A New Reduced-Length Genetic Representation for Evolutionary Multiobjective Clustering

Mario Garza-Fabre[1(✉)], Julia Handl[1], and Joshua Knowles[2]

[1] Decision and Cognitive Sciences Research Centre, University of Manchester, Manchester M15 6PB, UK
{mario.garza-fabre,julia.handl}@manchester.ac.uk
[2] School of Computer Science, University of Birmingham, Birmingham B15 2TT, UK
j.knowles@cs.bham.ac.uk

Abstract. The last decade has seen a growing body of research illustrating the advantages of the evolutionary multiobjective approach to data clustering. The scalability of such an approach, however, is a topic which merits more attention given the unprecedented volumes of data generated nowadays. This paper proposes a reduced-length representation for evolutionary multiobjective clustering. The new encoding explicitly prunes the solution space and allows the search method to focus on its most promising regions. Moreover, it allows us to precompute information in order to alleviate the computational overhead caused by the processing of candidate individuals during optimisation. We investigate the suitability of this proposal in the context of a representative algorithm from the literature: MOCK. Our results indicate that the new reduced-length representation significantly improves the effectiveness and computational efficiency of MOCK specifically, and can be seen as a further step towards a better scalability of evolutionary multiobjective clustering in general.

Keywords: Data clustering · Evolutionary multiobjective optimisation · Genetic representation · Scalability

1 Introduction

Data clustering is the unsupervised task concerned with classifying a collection of data points, based on some notion of similarity, into a finite number of disjoint subsets called *clusters* [10]. This task is usually modelled as an optimisation problem (tackled *e.g.* by evolutionary algorithms), relying on the use of a clustering quality criterion (or *validity index* [5]) in order to guide the search process [8]. It is often the case, however, that a single criterion is unable to capture all desirable aspects of a clustering solution; this, together with the fact that we generally have little (or poor) information about how to choose k so it corresponds to the number of true natural clusters in the data (or number of classes), renders the conceptual advantages of *evolutionary multiobjective clustering* (EMC) evident [6,13]. EMC approaches are capable of producing, in a single run, a *Pareto front approximation* (PFA) of candidate partitions yielding

H. Trautmann et al. (Eds.): EMO 2017, LNCS 10173, pp. 236–251, 2017.
DOI: 10.1007/978-3-319-54157-0_17

different trade-offs between multiple clustering criteria and potentially covering a wide range of values of k. Intuitively, the EMC methods need to be equipped with appropriate representations and operators which facilitate this outcome.

In the literature, there exists a variety of representations which have been proposed for the data clustering problem. These approaches range from those directly encoding cluster memberships of all data points, or the interaction (shared cluster membership) between points, to the increasingly popular prototype-based strategies encoding cluster centres or representatives. The selection of an appropriate encoding scheme can be seen as a multiobjective problem itself, as usually the approach excelling in one aspect presents weaknesses in some other aspects. The suitability of a representation can be judged in terms of its degree of problematic *non-synonymous redundancy* [16], its capacity to capture arbitrary cluster shapes, or its scalability properties generally related to the length of the genotype (and hence the size of the search pace) which may depend on problem size, dimensionality, and/or number of clusters. Some approaches are also better suited than others for encoding partitions with a non-fixed k, which is essential if this (domain-specific) information is unavailable *a priori*, as discussed above. Finally, some representations (*e.g.* those involving variable-length genotypes) introduce additional complexity and demand a more careful design of the genetic operators. The reader is referred to [8,13] for a review and discussion of problem representations used in evolutionary data clustering.

We introduce a new representation which attempts to provide a better trade-off between the aforementioned aspects in comparison to existing approaches from the literature. The proposed encoding scheme is based on the *locus-based adjacency representation* originally reported in [14]. This graph-based representation (described in Sect. 2.1) is used by one of the most representative methods from the state-of-the-art in EMC: the *multiobjective clustering with automatic k-determination* (MOCK) algorithm [6]. Strengths of this representation include the straightforward definition of meaningful genetic operators, and the ability to *naturally* encode partitions of varying k (removing the need of predefining this parameter). Moreover, this representation has been found to be less non-synonymously redundant than other approaches [7]. Nevertheless, it has also been criticized because its encoding length corresponds to the number of points in the data set. Due to specific strategies adopted during initialisation and genetic variation (discussed later in Sect. 2.3), the resulting size of the search space has not represented a major bottleneck with respect to the scalability of MOCK (as is evident from MOCK's existing performance on data sets with up to ∼4,000 points [6]). The use of the reduced-length representation proposed in this paper, however, can further improve the scalability of MOCK, providing clear benefits in terms of both clustering performance and computational efficiency.

Our new representation preserves the conceptual advantages of the locus-based adjacency representation, but can significantly reduce the length of the genotype and explicitly prune uninteresting regions of the solution space. This is achieved by exploiting relevant information from the *minimum spanning tree* (MST). Furthermore, the new representation enables the incremental processing

and evaluation of candidate solutions starting from an initially precomputed state, thus decreasing the computational burden. We implemented this reduced-length representation within the framework of MOCK, which allows us to directly assess the suitability of this proposal with respect to the method's original full-length representation. It is important to remark that such an assessment focuses only on the ability of the reduced-length representation to improve MOCK's performance and efficiency during its *clustering phase*; analysis of the subsequent *model-selection phase* is therefore beyond the scope of this study.[1] Also, the version of MOCK considered here, particularly its optimisation criteria and initialisation scheme, assumes that input data elements are represented by vectors of numerical attributes; scenarios where data can be described by non-numerical attributes, or where only dissimilarity (or similarity) data describing the relationships between elements is available, are not covered by this paper.

This paper proceeds as follows. First, the new reduced-length encoding is described in detail in Sect. 2. Also, this section briefly discusses our implementation of MOCK, which has been adapted in this study to take full advantage of the new representation scheme. Section 3 presents our experimental evaluation and discusses the results obtained. Finally, Sect. 4 summarises the main findings of this study and highlights potential directions for future research.

