Automated Biological Sequence Description by Genetic Multiobjective Generalized Clustering

I. ZWIR,[a] R. ROMERO ZALIZ,[b] AND E.H. RUSPINI[c]

[a]*Department of Molecular Microbiology,*
Howard Hughes Medical Institute Research Laboratories,
Washington University School of Medicine, Saint Louis, Missouri 63110-1093, USA

[b]*Department of Computer Science, Facultad de Ciencias Exactas y Naturales,*
Universidad of Buenos Aires, Buenos Aires, Argentina

[c]*Artificial Intelligence Center, SRI International, Menlo Park, California, USA*

ABSTRACT: **Recent advances in the accessibility of databases containing representations of complex objects—exemplified by repositories of time-series data, information about biological macromolecules, or knowledge about metabolic pathways—have not been matched by availability of tools that facilitate the retrieval of objects of particular interest and aid understanding their structure and relations. In applications, such as the analysis of DNA sequences, on the other hand, requirements to retrieve objects on the basis of qualitative characteristics are poorly met by descriptions that emphasize precision and detail rather than structural features. This paper presents a method for identification of interesting qualitative features in biological sequences. Our approach relies on a generalized clustering methodology in which the features being sought correspond to the solutions of a multivariable, multiobjective optimization problem with features generally corresponding to fuzzy subsets of the object being represented. Foremost among the optimization objectives being considered are measures of the degree by which features resemble prototypical structures deemed to be interesting by database users. Other objectives include feature size and, in some cases, performance criteria related to domain-specific constraints. Genetic-algorithm methods are employed to solve the multiobjective optimization problem. These optimization algorithms discover candidate features as subsets of the object being described and that lie in the set of all Pareto-optimal solutions—of that problem. These candidate features are then summarized, employing again evolutionary-computation methods, and interrelated by employing domain-specific relations of interest to the end users. We present results of the application of this two-step method to the recognition and summarization of interesting features in DNA sequences of *Tripanosoma cruzi.***

KEYWORDS: **qualitative description; feature elicitation; generalized clustering; biological DNA sequences; multiobjective genetic algorithms; Pareto optimality; hierarchy of evolution programs**

Address for correspondence: I. Zwir, Department of Molecular Microbiology, Howard Hughes Medical Institute Research Laboratories, Washington University School of Medicine, St. Louis, MO 63110-1093, USA. Voice: 314-362-3691; fax: 314-747-8228.
zwir@borcim.wustl.edu

Ann. N.Y. Acad. Sci. **980: 65–82 (2002).**

INTRODUCTION

This paper presents an application of methods for the discovery of qualitative features and relations in complex objects, such as time series or large biological molecules. The motivation for the development of this methodology is provided by requirements for searching and interpreting databases containing representations of this type of object in terms that are close to the needs and experience of the users of those data-based descriptions.

In most applications of data and knowledge storage technology, however, the user must cope with representation methods, developed typically in the context of a different class of applications that hinder, rather than aid, in understanding either individual objects or the systems represented by the whole collection. As was pointed out by Zadeh,[1] most of the existing analytic techniques emphasize the processing of detailed system measurements rather than that of qualitative features of direct meaning to users (called *perceptions* by Zadeh).

This paper presents an application of generalized clustering techniques[2,3] to the discovery of qualitative features in complex biological sequences. These qualitative features include both interesting substructures and interesting relations between those structures. The notion of "interestingness" is provided by domain experts by means of abstract qualitative models of both features and relations.

We present a two-level methodology for the elicitation and summarization of qualitative features in DNA sequences of *Tripanosoma cruzi*.[4] This work is part of a general program of research that seeks, in addition to discovery and summarization techniques, the development of techniques for the annotation of complex objects and of data mining techniques that exploit the elicited qualitative representations.

We present first, generalized clustering ideas providing the basic framework for our techniques. We then deal with a description of the biological object description problem and discuss our two-level methodology, discussing first the multiobjective genetic-based clustering method for biological sequence recognition (MGCM-BSR), a generalized clustering method[2,3] for identification of interesting features. The features identified by this method—lying in the Pareto-optimal frontier of an optimization-based approach to the clustering problem—are then summarized by employing the genetic-based method for biological sequence summarization (GM-BSS), which relies on a hierarchy of evolution programs.[5] Genetic-based metrics are used to evaluate the performance of the methods. The final section presents results of the application of this method to the DNA sequence description problem.

GENERALIZED CLUSTERING

The methods presented in this paper belong to a family of techniques for the discovery of interesting structures in data sets by classification of its points into a finite number of fuzzy subsets, or *fuzzy clustering.* Fuzzy clustering methods were introduced by Ruspini[6] to provide a richer representation scheme, based on a flexible notion of partition, for the summarization of data set structure and to take advantage of the ability of continuous-analysis techniques to express and treat classification problems in a formal manner.

In Ruspini's original formulation, the clustering problem was presented as a continuous variable optimization problem over the space of fuzzy partitions of the data set. This original formulation of the clustering problem as an optimization problem has been largely retained in various extensions of the approach that differ primarily in the nature of the functionals being optimized and in the constraints that the partition must satisfy.[7]

The original approach proposed by Ruspini focused on the determination of the clustering as a whole; that is, a family of fuzzy subsets of the data set providing a disjoint, exhaustive partition of the set into interesting structures. Recent developments have emphasized the determination of individual clusters as fuzzy subsets having certain optimal properties. From this perspective, a fuzzy clustering is a collection of optimal fuzzy clusters—that is, each cluster is optimal in some sense and the partition satisfies certain conditions—rather than an optimal partition—that is, the partition, as a whole, is optimal in the sense that it minimizes some predefined functional defining classification quality. Redirecting the focus of the clustering process to the isolation of individual subsets having certain desirable properties also provides a better foundation for the direct characterization of interesting structures frees the clustering process from the requirement that clusters be disjoint and that partitions be exhaustive.

