set of experiments that establish the validity of VorEAl and studies the impacts of the different components presented. That is the focus of this section.

One of the main questions regarding VorEAl is at what point a multi-objective affinity function would actually generate better results at an admissible cost. It could be argued that there exists the possibility that adding more objectives would just make the search process more complex and resource demanding.

An important matter to be clarified was the impact of each of the objectives presented in previous section. For that reason different combinations were tested. In particular, we tested accuracy and compactness (a/c); accuracy, compactness and total empty volume (a/c/t); accuracy and multiplicative compactness (a/m) and accuracy, multiplicative compactness and total empty volume (a/m/t).

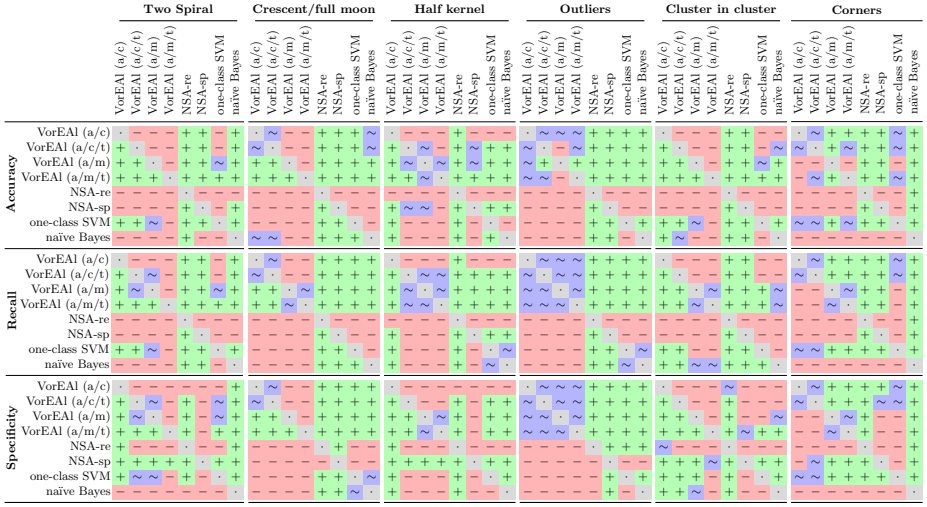
In order to provide grounds for comparison with similar approaches as well as well-known approaches, other methods were included in the experiments. In particular, we included the negative selection algorithm (NSA) [6] using both variable-sized hyper-spheres and hyper-rectangles. For fair comparisons, we applied the NSA^{+}_{sp} and NSA^{+}_{re} in which non-self training samples are subsequently used to enrich the detector library generated by NSA.

Similarly, we have included in the experiments two well-known classifiers: one-class vector machines (SVMs) [12] and the naïve Bayes classifier.

The experiments involved six classification benchmarks problems: the 'two spiral', 'crescent and full moon', 'half densities', 'corners', 'outliers' and 'cluster in cluster' problems. They have the advantage that they can be visualized in 2D while still posing a substantial challenge to the algorithm. One key element that must be addressed is the ability of the method to detect anomalies that were present in the original dataset and also those that were not present. Six tests were prepared with that goal in mind by adding random anomaly data in the areas that did not had any data in the training dataset. The resulting training and test datasets can be observed in Figure 3. Besides fixing these parameters we limited the population to 100 individuals and ran the algorithms for 500 generations. The rest of the parameters are tuned using a grid search procedure on a reduced-size problem. The same parameters were used for all problems. The mutation of the parameters were $p_{\text{s}} = 0.5$, $p_{\text{f}} = 0.5$, $p_{\text{t}} = 0.1$, $p_{+} = 0.2$, $p_{-} = 0.1$ and $\eta = 0.5$, while the mating probability was 0.5, the minimum and maximum number of sites in an individual was set to 20 and 100, respectively and the committee selection percentile (ρ) was set to 5% of the population.