2 The New Reduced-Length Genetic Representation

This section introduces a new reduced-length representation which seeks to improve the scalability of EMC. The proposed representation is incorporated into the MOCK algorithm in order to investigate its advantages with respect to the method's original (full-length) encoding. Besides equipping MOCK with the new representation, additional changes to the optimisation criteria and search strategy allow us to, respectively, take full advantage of this proposal and of MOCK's specialised initialisation routine.[2] The reduced-length representation and the corresponding adaptation of MOCK are described in Sects. 2.1 and 2.2, respectively. Then, Sect. 2.3 discusses how the new encoding scheme impacts on the size of the solution space which is accessible to the search method.

2.1 Reduced-Length Representation

The new genetic encoding draws from the locus-based adjacency representation originally used by MOCK [14]. As shown in Fig. 1, data points are seen as the

[1] Whereas the clustering phase is responsible for generating a PFA comprising high-quality partitions, the model-selection phase is concerned with selecting and reporting one (or more) candidate partition(s) from this PFA as the problem's solution.

[2] It should be noted that changing the optimisation criteria is the only adaptation of MOCK required by the representation scheme proposed in this paper. Such an adaptation, however, is only intended to exploit the advantages that the new representation can provide in terms of computational efficiency; this change is not found to affect MOCK's behaviour and performance as discussed at the end of Sect. 3.2.

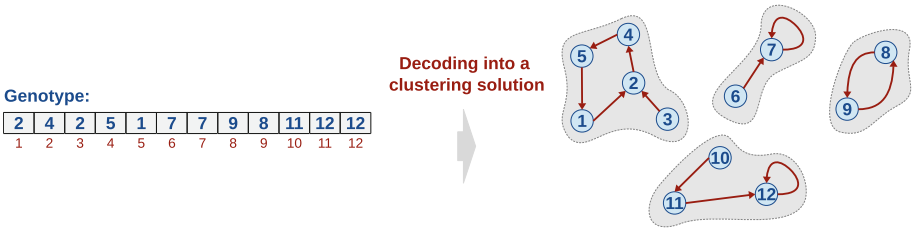

Fig. 1. Locus-based adjacency representation. A data set of size $N = 12$ is considered. A gene $x_i = j$ in the genotype denotes a link $i \rightarrow j$ from datum i to another datum j, $i, j \in \{1, \ldots, N\}$. Each connected component in the resulting graph is seen as a different cluster. Thus, the genotype of this example encodes a partition with $k = 4$ clusters.

nodes of a graph and the genotype of an individual defines the links between them. These links result in a set of connected components which represents a candidate partition. Despite presenting clear advantages with respect to other existing representations [7], the length of the genotype, given by the size of the problem N, can be seen as the main scalability issue of this approach.

As illustrated in Figs. 2 and 3, the reduced-length representation predefines a potentially large subset of the links based on the MST in order to (explicitly) limit exploration to the most promising regions of the search space. This requires identifying the subsets of relevant and fixed links from the MST, respectively denoted Γ and Δ. Classification of the MST links depends on their *degree of interestingness* (DI) and the setting of parameter δ. As can be seen, the fixed set Δ, consisting of (roughly) the $\delta\%$ less interesting MST links, defines a partial clustering which serves as the basis for the generation of all candidate solutions during the search process. In this way, the optimisation problem is redefined as that of determining only the links not yet defined in the partial clustering solution; such missing pieces of information relate to the relevant links in set Γ and are encoded in the $|\Gamma|$-length genotype of the new representation.[3]

By fixing all non-relevant MST links, only the removal (or replacement) of the relevant MST links is considered. We want to partition the MST links into relevant and non-relevant links by some criterion. We propose here the use of the criterion DI, which was originally introduced in [6] as a means to guide part of the initialisation routine of MOCK (refer to Sect. 2.2 for details). In the ideal case the relevant links would be those whose removal leads to a separation of the MST which is consistent with the inherent cluster structure, but that is not possible to know *a priori*. However, DI seems to do a good job; specifically, the DI approach has been found to be less biased, in comparison to directly using the dissimilarity (distance) between data points, towards classifying as relevant those links connecting outliers. As defined in [6], the DI of a link $i \rightarrow j$ is given

[3] Notice that when parameter δ is set to $\delta = 0$, the encoding scheme proposed here is equivalent to the original (full-length) locus-based adjacency representation.

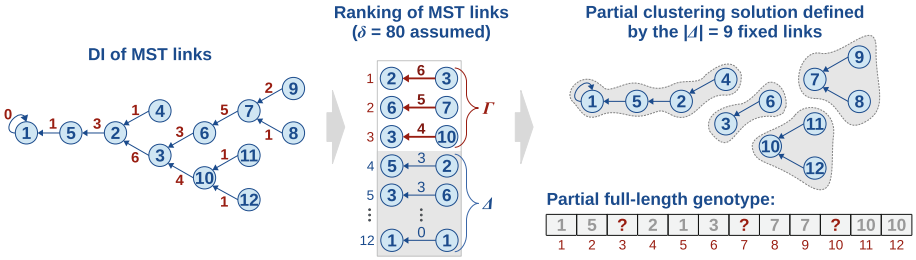

Fig. 2. Classifying MST links based on their degree of interestingness (DI) and the user-defined parameter δ ($0 \leq \delta \leq 100$). Whereas set Γ is formed by the $|\Gamma| = \lceil \frac{(100-\delta)}{100} N \rceil$ most prominent (highest DI) links, set Δ consists of the $|\Delta| = \lfloor \frac{\delta}{100} N \rfloor$ links with the lowest DI. A value of $\delta = 80$ is used in this example, which produces $|\Gamma| = 3$ and $|\Delta| = 9$. The nine links in Δ (and their corresponding genes in the full-length genotype) are assumed fixed and lead to a partial clustering (with $k = 4$ clusters) which forms the basis for all candidate solutions to be explored during the search process.

by $int(i \rightarrow j) = \min\{nn_i(j), nn_j(i)\}$, where $nn_a(b)$ refers to the ranking position of data point b in the list of nearest neighbours of data point a.