In the context of image-processing applications, for example, features may correspond to certain interesting prototypical shapes. In these applications, not every image element may belong to an interesting feature while some points might belong to more than one cluster (e.g., the intersection of two linear structures). It was, indeed, in the context of image-processing applications that Krishnapuram and Keller[8] reformulated the fuzzy clustering problem so as to permit the sequential isolation of clusters. This methodology, called *possibilistic clustering,* does not rely, like previous approaches, on prior knowledge about the number of clusters and permits it to take full advantage of clustering methods based on the idea of *prototype.*

Prototype-based classification methods[7] are based on the idea that a data set can be represented, in a compact manner, by a number of prototypical points. The well-known *fuzzy c-means* method of Bezdek—the earliest fuzzy-clustering approach exploiting this idea—seeks to describe a data set by a number of prototypical points lying in the same domain as the members of that data set. Extensions of this basic idea, based on generalizing the notion of prototypical structure in a variety of ways (e.g., as line or curve segments in some Euclidean space) are the basis for methods that seek to represent data sets in terms of structures that have been predefined as being of particular interest to those seeking to understand the underlying physical systems being studied. Generally speaking, however, these methods require that prototypical structures belong to certain restricted families of objects, so as to exploit their structural properties (e.g., the linear structure of line segments or hyperplane patches).

The generalized clustering methodology presented in this paper belongs to this type of approach, extending it by considering arbitrary definitions of interesting structures provided by users by means of a family of parameterized models $M = [M_\alpha]$ and a set of relations between them.[3,9] In addition to a variety of geometric structures, these models may also be described by means of structures (e.g., neural networks) learned from significant examples of the features being defined or in terms

of very general constraints that features might satisfy to some degree (*soft* or *fuzzy* constraints). As is the case with possibilistic clustering methods, our approach is based on the formulation of the qualitative–feature identification problem in terms of the optimization of a continuous functional $Q(F,M_\alpha)$ that measures the degree of matching between a fuzzy subset F of the data set and some instantiation M_α of the family of interesting models.[2]

Our approach recognizes, however, that simple reliance on optimization of a *single* performance index Q would typically result in the generation of a large number of features with small extent and poor generalization, since it is usually easier to match smaller subsets of the data set than significant portions of it. For this reason, it is necessary to consider, in addition to measures Q of representation quality, additional criteria S gauging the size of the structure being represented. Furthermore, it may also be necessary to consider application-specific criteria introduced to assure that the resulting features are valid and meaningful (e.g., constraints preventing selective picking of sample points so that they lie, for example, close to a line in sample space).

This multiobjective problem might be treated by aggregation of the multiple measures of feature desirability into a global measure of cluster quality.[9] A problem with this type of approach, which is close in spirit to minimum description length methods,[10] is the requirement to provide *a priori* relative weights to each of the objectives being aggregated. It should be clear that assignment of larger weight to measures Q of quality representation would lead to small features with higher degrees of matching to models in the prototype families. Conversely, assigning higher weights to measures S of cluster extent would tend to produce larger clusters, albeit with poor modeling ability. Ideally, a family of optimization problems, each similar in character to the others but with different weights assigned to each of the aggregated objectives, should be solved so as to produce a full spectrum of candidate clusters.

Rather than following such a path—involving the solution of multiple problems—our approach relies, instead, on a reformulation of the generalized clustering problem as a multiobjective optimization problem involving several measures of cluster desirability.[2] In this formulation, subsets of the data set of potential interest are *locally optimal* in the *Pareto sense;* that is, they are *locally nondominated* solutions of the optimization problem. (The notions of proximity and neighborhood in feature space are application dependent). Locally nondominated solutions of a multiobjective optimization problem are those points in feature space such that their neighbors do not have better objective values for all objectives while being strictly superior in at least one of them. (i.e., a better value, for a neighbor, of some objective implies a lower value of another). The set of these solutions is called the *local Pareto-optimal* or *local effective frontier.*

We employ a multiobjective genetic algorithm (MGA)[2] based on an extension of methods originally proposed by Horn, Napfliotis, and Goldberg[11,12] to solve this problem. These methods are particularly attractive tools to solve such complex optimization problems because of their generality and their ability, stemming from application of *niched optimization* procedures, to isolate local optima. The set of solutions produced by the MGA is then analyzed by a hierarchy of evolution

programs that produce a compact representation of the features and of the interesting relations between them.

PROBLEM

Biological sequences, such as DNA or protein sequences, are good examples of the type of complex objects that maybe described in terms of meaningful structural patterns. Availability of tools to discover these structures and to annotate the sequences on the basis of those discoveries would greatly improve the usefulness of these repositories that currently rely on methods developed on the basis of computational efficiency and representation accuracy, rather than on terms of structural and functional properties deemed to be important by molecular biologists.

An important example of biological sequences are DNA sequences of gene upstreams that contain significant promoters or regulatory elements viewed as dyad spaces (D).[13] Members of this space correspond to short words (w_1, w_2) with a variable nucleotide content (n_s) such as: $D = w_1.n_s.w_2$.

Another important problem involving DNA sequences, those involving *repetitive elements,* is similar in character to the questions considered in this paper. These sequences include nucleotide subsequences, currently lacking adequate knowledge of their function, that appear repeatedly in the genome of species, such as *Tripanosoma cruzi*, and that are characterized by a higher mutation rate than other nucleotide sequences.