The stochastic nature of the algorithms being analyzed calls for the use of an experimental methodology that relies on statistical hypothesis tests. Using those tests, we are able to determine in a statistical sound way if one algorithm instance outperforms another. The topic of assessing stochastic classification algorithms is studied in depth in [4]. There, it is shown that the Bergmann–Hommel procedure is the most suitable for our class of problem. In all cases, we have used a base level of significance of 0.05 and we run the same experiment instances 50 times. The results of this experiments are shown as box plots in Figure 4. It can be inferred

Table 1: Summary of the outcome of the statistical hypothesis tests for each problem and performance indicator. When an algorithm in the row has been significantly better than the one in the column the corresponding cell is marked with a "+". If it has been worst then the cell contains a "−". Cases where no significant difference was established are identified with a "∼".



from those plots that the three-objective form of with accuracy, multiplicative compactness, and total empty volume VorEAl yielded the best results.

When many tests are carried out, a comprehensive analysis of the results is rather difficult as it implies cross-examining and comparing the results presented separately. Consequently, we present them in summarized form in Table 1. It should be noted that experiment parameters and results are available online at `http://lmarti.com/VorEAl`. To further simplify the understanding of results, why we decided to adopt a more integrative representation like the one proposed in [1]. Figure 5 summarizes the outcome of the hypothesis tests by grouping them by metric and problem, as explained in the previous section. Here it is clearly visible how VorEAl with multiplicative compactness and total empty volume objectives is generally able to yield better results. Finally, as an illustrative example, we show in Figure 5c an example of an evolved Voronoi diagram.

5 Discussion and conclusion

In this paper we have presented VorEAl, a multi-objective evolutionary algorithm that relies on Voronoi diagrams for representation. VorEAl has been devised with the problem of anomaly detection in mind. The experimental results obtained as part of this work point out that this is a promising direction of work. However, there are many areas that should be further studied and ex-

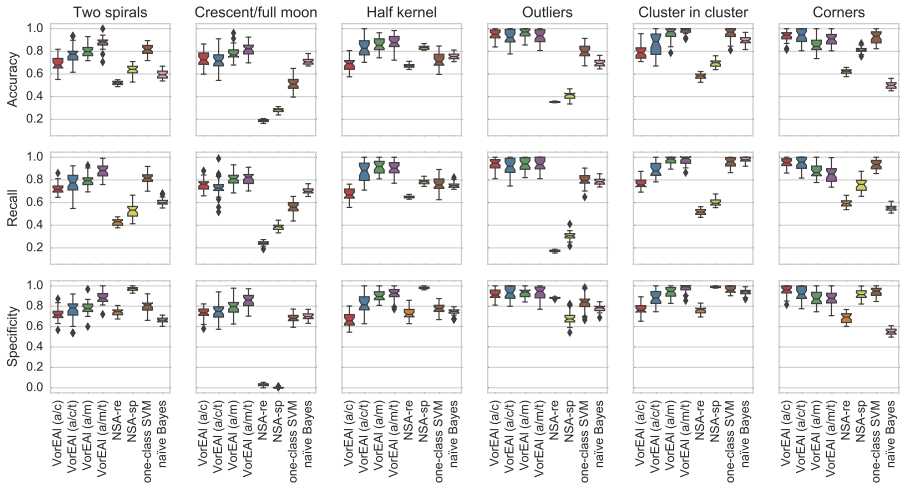


Fig. 4: Box plots of the experimental evaluations on the anomaly detection test sets.

plored. From an algorithmic point of view, we should explore other classification objectives (metrics).