2.2 Adaptation of MOCK

Below, the specifics of MOCK's implementation used during the experiments of this study are described, with particular emphasis on the components which vary with respect to the original version of the algorithm reported in [6].

Optimisation Criteria. MOCK, as reported in [6], optimises two complementary clustering criteria: *overall deviation* (ODV) and *connectivity* (CNN). With the aim of exploiting the benefits that the new encoding can provide in terms of the incremental processing of solutions (see Delta-Evaluation below), ODV is replaced here with a highly correlated criterion: *intracluster variance* (VAR). VAR accounts for cluster compactness (homogeneity) and is given by:

$$var(\mathcal{C}) = \frac{1}{N} \sum_{c \in \mathcal{C}} v(c) \ , \tag{1}$$

where $\mathcal{C}$ is the set of clusters in the candidate partition and $v(c)$ represents the individual contribution of cluster c to this measure: $v(c) = \sum_{i \in c} \sigma(i, \mu_c)^2$. Here, μ_c denotes the centroid of cluster c, and $\sigma(i, \mu_c)$ refers to the dissimilarity between data point i and μ_c (the Euclidean distance is used in this study).

The CNN criterion is preserved as in the original implementation of MOCK. This criterion captures cluster connectedness, reflecting the degree to which neighbouring data points are identified as members of the same cluster:

$$cnn(\mathcal{C}) = \sum_{i=1}^{N} \sum_{l=1}^{L} \rho(i, l) \ , \tag{2}$$

where L specifies the size of the neighbourhood and $\rho(i, l) = \frac{1}{l}$ iff point i and its l-th nearest neighbour are not in the same cluster and $\rho(i, l) = 0$ otherwise.

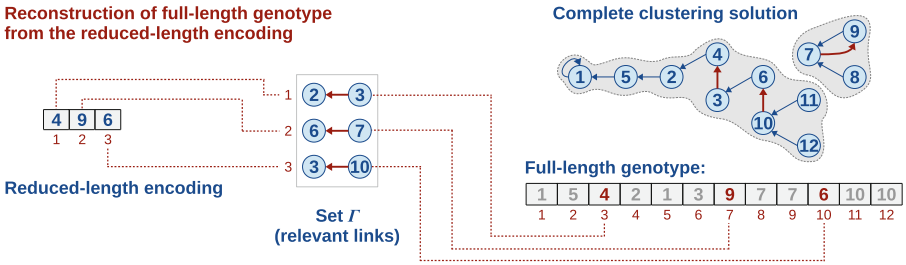

Fig. 3. The new reduced-length genetic representation and the process of reconstructing the full-length genotype which is a first step towards decoding. The new representation operates with genotypes of length $|\Gamma|$, where Γ is the set of relevant MST links (see Fig. 2). Starting from the partial solution given by the set of fixed MST links Δ, the $|\Gamma|$-length encoding is used to define the missing pieces of information in the full-length genotype, which is then decoded into a complete clustering solution.

Both VAR and CNN are to be minimised. While the optimisation of VAR tends to increase k, CNN presents the opposite tendency. Therefore, the simultaneous optimisation of VAR and CNN compensates for these individual biases and produces a PFA of good-quality partitions with a diversity of values for k.

Delta-Evaluation. Given the partial clustering derived from the set of fixed MST links Δ, the decoding and evaluation of a candidate solution only requires processing the non-fixed information encoded in its $|\Gamma|$-length genotype (Figs. 2 and 3). Thus, we can precompute the decoding and evaluation of such a partial solution in order to speed up the processing of individuals during the search.

The decoding of the $|\Gamma|$-length genotype of an individual creates new links which can merge originally separate clusters of the partial solution. Such a change in the phenotype implies an amendment to the initially precomputed values of the VAR and CNN criteria. Adhering to its original definition provided in (1), VAR is recomputed by averaging the individual contributions of all the final clusters to this measure. In this case, however, the contribution of a new joint cluster $c_h = c_i \cup c_j$ to VAR can be more efficiently obtained by leveraging on the original (precomputed) contributions of the clusters being combined [1]:

$$v(c_h) = v(c_i) + v(c_j) + |c_i| \times \sigma(\mu_{c_i}, \mu_{c_h})^2 + |c_j| \times \sigma(\mu_{c_j}, \mu_{c_h})^2 \ , \qquad (3)$$

where μ_{c_h} denotes the centroid of c_h and is computed as the weighted average of the original centroids μ_{c_i} and μ_{c_j} (this is generalisable to the case where an arbitrary number of clusters are combined). Similarly, adjusting CNN due to the combination of two clusters c_i and c_j requires subtracting all contributions made to this measure as a consequence of the original separation of c_i and c_j.

Search Engine. MOCK's original version is based on the *Pareto envelope-based selection algorithm version 2* (PESA-II) [2]. PESA-II is strongly elitist,

and this causes that part of the individuals generated during initialisation (the dominated ones) are filtered and discarded without being considered during the search process. An approach with less selection pressure seems advantageous in this scenario where highly optimised solutions are generated by MOCK's specialised initialisation. We study a different implementation of MOCK based on the *nondominated sorting genetic algorithm version 2* (NSGA-II) [3]. This change in the search engine makes it possible to exploit all of the genetic material introduced during initialisation, as it forms the basis of the initial population.