One of such kind of repeated sequence is that called *short interspersed repetitive element* (SIRE).[4] SIRE, which is distributed in all chromosomes and has between 1,500 and 3,000 copies per genome, and is delimited by two vague subsequences that, in some cases, can only be specified in an imprecise or partial way. Usually, a sequence such as `TTTTTTNTTTTTNTT` appears before SIRE, whereas a sequence like `TTATT` may appear at the end. (The letter `N` indicates that any nucleotide is considered a good match for the position.)

A number of approaches, based primarily on computational-efficiency considerations. have been developed to recognize this type of patterns by alignment of sequence and pattern strings. Heuristic methods, such as FASTA, BLAST, or procedures based on dynamic programming, produce sequence descriptions on the basis of global, semiglobal, or local criteria that either incorporate various weights and parameter values, or that make assumptions about undesirable measurements or domain knowledge.

Biologists, however, are usually interested in obtaining all possible descriptions of the sequences in terms of interesting patterns without being burdened, for example, with the chore of making assumptions about the possible location of the pattern in the sequence (see FIGURE 1 A and B), or with the problems associated with imprecise or incomplete knowledge about the pattern being sought (see FIG. 1 C and D, respectively).

This paper focuses on methods to treat problems similar to those briefly sketched in the previous paragraphs. These methods seek to describe DNA sequences in terms of interesting elastic or fuzzy patterns that are meaningful to experts, but that require the application of methods capable of dealing with vague and imprecise information and knowledge.

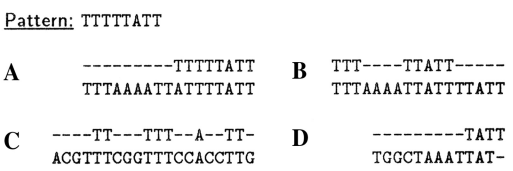

FIGURE 1. A and **B**. Different pattern locations. **C** Imprecise or fuzzy patterns. **D**. Incomplete patterns.

BIOLOGICAL SEQUENCE DESCRIPTION METHODS

In this paper we describe results of the application of the ideas discussed in GENERALIZED CLUSTERING to the discovery of interesting qualitative features in DNA sequences. The notion of interesting feature is formally defined by means of a family of parameterized models $M = \{M_\alpha\}$ specified by domain experts[2] who are interested in finding patterns, such as epoch descriptors of individual or multiple DNA sequences. These idealized versions of prototypical models are the basis for a characterization of clusters as cohesive sets that is more general than their customary interpretation as "subsets of close points." Our approach to the treatment of this problem is based on a two-level methodology consisting of a *model recognition* or *pattern matching* step followed by a *description summarization* process.

- The *multiobjective genetic-based clustering method for biological sequence recognition* (MGCM-BSR) was designed to recognize instances of interesting features (or *model recognition*) by solution of an optimization problem defined over the space of potential features (usually corresponding to the some subset of the set of all fuzzy subsets of the data set). Our approach is noteworthy in that, recognizing that there are multiple measures of cluster quality and desirability, poses the clustering problem as a multiobjective optimization problem rather than relying on weighted linear combinations of performance and penalty functions that are sensitive to small changes in the weighting factors. Our method, based on evolutionary computation techniques, seeks to find interesting features that are not locally dominated; that is, that are locally optimal in the sense that there are not neighboring solutions that are at least equal in all objectives and strictly superior in at least one of them. The set of these solutions is called the *local Pareto-optimal,* or *local effective,* frontier.
- The *genetic-based method for biological sequence summarization* (GM-BSS) is employed, after application of MGCM-BSR, to summarize the local effective frontier and to produce a compact description of the set of interesting features. The major reason for this summarization step is the usually large (even infinite) cardinality of the local effective frontier[11,14] thus limiting its usefulness. The identified set may also contain suboptimal or spurious solutions—an inherent difficulty of algorithms for the solution of multiobjective optimization problems[15]—that must be repaired or eliminated. Furthermore, certain solutions are so close in character and interpretation that they may be

summarized, once again, by prototypical examples. The summarization method, GM-BSS, is based on a family of hierarchical GAs;[5] that is, a collection of several nested summarization algorithms that work in sequential order over solutions and domains. These methods produce compact representations of the Pareto-optimal frontier by extraction of its significant characteristics, summarization of such salient aspects, and descriptions of interesting relations between them.

The remainder of this paper is devoted to the application of these methods to the description of DNA sequences.

Model

Our methodology seeks to extract substrings of a DNA sequence satisfying the constrains imposed by experts through a family M_α of models of interesting features. These models are typically formulated by experts who may already know, for example, that some substrings (or classes of substrings) codify proteins or regulate a particular form of gene expression.

In our application to-DNA sequence identification, flexible or elastic models are provided in the form of a string of DNA nucleotides in linear order (e.g. $A_1T_2T_3C_4G_5G_6$). These patterns are elastic in the sense that they might fit a particular DNA subsequence to various degrees, measured on a $[0,1]$-scale. Domain experts might define patterns, for example, employing vague characterizations, such as: a partial or imprecise match with TTTTTATT, followed by a sequence of arbitrary length, and ending, exactly, in TTATT.

The elasticity of these patterns lies primarily in the fact that the sequence being sought might not match a string of n successive characters in the nucleotide chain but, rather, it may approximately match various substrings while still maintaining the linear order specified in the pattern. Mismatches, lengthy strings between matches, or matches characterized by partial matches with very small model substrings result in penalties that decrease the value of the functional that defines quality of matching. In FIGURE 2 are shown different types of elastic matching between a DNA sequence s and a model M defined by means of the pattern TTTTTATT.