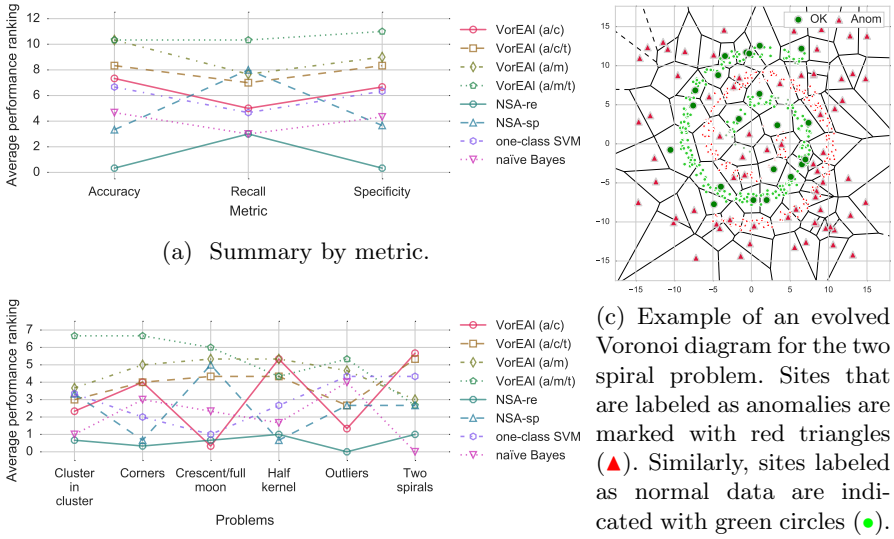
It is important to try other multi-objective fitness assignments, like those based on multi-objective performance indicators or reference points. This last approach is of particular interest as, as we already mentioned, in our case we have an *a priori* known ideal solution that can be used to guide the search. In parallel, work should the done in understanding and reducing the computational complexity of the algorithm. In this direction, we are already working on creating approximative versions of the volume meant to decrease the computational cost of the computation of the objective functions.

Acknowledgements

This work has been funded by the project PIA-FSN-P3344-146479. Authors wish to thank the reviewers for their fruitful comments.

References

1. Bader, J.: Hypervolume-Based Search for Multiobjective Optimization: Theory and Methods. Ph.D. thesis, ETH Zurich, Switzerland (2010)
2. Chandola, V., Banerjee, A., Kumar, V.: Anomaly detection: A survey. ACM Computing Surveys (CSUR) 41(3), 15 (2009)
3. Deb, K., Pratap, A., Agarwal, S., Meyarivan, T.: A Fast and Elitist Multiobjective Genetic Algorithm: NSGA–II. IEEE Transactions on Evolutionary Computation 6(2), 182–197 (April 2002)



(a) Summary by metric.

(b) Summary by problem.

(c) Example of an evolved Voronoi diagram for the two spiral problem. Sites that are labeled as anomalies are marked with red triangles (▲). Similarly, sites labeled as normal data are indicated with green circles (●).

Fig. 5: Summaries of the statistical tests and an illustrative example.

4. García, S., Herrera, F.: An extension on "statistical comparisons of classifiers over multiple data sets" for all pairwise comparisons. Journal of Machine Learning Research 9, 2677–2694 (2008)
5. Hamda, H., Jouve, F., Lutton, E., Schoenauer, M., Sebag, M.: Compact unstructured representations for evolutionary design. Applied Intelligence 16, 139–155 (2002)
6. Ji, Z., Dasgupta, D.: Real-valued negative selection algorithm with variable-sized detectors. Lect Notes Comput Sc 3102, 287–298 (2004)
7. Kim, J., Bentley, P.J., Aickelin, U., Greensmith, J., Tedesco, G., Twycross, J.P.: Immune system approaches to intrusion detection – a review. Natural Computing 6(4), 413–466 (2007)
8. Martí, L., Fansi-Tchango, A., Navarro, L., Schoenauer, M.: VorAIS: A multi-objective Voronoi diagram-based artificial immune system. In: Proceedings of the 11th Annual conference on Genetic and evolutionary computation (GECCO'09). ACM (2016)
9. Northcutt, S., Novak, J.: Network intrusion detection. Sams Publishing (2002)
10. Schoenauer, M.: Shape representation for evolutionary optimization and identification in structural mechanics. In: Winter, G., Périaux, J., Galán, M., Cuesta, P. (eds.) Genetic Algorithms in Engineering and Computer Science (EUROGEN95). pp. 443–464 (1995)
11. Schwefel, H.P.: Numerical Optimization of Computer Models. John Wiley & Sons, New-York (1981), 1995 – 2^{nd} edition
12. Tax, D.M.J., Duin, R.P.W.: Support vector data description. Machine learning 54(1), 45–66 (2004)