Initialisation and Genetic Operators. MOCK's implementation studied herein preserves the specialised initialisation routine and the same set of genetic operators as reported in [6]. Initialisation attempts to provide MOCK with a close initial approximation to the Pareto front. An initial population of MST-derived solutions is generated following a two-phase process. The first phase constructs (at most) half of the population. Each individual resulting from this phase encodes a partition created by removing the n highest DI links from the MST (see definition of DI in Sect. 2.1). To obtain a diverse set of initial partitions, n is chosen uniformly and without replacement from the set $\{0, 1, \ldots, \min(k_{user} - 1, I)\}$. Here, k_{user} can be seen as an upper bound on the number of clusters expected; based on preliminary testing this parameter is set to $k_{user} = 2k^*$ in this study, where k^* denotes the real (or estimated) number of clusters in the data set. I is the cardinality of the subset of MST links which are allowed to be removed during this phase; these links, called the *interesting links* in [6], are those links $i \rightarrow j$ such that neither i nor j is one of the L nearest neighbours of the other. Therefore, although this phase attempts to create half of the initial population, the actual number of individuals produced also depends on k_{user} and I. The second phase generates the remainder (at least half) of the population. Every remaining individual is created by first running (a single execution of) k-means [12] for a given target k, and then removing all MST links crossing the cluster boundaries defined by the partition obtained. Each time the target k value used for k-means is drawn uniformly without replacement from the set $\{2, 3, \ldots, k_{user}\}$, which contributes to the diversity of the initial population. Note that when using the reduced-length encoding proposed in this paper, removal of MST links (in both phases defined above) is only permitted for links in Γ (links in set Δ are fixed for all candidate partitions, see Sect. 2.1).

For the genetic operators, we use *uniform crossover* [17] which can produce any possible combination of genetic material from the parent genotypes being recombined. Also, we use the *neighbourhood-biased mutation* [6] which defines individual mutation probabilities for each gene in the genotype based on the specific link it encodes. More precisely, the mutation probability of a gene $x_i = j$, encoding link $i \rightarrow j$, is given by $p_m(x_i) = 1/N + (nn_i(j)/N)^2$. This increases the chances of discarding unfavourable links. Note that the recombination and mutation strategies operate on the $|\Gamma|$-length genotypes of the new encoding.

During initialisation and mutation, any link $i \rightarrow j$ which is removed is replaced either with a self-connecting link $i \rightarrow i$ or with a link from i to one

of its L nearest neighbours. The new link is decided uniformly at random from these $L+1$ choices, but excluding the reintroduction of the original link $i \rightarrow j$. It is worth realising that if $i \rightarrow j$ is one of the links of the MST, j is not necessarily one of i's L nearest neighbours; every gene x_i in the genotype can thus encode any of (at most) $L+2$ possible links during the evolutionary process.

2.3 Search Space Reduction

Although the (full-length) locus-based adjacency representation conceptually defines a huge search space of size N^N, MOCK's strategies for the creation of MST-derived solutions and link replacement (discussed at the end of Sect. 2.2) result in a much reduced search space whose size is bounded by $(L+2)^N$. Moreover, the use of a problem-specific initialisation routine to generate high-quality base partitions contributes to (implicitly) bias exploration in an important manner.[4]

Depending on the setting of parameter δ, the new representation can reduce significantly the length of the genotype and thus explicitly prune the search space further. Specifically, using the reduced-length genetic encoding proposed in this paper the size of the search space is at most $(L+2)^{|\Gamma|}$.

3 Experiments and Results

This section presents the findings of our experimental study which aims to investigate the suitability of the reduced-length representation proposed in this paper. The new representation is compared with respect to the use of the original full-length representation and is evaluated in terms of its impact on the behaviour and performance of the MOCK algorithm. The data sets, performance assessment measures, and settings adopted for this study are first described in Sect. 3.1. The results of our experiments are then discussed in Sect. 3.2.

3.1 Experimental Setup

A total of 280 data sets are considered for the experiments of this study. As shown in Fig. 4, these data sets present varying sizes and are organised into 28 problem configurations according to their dimensionality and number of clusters. All the data sets were generated using the ellipsoidal generator previously used during the evaluation of MOCK, which is described in detail in [6].

All the experiments of this study consider a total of 21 independent executions of MOCK for each problem instance. Results are evaluated in terms of both the PFAs obtained and clustering performance. PFAs are investigated by visualising the differences between the (first-order) *empirical attainment functions*

[4] Since the creation of new solutions relies mainly on recombination (due to the low mutation rates used), the genetic material introduced during initialisation plays a key role in delimiting the extent of the solution space that is reached by the method.

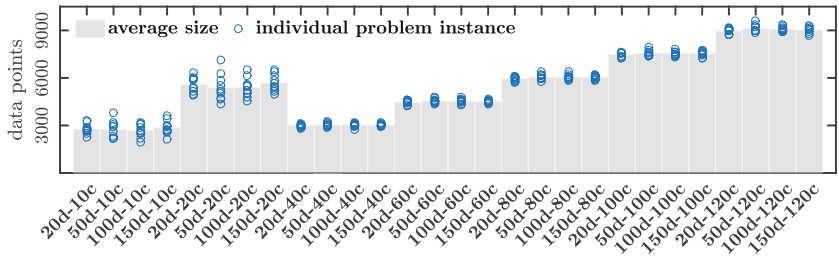

Fig. 4. Randomly generated data sets. 28 problem configurations are considered, each to be referred to as dd-kc, where d is the dimensionality and k is the number of clusters in the data set. 10 random instances were generated for each problem configuration, leading to a total of 280 data sets. The plot shows the size (N) of all individual problem instances, as well as the average size for each given problem configuration.