A pattern *perfectly matches* a sequence if, as shown in FIGURE 2A, it exactly matches, without gaps, a sequence substring. The quality of matching decreases as spaces or mismatches are introduced to match sequence and pattern, as shown in FIGURE 2C. From a biological viewpoint, identification of mismatches and matching gaps (spaces) is important since they are known to be related to phylogenetic mutations and gaps.[16]

Models such as those considered in FIGURE 2 may be formally defined as follows. The degree of matching M_Q between a pattern P and a sequence $s = b_1, \ldots, b_n$, is given by the value

$$M_Q(P, s) = \max_F I[F(P), s],$$

where $I[F(P),s]$ is a measure of the degree of matching of sequence $F(P) = a_1, \ldots, a_n$, obtained from P by addition of spaces and of the transformation F itself.

The degree of matching I is typically defined as a function of the nucleotide to nucleotide correspondence between the same positions in $F(P)$ and s

$$M_Q(F(P), s) = (a_1 \simeq b_1) \wedge (a_2 \simeq b_2) \wedge \ldots \wedge (a_n \simeq b_n),$$

```
P  =  ------TTTTTATT      A
s  =  TTTGGGTTTTTATT

P  =  TTT------TTATT      B
s  =  TTTGGGTTTTTATT

P  =  TTTTT------ATT      C
s  =  TTTGGGTTTTTATT
```

FIGURE 2. Model matching: **(A)** perfect matching, **(B** and **C)** imprecise and vague matching.

where $\simeq$ stands for the fuzzy predicate *approximately matches*. The ground predicate $\simeq$ is modeled, using standard conventions, by means of a fuzzy relation. This formula permits computation, by application of fuzzy-logic combination operators, of the degree by of matching between s and a sequence constructed from P via the transformation F.

In our application to DNA-sequence modeling, the ground predicate $\simeq$ is calculated by means of several fuzzy relations on the components of the DNA alphabet. FIGURE 3 shows several such relations.[17] (In affine gaps, $_{-1}$ corresponds to an initial gap and $_{-2}$ to the extension of a previous adjacent gap.) Finally, the conjunction operators employed to compute the degree of approximation include compensatory functionals such as product, arithmetic mean, or "anding" aggregation parameterized operators.

Biological Sequence Recognition

Among various possible MGA approaches to the treatment of multiobjective problems[15] we have chosen to base our methodology on the well known method of Horn, Napfliotis, and Goldberg,[11,12,18] which does not have certain weaknesses that characterizes alternative algorithms. A significant feature of this method is its reliance on restricted competition ("niches") between chromosomes to determine all non-dominated solutions of the multiobjective optimization problem. This niched-based approach easily permits the introduction of various changes in order to better

	A	T	C	G
A	0	5	5	1
T	5	0	1	5
C	5	1	0	5
G	1	5	5	0

	A	T	C	G	$_{-1}$	$_{-2}$
A	1	-1	-1	-1	-4	-2
T	-1	1	-1	-1	-4	-2
C	-1	-1	1	-1	-4	-2
G	-1	-1	-1	1	-4	-2
$_{-1}$	-4	-4	-4	-4	$\perp$	$\perp$
$_{-2}$	-2	-2	-2	-2	$\perp$	$\perp$

FIGURE 3. Relations between model components: **(A)** transition/transduction, **(B)** affine gaps.

handle the identification of local optima by performing nested or simultaneous sharing in objective and/or decision variable spaces. Furthermore, the method relies on Pareto domination tournaments—determining the dominance status of competitors by comparing each of the selected competitors to a sample and selecting a winner—to introduce locality. Finally, our approach introduces various GA features like elitism[14,19] and mating restrictions by speciation.[15,18,19] We specifically evaluate the integration of these features in the treatment of the localized problem, basing our considerations on the most general performance indexes considered in MGA.[14,15]

Two objectives were considered, corresponding to the quality Q and extent S of a particular description, respectively. Clearly, these objectives are conflicting in the sense that it is easier to generate more accurate explanations of smaller rather than larger subsets of the data set. The multiobjective optimization problem is that of maximizing Q and S in the local Pareto sense.

We describe our algorithm, first introducing the notation.

- Σ is the nucleotide alphabet, $\Sigma = \{A, C, G, T\}$, and Σ^* is the set of all finite sequences of Σ.
- Λ is the extended alphabet, $\Lambda = \Sigma \cup \{-\} = \{A, C, G, T, -\}$, and Λ^* is the set of all finite sequences of Λ.
- s, p, t and s', p' are sequences in Σ^* and in Λ^*, respectively.
- $|s|$ is the length of s.
- g_t is the number of gaps or spaces introduced in t to obtain t'.
- Q and S are the quality and extent objectives, respectively.

Our MGA includes the following computational steps:

1. Initialize the population P with $r_1, \ldots, r_{\text{pop}}$, selected randomly; that is, select the number of gaps g_p that will be included in p' ($|p'| = |p| + g_p$); define their places in the sequence p and, based on these selections, add the necessary number of gaps g_s in s' in valid positions, so as to obtain a global alignment. Note that $g_s = |s| - |p| + g_p$ (see FIGURE 4).
2. Evaluate each individual in P in both objectives, Q (quality) and S (extent):
   - S is measured as the size of the pattern present on the objective sequence. Note that gaps may be inserted in both objective and pattern sequence.

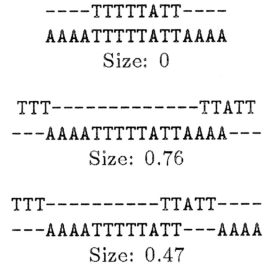

FIGURE 4. Chromosome description.