(EAFs) produced by the two representations [4,11]. This allows us to identify whether, and in which particular regions of the objective space, a representation performs better than the other. Plots of the EAFs were generated using the tools reported in [11]. In all the cases, objective values were normalised to the range $[0, 1]$ and, for visualisation purposes, replaced with their square roots. In addition, we use the *hypervolume indicator* (HV) [18] and the *inverted generational distance indicator with modified distance calculation* (IGD^+) [9], both computed after normalising objective values to the range $[0, 1]$. Both HV and IGD^+ capture the quality of a PFA with regard to both extent and proximity with respect to the true Pareto front. The reference point for HV was always set to $r = (1.01, 1.01)$ given the normalisation of the objective values. For IGD^+, the reference set was constructed in all the cases by merging the PFAs from all runs of all the approaches analysed and then removing the dominated vectors. Whereas HV is to be maximised, IGD^+ is to be minimised.

Clustering performance is assessed using the *Adjusted Rand Index* (ARI) measure [15]. ARI is defined in the range $[\sim 0, 1]$; the larger the value for ARI, the better the correspondence between the partition obtained and the known cluster structure of the test data set. Each run of MOCK produces a set of candidate partitions. From this set, only the best solution, according to the ARI measure, is selected and considered in the results reported in Sect. 3.2.

Finally, the settings for MOCK adopted in our experiments are as follows. Population size: $P = 100$. Number of generations: $G_{max} = 100$. Recombination probability: $p_r = 1.0$. Mutation probability: defined for each individual link, see Sect. 2.2. Neighbourhood size: $L = 10$. Initialisation parameter: $k_{user} = 2k^*$.

3.2 Results

This section investigates the advantages of the new reduced-length representation from three different perspectives: (i) quality and characteristics of the PFAs obtained; (ii) clustering performance; and (iii) computational efficiency.

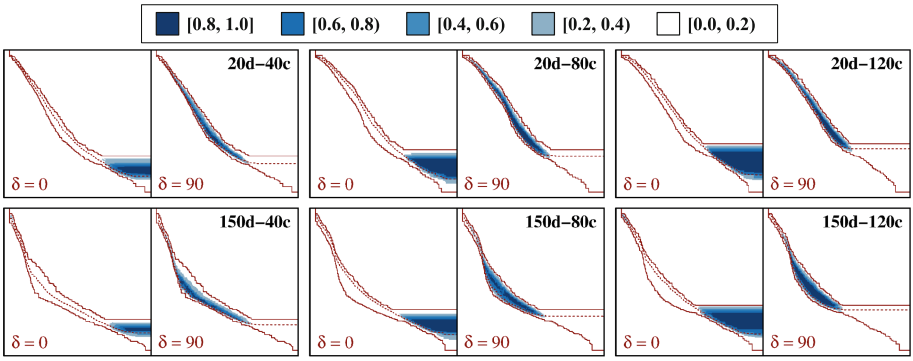

Fig. 5. Differences between the EAFs of the full-length ($\delta = 0$) and reduced-length ($\delta = 90$) representations. Results for a single instance of six problem configurations are shown. In the plots, the x-axis and y-axis denote respectively the CNN and VAR criteria. The magnitudes of the differences in the point attainment probabilities between the two settings are encoded using different intensities of blue; the darker the blue, the larger the difference. Lower and upper solid lines represent the grand best and grand worst attainment surfaces, and the dashed line denotes the median attainment surface.

First, Fig. 5 contrasts the EAFs that were computed from the PFAs obtained when using the full-length representation, *i.e.* adopting a setting of $\delta = 0$, and the reduced-length representation, with $\delta = 90$, which uses only about 10% of the original encoding length. The EAFs in Fig. 5 indicate that the full-length representation performs better with respect to the optimisation of the VAR criterion. This behaviour can be explained by the fact that VAR is relatively easy to optimise; the value of this criterion naturally decreases as the number of clusters (k) in the solution evaluated increases. When considering a large solution space (as that defined by the full-length encoding), it becomes easier for the search method to identify and exploit regions of the space favouring the optimisation of such a criterion. This behaviour is further evidenced by Fig. 6, which highlights that the full-length representation tends to produce PFAs comprising partitions with clearly higher k values (which, again, correlates with lower values of VAR) in comparison to the reduced-length encoding. The figure also illustrates that the full-length encoding obtains k values substantially exceeding the number of clusters in the real cluster structure (k^*) in most cases, including k values which can be considered far beyond practical relevance (in average, though, the k values produced tend to be close to k^*). The differences in the range of k values in the PFAs produced by the two representations are particularly evident (from Fig. 6) for problem configurations with $k^* > 40$ clusters. Consistently, for this specific subset of problem configurations (with $k^* > 40$), the PFAs of the full-length encoding cover a sufficiently larger extent of the objective space (extending widely and deeply into the uninteresting low-VAR, high-CNN regions), leading to significantly higher values for the HV indicator, see Table 1.

The reduced-length representation constrains the number of clusters a partition may involve and the range of values of VAR which can be reached by the method: ∼90% of the MST links are fixed for all phenotypes (given the use of $\delta = 90$), and the partial solution defined by such fixed links (see Fig. 2) sets the maximum k and the minimum VAR which can be seen during optimisation.[5] Note, however, that according to Fig. 6 the new representation still produces PFAs covering k values in a wide relevant range around k^*. Moreover, the reduced-length encoding allows the optimisation to be performed within a small promising area of the search space. As can be seen from Fig. 5, this results in an increased convergence ability towards the more challenging central regions of the Pareto front presenting better compromises between the VAR and CNN criteria. This enhanced convergence behaviour is reflected in significantly better scores for the IGD^+ indicator across all 28 problem configurations (Table 1). Similar results (with even more marked differences) to those of IGD^+ have been observed for the GD^+ indicator [9] which focuses exclusively on proximity to the true Pareto front (as represented by a reference set). Such results for the GD^+ indicator have not been included in this paper due to space restrictions.