- Q is calculated considering only the subsequence defined by the extension of the pattern since we are trying to measure the similarity between the pattern and the objective sequence and we are not interested in what occurs before or after it. The evaluation of Q is based on the model described in previous section, which relies on fuzzy relations based on matches, mismatches, and gaps. These relations are used in a less restrictive way than BLAST or FASTA algorithms.[15] Initial and final gaps are not taken into account either in the quality or in the extent functions.

3. Select P individuals with reposition (tournament). This selection is made by comparing two individuals r_i and r_j against a comparison set C (of size *com*). If r_i is not *locally dominated* by any element of C and r_j is, then r_i is the tournament winner. If r_j is not locally dominated by any element of C and r_i is, then r_j is winner. If both individuals are locally dominated or if neither r_i nor r_j are locally dominated, employ a sharing process in *variable* (global Hamming distance[20]), objective, or simultaneous spaces (see FIGURE 5).
4. If *elitism*, replace some solutions from the *elite set* with the best solutions found in the population and replace the worst solutions from the population with the best solutions from the elite set.[14]
5. Apply crossover operators with probability p_{cro}. If *speciation*, use mating restrictions. For this particular implementation, two crossover operators were applied employing the approach suggested by Horng.[21] These operators differ from classic GAs operators in that they are applied to *crossover blocks* rather than to the whole chromosome. Thus, select two chromosomes randomly, create two crossover blocks for each, apply the following operators to each block separately and join the resulting blocks:

- One-point-combine selects a point in the crossover block and takes the existent gaps from that point to the left from its father, and the existing gaps to the right of that point from its mother, to generate a new child.
- Good-pos-combine extracts all the gaps that are common to both, father and mother, and copies them to the child adding the necessary extra gaps (taken from the father or mother) to perform a global alignment. (Our procedure to process unfeasible solutions always maintains a valid population, changing those individuals that do not satisfy required constraints to the nearest valid solution.)

6. Apply mutation operators with probability p_{mut} using the following operators:

```
Algorithm Sharing
input: candidate1, candidate2
   for i ← 1 to Newpop do
      #niche(candidate1) and #niche(candidate2)
   if #niche(candidate1) > #niche(candidate2) then
      returncandidate2
   else return candidate1
```

FIGURE 5. Sharing pseudocode.

- Add-gap, adds a new gap in any position of the chromosome. Note that both the pattern and the objective sequence may be changed.
- Delete-gap, deletes an existent gap from a chromosome.
- Shift-gap, deletes an existent gap and introduces a new adjacent gap on left or right hand of the original one.

7. Return to Step 2, if the number of generations completed is less than *gen.*
8. Return the elements r_i of the final population P.

Qualitative Feature Summarization

Summarization procedures seek to produce compact representations of the Pareto-optimal frontier by producing compact descriptions of its significant characteristics and identifying important relations between their features. In the DNA-sequence applications being discussed, these processes include exclusion of solutions of the multiobjective optimization problem that are dominated by similar solutions (fuzzy domination), grouping similar solutions (clustering) by prototypes having close values to the objective functional Q and corresponding to the same variable space, extraction of irrelevant or repeated solutions, and finally, the organization of an effective frontier on the basis of the notion of approximate inclusion as interval hierarchies (trees). Three different criteria were defined to guide the performance of the summarization processes:

- *Fuzzy domination.* Certain solutions—although lying in the effective frontier and being thus non-dominated—are in fact dominated by other, similar, more relevant solutions. To take care of this problem, we employ the following feature summarization criteria.

$$\text{If} \left\{ \begin{array}{ll} a \sqsubseteq b, & \text{and if} \\ Q(a) \gtrsim Q(b), & \text{and if} \\ \neg(S(a) \lll S(b)), & \end{array} \right\} \text{ then delete } a,$$

  where $\sqsubseteq$ is the fuzzy relation *approximately included*, $\gtrsim$ is the fuzzy relation *approximately larger*, $\lll$ is the fuzzy relation *approximately very smaller*, and S and Q are the extent and quality objectives, respectively.
- *Exclusion of irrelevant solutions.* Introduction of new fuzzy relations based on expert biological criteria, called *affine-gaps relations* (see TABLE 3 below)[16] allows modification of the feature-quality measure to benefit models with adjacent gaps. This modification permits the elimination of features with an excessive number of gaps that may have little biological relevance.
- *Hierarchical organization of remaining features by inclusion.* The final step of the summarization process is the hierarchical organization of relations by inter-features relations defined as being interesting by the user. In our DNA-sequence application only one such relation (set inclusion) was considered.

Summarization procedures based on these criteria are implemented by a hierarchy of evolution programs[5] where, if $EP_i \prec EP_{i+1}$, then the evolution program EP_i

is a weaker method than EP_{i+1}; that is, $dom(EP_{i+1}) \subseteq dom(EP_i)$. Procedures enforcing related constraints are implemented as multimodal EP (EP_1 and EP_2) that rely solely on a shift-gap mutation operator, elitism, and niching in the variable space based on global Hamming distance.[20] The fitness function employed in these GA relies on a quality measure Q modified to account for consideration of the *affine gap* relation. Selection was implemented by a tournament procedure.

EXPERIMENTAL ALGORITHM EVALUATION

The methodology described in the previous section was applied to the discovery of SIRE patterns, described in the section PROBLEM, in the DNA sequence of *Tripanosoma cruzi.*

The model considered in our experiments, (TTTTTATT), was the result of a consensus between domain experts.[4] Partial matches, incorporating gap, were sought in the artificial sequence (TTTAAAATTATTTTATT).