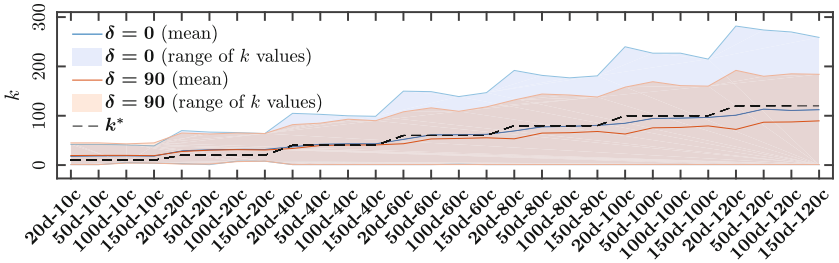

Fig. 6. Number of clusters (k) in the PFAs obtained when using two different settings for parameter δ, namely $\delta \in \{0, 90\}$. In the plot, curves show the arithmetic mean and shaded areas show the full range of k values observed for each problem configuration. A curve indicating the real number of clusters (k^*) is also shown as a reference.

From the perspective of clustering performance, Fig. 7 and Table 1 reveal that the above-discussed improvement to the convergence behaviour of MOCK translates into a better ability to discover partitions of higher quality (as captured by the ARI measure) throughout the search process. For all problem configurations, MOCK consistently reaches higher (significantly different statistically) ARI values at the end of the search process using the reduced-length representation. Two interesting behaviours are worth discussing from Fig. 7. Firstly, the use of the reduced-length encoding causes a drop in the performance of the initial population (generation 0 in the figure). This is due to the fixed nature of a large number (∼90%) of the MST links which are not available for removal by

[5] During the evolutionary process, the clusters defined by the partial solution are combined, which can only lead to the decrease of k and the increase of VAR.

Table 1. Results for the HV, IGD$^+$, and ARI performance indicators. The table contrasts the performance of two different encoding lengths, resulting from the use of two settings for the new representation: $\delta = 0$ and $\delta = 90$. A mark • indicates that the results of the latter setting are significantly different statistically to those of the former; this is investigated using the *Mann-Whitney U test*, considering a significance level of $\alpha = 0.05$ and *Bonferroni correction* of the p-values. Values for IGD$^+$ have been multiplied by 10^2 in all the cases. For ARI, the average k of the solutions selected for the measure computation (see Sect. 3.1) is indicated in parenthesis, and results for the original implementation of MOCK [6] are included as a reference. For all problem configurations, the best result scored for every performance indicator has been shaded.

	HV		**IGD$^+$** $\times 10^2$		**ARI (k)**		
Problem	$\delta = 0$	$\delta = 90$	$\delta = 0$	$\delta = 90$	$\delta = 0$	$\delta = 90$	**MOCK [6]**
20d-10c	0.932	**0.942** •	2.408	**1.197** •	0.998 (10)	**0.999 (10)** •	0.996 (10)
50d-10c	0.882	**0.899** •	3.407	**1.359** •	0.999 (10)	**1.000 (10)** •	0.998 (10)
100d-10c	0.863	**0.885** •	4.006	**1.319** •	0.999 (10)	**1.000 (10)** •	0.998 (10)
150d-10c	0.855	**0.873** •	3.838	**1.775** •	0.999 (10)	**1.000 (10)** •	0.998 (10)
20d-20c	0.958	**0.960** •	1.918	**1.257** •	0.992 (20)	**0.994 (20)** •	0.983 (20)
50d-20c	0.921	**0.930** •	3.203	**1.574** •	0.996 (20)	**0.999 (20)** •	0.990 (20)
100d-20c	0.853	**0.871** •	5.001	**2.193** •	0.997 (20)	**0.998 (20)** •	0.993 (20)
150d-20c	0.839	**0.852** •	4.903	**2.357** •	0.997 (20)	**0.999 (20)** •	0.993 (20)
20d-40c	**0.911**	0.908	3.613	**2.216** •	0.865 (47)	**0.910 (45)** •	0.806 (50)
50d-40c	0.936	**0.937**	3.101	**1.830** •	0.932 (44)	**0.960 (42)** •	0.888 (45)
100d-40c	0.941	**0.944** •	3.444	**1.773** •	0.929 (43)	**0.966 (42)** •	0.892 (45)
150d-40c	0.939	**0.943** •	3.359	**1.789** •	0.926 (43)	**0.963 (42)** •	0.885 (45)
20d-60c	**0.896**	0.889 •	4.513	**2.701** •	0.833 (76)	**0.884 (68)** •	0.781 (79)
50d-60c	**0.933**	0.924 •	3.616	**2.678** •	0.912 (68)	**0.941 (65)** •	0.874 (69)
100d-60c	**0.934**	0.926 •	3.768	**2.820** •	0.913 (68)	**0.950 (64)** •	0.867 (70)
150d-60c	**0.939**	0.931 •	3.564	**2.394** •	0.911 (68)	**0.947 (64)** •	0.865 (70)
20d-80c	**0.885**	0.874 •	5.146	**3.151** •	0.806 (108)	**0.850 (93)** •	0.751 (111)
50d-80c	**0.921**	0.910 •	4.204	**2.976** •	0.889 (95)	**0.928 (88)** •	0.847 (97)
100d-80c	**0.926**	0.912 •	4.314	**3.362** •	0.897 (93)	**0.934 (87)** •	0.851 (96)
150d-80c	**0.931**	0.917 •	4.066	**3.444** •	0.904 (91)	**0.939 (86)** •	0.861 (93)
20d-100c	**0.873**	0.864 •	5.745	**3.741** •	0.791 (139)	**0.830 (117)** •	0.743 (141)
50d-100c	**0.913**	0.900 •	4.548	**3.446** •	0.869 (125)	**0.910 (112)** •	0.822 (126)
100d-100c	**0.920**	0.902 •	4.560	**3.918** •	0.880 (120)	**0.919 (110)** •	0.825 (124)
150d-100c	**0.925**	0.907 •	4.322	**3.976** •	0.890 (118)	**0.926 (109)** •	0.834 (120)
20d-120c	**0.866**	0.858 •	6.187	**3.853** •	0.773 (180)	**0.809 (140)** •	0.722 (175)
50d-120c	**0.908**	0.892 •	4.841	**4.204** •	0.863 (152)	**0.904 (134)** •	0.815 (154)
100d-120c	**0.917**	0.898 •	4.773	**3.743** •	0.861 (150)	**0.903 (132)** •	0.806 (151)
150d-120c	**0.922**	0.900 •	4.647	**4.284** •	0.872 (146)	**0.910 (131)** •	0.799 (151)