Our experimental objectives include—in addition to the discovery of meaningful biological sequence descriptions—the evaluation of various alternative architectural variations of the MGCM-BSR algorithm so as to determine the most effective implementation for the identification of multiple, localized, interesting epochs. In our analysis we evaluated, among many possible combinations of evolutionary architectures, corresponding to selective implementation of various processes. That is,

- niching in variable (V), objective (O), and variable and objective ($V+O$) spaces,
- elitism strategies ($O+E$, $V+E$), and
- speciation ($V+O+S$, $V+O+S+E$).

We first make a brief presentation of MGA performance metrics, examining later, from their perspective, the results produced by the algorithm MGCM-BSR. Finally, we present and analyze results of application of our two-step methodology (MGCM-BSR+GM-BSS).

Performance Metrics

To compare the quality of the various features of the MGCM-BSR algorithm, we extend metrics of performance proposed by Zitzler, Thiele, and Deb[14] so as to be able to deal with localized solutions.

Distance to the Pareto-Optimal Set (M_1)

f_1, distance in objective space

$$f_1(X) = \frac{1}{|X|} \sum_{p \in X} \min\{\|p - \bar{p}\| : \bar{p} \in \bar{Y}\},$$

where X is the set of solutions in the last generation, Y is the set of optimal Pareto solutions, and $\|\cdot\|$ denotes the Euclidean norm.

f_1', distance between nondominated solutions in variable space

$$f_1'(X') = \frac{1}{|X'|} \sum_{a' \in X'} \min\{\|a' - \bar{a}\|_H : \bar{a} \in \bar{X}\},$$

where X' is the set of the nondominated solutions in the previous generation, X is the set of optimal Pareto solutions, and $\|\cdot\|_H$ is a global Hamming distance.[19]

f_1'', distance in objective space (similar to f_1, but with nondominated solutions)

$$f_1''(X') = \frac{1}{|X'|}\sum_{p' \in X'} \min\{\|p' - \bar{p}\| : \bar{p} \in \bar{Y}\}.$$

Distribution of the Front (M_2)

$$f_2(X') = \frac{1}{|S(X')|}\left[\left(\sum_{a' \in S(X')} |\{a' \in X'\}|\right) - \frac{|X'|}{S(X')^2}\right],$$

where $S(X')$ is the set of different solutions in last generation of the nondominant set X'.

Extent of the Front (M_3)

f_3', dispersion of last generation nondominated solutions in variable space

$$f_3'(X') = \sqrt{\sum_{i=1}^{m} \max\{\|a_i' - b_i'\|_H : a', b' \in X'\}},$$

where m is the dimension of the vectors representing individuals.

f_3'', idem f_3' but in the objective space.

Relative Quality of Solutions Produced by Alternative Algorithms[14]

$$C(X_1, X_2) = \frac{\left|\{a_2 \in X_2; \exists a_1 \in X_1 : a_1 \trianglelefteq a_2\}\right|}{|X_2|},$$

where $\trianglelefteq$ means "is dominated by or equal to". A value of $C(X_1,X_2) = 1$ means that all solutions in X_2 are dominated by (or equal to) solutions in X_1, and $C(X_1,X_2) = 0$ means that there does not exist a solution in X_2 covered by X_1. It is important to note that $C(X_1,X_2)$ and $C(X_2,X_1)$ should both be calculated because

$$C(X_1, X_2) \neq 1 - C(X_2, X_1).$$

Experiments

The parameters employed in all experiments for the identification of SIRE patterns are listed in TABLE 1. The values of various performance metrics for different runs of the algorithm MGCM-BSR, under various conditions, are shown in

TABLE 1. Parameters of the MGCM-BSR algorithm

Parameter	Value
Number of generations	150
Population sizes	3,000
Crossover probability	0.6
Mutation probability	0.3
Comparison set	2,500
Niche size (σ)	4

TABLE 2. Performance metrics M_1, M_2, and M_3

		V	*O*	*O+E*	*V+O*	*V+S*	*V+O+S*	*V+O+S+E*
M_1	f_1	2.14	1.18	0.95	1.63	2.09	0.74	0.62
	f_1'	0.11	0.02	0	0	0.20	0.97	0
	f_1''	0.06	0.01	0	0	0.14	0.08	0
M_2	f_2	1.82	32.21	578.52	16.86	0.94	321.68	240.98
M_3	f_3'	17.00	16.00	17.00	18.00	19.00	17.00	20.00
	f_3''	29.52	33.10	35.77	35.77	29.15	35.77	35.77

TABLES 2 and 3. (The true Pareto optimal front for this problem is unknown.[11,12] In our experiments, this frontier has been approximated employing the optimal solutions of various algorithmic runs.)

Several variants of the generic algorithm were evaluated to provide knowledge to help understand the relations between parametric and architectural choices and algorithmic performance:

- M_1. TABLE 2 shows that, if the distance is greater than zero then, some nondominated solutions are not present in the set of solutions produced by the algorithm. Some solutions were not found for variants including either *O* or *V* options (see f_1' and f_1'' in TABLE 2). This problem is caused because the sharing process ignored genotypic composition of the individuals. A similar problem occurs in the case of *V* and *V+S*, although here the objective functions are ignored by the sharing process. Variants that include *O+V* present the best trade-off behavior.
- M_2. Low values of this index agree with well know results. As shown in TABLE 2, the options including *V* resulted in the lowest performance metric values. These low values are caused by the focused attention (by the sharing process), when only the *V* option is employed, on providing good niches with equal size without concern for their quality. Elitism also influenced the values of this performance measure, since the corresponding process stores a fixed number (not proportional) of elitist solutions (not necessarily different) per pattern size.

TABLE 3. Results obtains with variants of MGCM-BSR

	V	*V+O*	*O*	*O+E*	*V+S*	*V+O+S*	*V+O+S+E*
V	—	0.26	0.17	0.40	0.14	0.38	0.38
V+O	0.14	—	0.20	0.59	0.12	0.64	0.45
O	0.10	0.21	—	0.36	0.10	0.24	0.25
O+E	0.12	0.29	0.14	—	0.10	0.57	0.46
V+S	0.15	0.21	0.21	0.28	—	0.25	0.26
V+O+S	0.06	0.23	0.03	0.56	0.05	—	0.55
V+O+S+E	0.07	0.20	0.03	0.56	0.06	0.58	—

- M_3. These performance measures evaluate how distant are the extreme individuals in the nondominated frontier. A higher value of these measures indicates that the variant is more effective in the identification of extreme solutions. The experimental results indicate that the performance of various alternatives is similar when seen from the perspective of variable-space measures (see f_3'). Although the difference is more marked when objective-space measures are concerned (see f_3''), the values indicate that all variants are sensitive to extremes of the frontier. Moreover, solutions are correctly preserved during the evolutionary process (see $V+O+S+E$ option).
- *Comparison between algorithms*. TABLE 3 shows the proportion of equal or dominated solutions between pairs of algorithmic variants employing various options. A value of 1 (the highest possible) indicates that all solutions produced by an algorithm are either dominated or equal to those of the comparison algorithm, whereas a value of 0 (the lowest possible) indicates that no solution is dominated by that of the comparison algorithm. The variants $V+O+S$ and $V+O+S+E$ have high values, showing that the solutions produced by the corresponding algorithms are dominated by alternative variants. Detailed analysis of the results shows, however, that the solutions produced by the $V+O+S$ and $V+O+S+E$ variants—although being dominated by solutions produced by alternative options—do not differ significantly from them, that is, they correspond approximately to same epochs or spatial intervals. We conclude, therefore, that this index cannot be relied upon to provide an effective measure of algorithmic performance.

In view of the results shown in TABLES 2 and 3, it is our conclusion that $V+O+S+E$ is a good architectural choice providing a reliable, robust, algorithm. The results presented in the rest of this section are those produced by application of this algorithmic choice.

FIGURE 6 shows local and global results (A and C, respectively) of application of MGCM-BSR to the problem of identifying SIRE patterns. Other "interesting" descriptions, such as the improved global, intermediate and semiglobal solutions shown in FIGURES 7A, 6B, and 7B, respectively, were also identified by this algorithm.

Although there were other interesting descriptions in the Pareto-optimal frontier (see FIGURES 8 and 9) covered by the MGA, we only show only a small sample because of space limitations.[19]

It is important to note also that our MGA produced all global Pareto-optimal solutions for patterns of size 0.23 and 0.29. Other interesting solutions were identified by the MGCM-BSR localized selection policy and by nested/simultaneous

FIGURE 6. **(A)** local, **(B)** intermediate, and **(C)** global descriptions of SIRE.

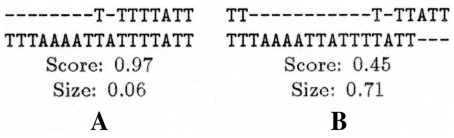

FIGURE 7. **(A)** Improved and **(B)** semiglobal descriptions of SIRE.

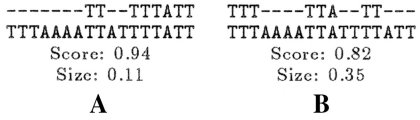

FIGURE 8. Other relevant descriptions of SIRE.

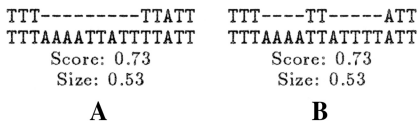

FIGURE 9. Different epochs, similar objective-value solutions.

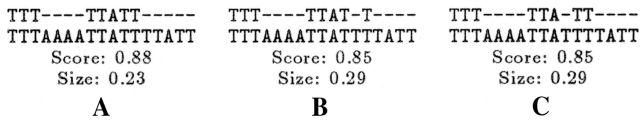

FIGURE 10. Redundant nondominated solutions.

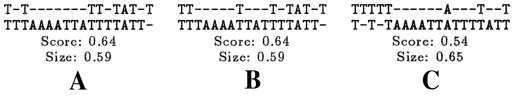

FIGURE 11. Nondominated solutions with low significance.

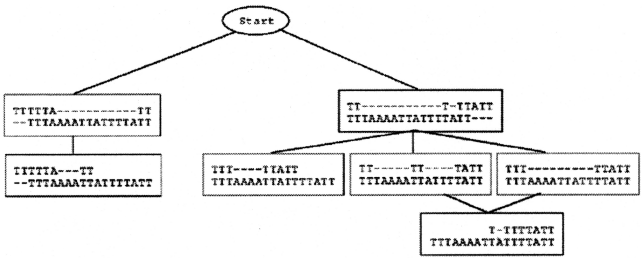

FIGURE 12. Summarized effective frontier.

sharing processes. FIGURE 9 (e.g., the solution having extent 0.73 and quality 0.53) shows several examples of this class of solutions.

We remarked previously, that the solution set may have a very large cardinality, or even contain an infinite number of solutions.[11,14] This inconvenient fact is illustrated in FIGURE 10, which shows nondominated solutions in variable space.

To simplify the results produced by the MGA optimization algorithm, we employed a sequence of evolutionary methods EP_i, implemented as part of the GM-BSS algorithm to summarize the output of MGCM-BSR. The method EP_1 was applied to nondominated solutions produced by the $V+O+S+E$ of MGCM-BSR (EP_0).