the specialised initialisation routine (see Sect. 2.2), thus decreasing the diversity of the initially generated set of base partitions. Nevertheless, such a drop in performance is rapidly compensated by conducting a more-focused exploration in the substantially smaller solution space of the new representation. Secondly, the gradual increase in ARI as the search progresses indicates that the simultaneous optimisation of VAR and CNN implicitly and effectively leads to the optimisation of clustering quality. This provides corroborating evidence of the suitability of the clustering criteria considered as objective functions and of the conceptual advantages of the multiobjective approach to data clustering in general.

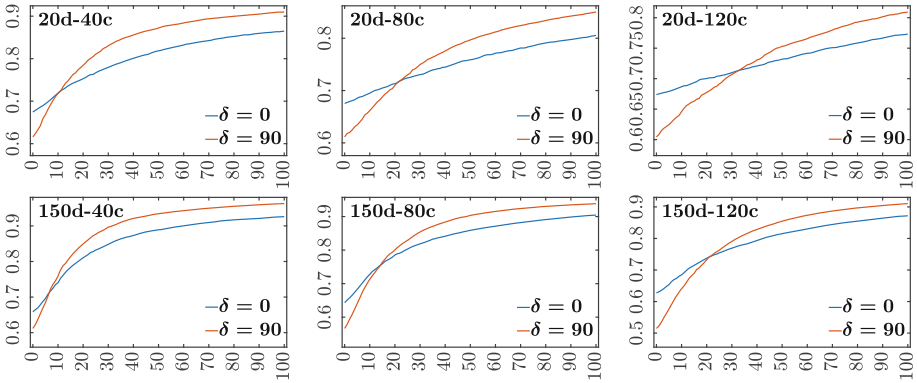

Fig. 7. Highest ARI (y-axis) in the population at every generation (x-axis) of the search process. Plots contrast the convergence behaviour using $\delta = 0$ and $\delta = 90$ for six problem configurations (average results for all instances and repetitions performed).

Figure 8 sustains that the reduced-length representation proposed in this paper delivers clear benefits in terms of computational efficiency. The use of compact (more efficient to handle) genotypes, as well as the strategy implemented for the incremental decoding and evaluation of candidate solutions, leads to a reduction of over 93% (in average) of the execution time derived from the optimisation process. Note that this excludes the computational costs involved with the generation of the initial population and those related to data loading and initial precomputations (*i.e.* distance matrix, nearest neighbours, and minimum spanning tree), since these processes are not affected by the change in representation. Such a decrease of ∼93% in the time of optimisation is found to correspond to an average decrease of ∼46% in the total execution time of MOCK.

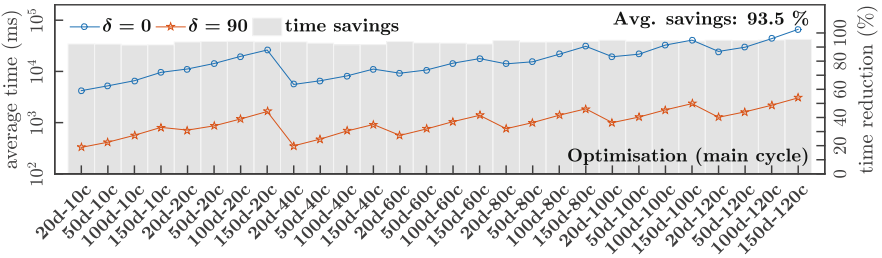

Fig. 8. Execution times scored by MOCK when using $\delta \in \{0, 90\}$. Curves show the average time (left y-axis, which is shown in logarithmic scale) for the main cycle of the optimisation strategy (discarding the time of data loading, initial computations, and generation of the initial population). Bars show the average time savings achieved by the reduced-length encoding with respect to the full-length encoding (right y-axis). The top-right corner indicates the average time savings achieved across all problems.

Finally, and despite that the analysis of other adaptations of MOCK besides the change in encoding is not the focus of this study, it is important to briefly discuss the performance differences observed with respect to the original MOCK reported in [6]. Table 1 indicates that, using a full-length encoding ($\delta = 0$), the new implementation of MOCK based on NSGA-II consistently outperforms the original implementation based on PESA-II in terms of clustering performance. This confirms the relevance of exploiting all the highly optimised solutions generated by MOCK's specialised initialisation, as discussed in Sect. 2.2. No meaningful impact on clustering performance has been observed as a consequence of replacing ODV with the VAR criterion (results not shown); this adaptation, as stated in Sect. 2, does not seek to alter the algorithm's behaviour, but is motivated by the advantages of VAR as it facilitates delta-evaluation.

4 Conclusions and Future Work

This paper studied a new reduced-length genetic representation for evolutionary multiobjective clustering. This representation exploits relevant information from the MST as a means to narrow the extent of exploration by focusing on the most prominent regions of the solution space. When implemented within the MOCK algorithm [6], the new representation scheme was found to offer significant advantages which were analysed from different perspectives.