For example, redundant solutions, such as those presented in FIGURE 10 were reduced to the prototype shown in FIGURE 10A. Subsequent application of the algorithm EP_2 with affine gaps relation, resulted (see FIGURE 11) in replacement of solutions of low biological significance with better examples.

The summarized effective frontier is shown in FIGURE 12, which graphically represents relations of inclusion between DNA intervals.

In closing this section it is important to remark that our generalized-clustering approach found all solutions identified by alternative methodologies, such as FASTA, BLAST, and dynamic programming (with local, global, and semiglobal options, and various parameter settings).

Furthermore, additional sequences of potential biological significance, were also determined to be interesting on the basis of tradeoff considerations between feature quality and extent.[20]

CONCLUDING REMARKS

Generalized-clustering algorithms—solving multivariable, multiobjective, optimization problems—provide effective tools to identify interesting features that help to understand complex objects, such as DNA sequences. Summarization algorithms implementing a hierarchy of evolutionary programs further aid in interpreting the results of these optimization methods, producing compact descriptions of interesting structures and of significant relations between them.

Our research currently seeks to extend our MGA-based methodology focusing particularly on parallel evolutionary implementations that might be applied to problems arising from the description of DNA sequences such as that of *Tripanosoma cruzi*.[21]

REFERENCES

1. ZADEH, L.A. 2000. Outline of a computational theory of perceptions based on computing with words. *In* Soft Computing and Intelligent Systems: Theory and Applications. N.K. Sinha, M.M. Gupta & L.A. Zadeh, Eds.: 3–22. Academic Press, San Diego.
2. RUSPINI, E.H. & I. ZWIR. 2001. Automated generation of qualitative representations of complex object by hybrid soft-computing methods. *In* Pattern Recognition: From Classical to Modern Approaches. S.K. Pal & A. Pal, Eds. World Scientific Company, Singapore.

3. Zwir, I. & E.H. Ruspini. 1999. Qualitative object description: initial reports of the exploration of the frontier. Proc. EUROFUSE-SIC\acute99, Budapest, Hungary. 485–490.
4. VÁZQUEZ, M., C. BEN-DOV, H. LORENZI, *et al*. 2000. The short interspersed repetitive element of *Trypanosoma Cruzi,* SIRE, is part of VIPER, an anusual retroelement related to long terminal repeat retrotransposon. Proc. Natl. Acad. Sci. USA **97**(5): 2128–2133.
5. MICHALEWICZ, Z. 1999. Genetic Algorithms + Data Structures = Evolution Programs. Springer.
6. RUSPINI, E.H. 1969. A new approach to clustering. Inform. Contl. **15**(1): 22–32.
7. BEZDEK, J.C. 1998. Fuzzy clustering. *In* Handbook of Fuzzy Computation. E.H. Ruspini, P.P. Bonissone & W. Pedrycz, Eds.: F6.2. Institute of Physics Press.
8. KRISHNAPURAM, R. & J. KELLER. 1993. A possibilistic approach to clustering. IEEE Transactions on Fuzzy Systems, 98-110.
9. RUSPINI, E.H. & I. ZWIR. 1999. Automated qualitative description of measurements. Proc. 16th IEEE Instrumentation and Measurement Technology Conf.
10. RISSANEN, J. 1989. Stochastic Complexity in Statistical Inquiry. World Scientific.
11. HORN, J. & N. NAFPLIOTIS. 1993. Multiobjective optimization using the niched pareto genetic algorithm. IlliGAL 93005. Illinois Genetic Algorithms Laboratory (IlliGAL), Department of General Engineering, University of Illinois at Urbana-Champaign.
12. HORN, J., N. NAFPLIOTIS & D. GOLDBERG. 1994. A niched Pareto genetic algorithm for multiobjective optimization. Proc. First IEEE Conf. on Evolutionary Computation, 82–87.
13. VAN HELDEN, J., A. RIOS & J. COLLADO-VIDES. 2000. Discovering regulatory elements in non-coding sequence by analysis of space dyads. Nucl. Acids Res. **28**(8): 1808–1818.
14. ZITZLER, E., L. THIELE & K. DEB. 2000. Comparison of multiobjective evolutionary algorithms: empirical results. Evol. Comput. **8**(2): 173–195.
15. DEB, K. 2001. Multi-Objective Optimization using Evolutionary Algorithms. John Wiley & Sons.
16. SETUBAL, J. & J. MEIDANIS. 1997. Introduction to Computational Molecular Biology. PWS Publishing Company.
17. CHICLANA, F., F. HERRERA & E. HERRERA-VIEDMA. 2002. A note on the internal consistency of various preference representations. Fuzzy Sets Systems. **131:** 75–78.
18. BÄCK, T., D. FOGEL & Z. MICHALEWICZ, Eds. 1997. Handbook of Evolutionary Computation. Institute of Physics Publishing and Oxford University Press.
19. FONSECA, C. & P. FLEMING. 1995. Multiobjective genetic algorithms made easy: selection, sharing and mating restriction. *In* Genetic Algorithms in Engineering Systems: Innovation and Applications. 42–52. IEEE.
20. ZALIZ, R.C.R. 2001. Reconocimiento y Descripción de Objetos Complejos en Biología Molecular. Masters Thesis, Universidad de Buenos Aires, Argentina.
21. HORNG, J., L. CHING-MEI, B. LIU & C. KAO. 2000. Using genetic algorithms to solve multiple sequence alignments. Proc. of the Genetic and Evolutionary Computation Conf., 883–890.
22. MACHI DNA SERVER. 2002. <http://machi.dc.uba.ar:8080>.