Owing to its potential to explicitly prune large portions of the search space, the reduced-length representation allowed MOCK to produce Pareto front approximations of much greater quality in comparison to the original full-length representation. These improved convergence capabilities were found to reliably translate into a significantly increased clustering performance on a large number of test data sets. Besides these clear advantages regarding search performance and clustering quality, the new representation reported additional benefits in terms of computational efficiency. By enabling the precomputation of a substantial amount of information with the aim of expediting the processing of candidate solutions during optimisation, the new representation considerably reduced the computational overhead. Together, hence, all these findings indicate that the reduced-length representation proposed in this paper can be seen as an important step towards a better scalability of MOCK (and, in general, of evolutionary approaches to multiobjective clustering), and should impact on the method's ability to deal with larger and more challenging clustering problems.

A single fixed setting ($\delta = 90$) for the new solution representation was considered in this study, which results in the use of only about 10% of the original encoding length. Although promising results have been achieved using this setting, a range of different settings for this approach need to be investigated. More importantly, there remains the question of how far we can go in reducing the length of the encoding, and thus the size of the corresponding solution space, without sacrificing performance. Evidently, this strongly depends on the suitability of the mechanisms through which such a reduction is accomplished. The strategy adopted in this study relies on the ranking of the MST links on the basis

of their degree of interestingness (a concept previously exploited with different purposes in [6]); failure to properly discriminate between the MST links could prune and discard key regions of the search space, thus compromising effectiveness. Despite showing promise during the experiments of this paper, therefore, a thorough analysis of this strategy, as well as the exploration of other alternative strategies, is certainly a topic which deserves further investigation and will constitute one of the main directions for our future work.

References

1. Chan, T., Golub, G., LeVeque, R.: Algorithms for computing the sample variance: analysis and recommendations. Am. Stat. **37**(3), 242–247 (1983)
2. Corne, D.W., Jerram, N.R., Knowles, J.D., Oates, M.J.: PESA-II: region-based selection in evolutionary multiobjective optimization. In: Genetic and Evolutionary Computation Conference, pp. 283–290. Morgan Kaufmann Publishers, San Francisco (2001)
3. Deb, K., Pratap, A., Agarwal, S., Meyarivan, T.: A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE Trans. Evol. Comput. **6**(2), 182–197 (2002)
4. Grunert da Fonseca, V., Fonseca, C.M., Hall, A.O.: Inferential performance assessment of stochastic optimisers and the attainment function. In: Zitzler, E., Thiele, L., Deb, K., Coello Coello, C.A., Corne, D. (eds.) EMO 2001. LNCS, vol. 1993, pp. 213–225. Springer, Heidelberg (2001). doi:10.1007/3-540-44719-9_15
5. Halkidi, M., Batistakis, Y., Vazirgiannis, M.: On clustering validation techniques. J. Intell. Inf. Syst. **17**(2), 107–145 (2001)
6. Handl, J., Knowles, J.: An evolutionary approach to multiobjective clustering. IEEE Trans. Evol. Comput. **11**(1), 56–76 (2007)
7. Handl, J., Knowles, J.: An investigation of representations and operators for evolutionary data clustering with a variable number of clusters. In: Runarsson, T.P., Beyer, H.-G., Burke, E., Merelo-Guervós, J.J., Whitley, L.D., Yao, X. (eds.) PPSN 2006. LNCS, vol. 4193, pp. 839–849. Springer, Heidelberg (2006). doi:10.1007/11844297_85
8. Hruschka, E.R., Campello, R.J.G.B., Freitas, A.A., de Carvalho, A.C.P.L.F.: A survey of evolutionary algorithms for clustering. IEEE Trans. Syst. Man Cybern. Part C (Appl. Rev.) **39**(2), 133–155 (2009)
9. Ishibuchi, H., Masuda, H., Tanigaki, Y., Nojima, Y.: Modified distance calculation in generational distance and inverted generational distance. In: Gaspar-Cunha, A., Henggeler Antunes, C., Coello, C.C. (eds.) EMO 2015. LNCS, vol. 9019, pp. 110–125. Springer, Cham (2015). doi:10.1007/978-3-319-15892-1_8
10. Jain, A.K.: Data clustering: 50 years beyond K-means. Pattern Recogn. Lett. **31**(8), 651–666 (2010)
11. López-Ibáñez, M., Paquete, L., Stützle, T.: Exploratory analysis of stochastic local search algorithms in biobjective optimization. In: Bartz-Beielstein, T., Chiarandini, M., Paquete, L., Preuss, M. (eds.) Experimental Methods for the Analysis of Optimization Algorithms, pp. 209–222. Springer, Heidelberg (2010)
12. MacQueen, J.: Some methods for classification and analysis of multivariate observations. In: Berkeley Symposium on Mathematical Statistics and Probability, vol. 1, pp. 281–297. University of California Press, Berkeley (1967)
13. Mukhopadhyay, A., Maulik, U., Bandyopadhyay, S.: A survey of multiobjective evolutionary clustering. ACM Comput. Surv. **47**(4), 61:1–61:46 (2015)

14. Park, Y.J., Song, M.S.: A genetic algorithm for clustering problems. In: Genetic Programming, pp. 568–575. Morgan Kaufmann, Madison, July 1998
15. Rand, W.M.: Objective criteria for the evaluation of clustering methods. J. Am. Stat. Assoc. **66**(336), 846–850 (1971)
16. Rothlauf, F., Goldberg, D.E.: Redundant representations in evolutionary computation. Evol. Comput. **11**(4), 381–415 (2003)
17. Syswerda, G.: Uniform crossover in genetic algorithms. In: International Conference on Genetic Algorithms, pp. 2–9. Morgan Kaufmann Publishers Inc., San Francisco (1989)
18. Zitzler, E., Thiele, L., Laumanns, M., Fonseca, C.M., Grunert da Fonseca, V.: Performance assessment of multiobjective optimizers: an analysis and review. IEEE Trans. Evol. Comput. **7**(2), 117–132 (2003)