Adaptive Evolutionary Algorithms for the Delineation of Local Labour Markets

F. Flórez-Revuelta, Member, IEEE, J.M. Casado-Díaz, and L. Martínez-Bernabeu

Abstract—Given a territory composed of basic geographical units, the delineation of local labour market areas (LLMAs) can be seen as a problem in which those units are grouped subject to multiple constraints. In previous research, standard genetic algorithms were not able to find valid solutions, and a specific evolutionary algorithm was developed. The inclusion of multiple ad hoc operators allowed the algorithm to find better solutions than those of a widely-used greedy method. The experimentation process showed that the rate of success of each operator in generating good individuals is different and evolves with time. We therefore propose different adaptive alternatives that modify the probabilities of application of each operator throughout the evolutionary process, and compare the results of such adaptive approaches with previous results and a greedy method.

I. INTRODUCTION

Local labour market areas (LLMAs) are geographical entities defined to serve as a territorial framework to design, implement and monitor effective labour market policies and statistical operations at sub-national levels. The success of these key policies crucially depends on the adequacy of the LLMAs delimitation. According to the code of good practices established by Eurostat [1] to guide the selection of a specific procedure, the resulting LLMAs geography must be conformed by disjoint areas exhaustively covering a given territory, characterised by a high degree of self-containment in terms of travel-to-work trips (i.e. most workers in a specific LLMA must live in that area and most of the LLMA’s employed residents should also work locally), and relatively homogeneous in population size (exceeding a minimum size constraint, for instance). The problem is therefore the grouping of basic spatial units (BSU) -such as districts, municipalities or counties- into functional areas so that the proportion of workers that cross their boundaries in their travel to work is low, while the number of defined areas is maximized. This problem is analogous to a Graph Partitioning Problem (GPP) where the optimal number \( k \) of partitions is unknown and the requisite of size homogeneity is relaxed or removed, so it is expected to be at least as hard as the standard GPP (that is NP-hard). Thus, an exhaustive resolution of the problem is not possible.

One of the more widely and successfully used official procedures is that of Travel-to-Work Areas (TTWAs) in the UK (it is fully described in [2], and has been applied with minor changes in other countries: [3], [4], [5], and [6]). This regionalization method can be defined as a greedy algorithm that iteratively aggregates a given set of BSUs based on the relative attraction (in terms of commuting flows) between them until all the defined functional areas meet both self-containment and size constraints (in terms of employed population). The method allows reaching adequate solutions with little CPU time.

In order to get solutions closer to the optimal, an evolutionary approach [7] was designed. The multiple constraints which are part of the problem cause the number of valid solutions –those that meet the constraints– to be extraordinarily small with regards to the search space, so standard genetic operators didn’t lead to valid solutions in a reasonable lapse of time. This is the reason why an extensive set of specific crossover and mutation operators was proposed [8]. Whilst some of which have similarities with those used in other grouping and clustering problems, others are much more related to the very specific nature of the problem.

One of the first questions that arose when using this set of ad-hoc operators was the fine-tuning of the probabilities of application of each operator, because each one showed different performance ratios changing with time and the problem’s constraints due to their specific characteristics (different complexities of the subalgorithms from \( O(1) \) to \( O(N) \)). This is, the quality of the solutions obtained at different CPU times varies with different settings of the operators probabilities, and good settings for one problem are not so good for another.

So, in this paper, we consider the dynamic adjustment of the probability of application of each one of the operators based on different parameters taken throughout the evolution. We have studied different possibilities in order to modify these probabilities:

- Reward to the operator that is applied if the new individual improves the best individual of the population, their parents or some individual in the current population.
- Reward based on the improvements in the fitness obtained with an operator.
- Reward based on improvements in the fitness with respect to the time consumed in generating the new individual.

The structure of the paper is as follows: in next section we introduce a formalization of the problem of delineating a territory into local labour markets. Section III presents the evolutionary proposal to solve the problem, indicating how individuals are represented, as well as the methods of selection and a summary of the crossover and mutation
operators that have been specifically designed. In that same section the use of an adaptive approach is justified through relevant data evidence. In section IV we propose different alternatives that allow the adjustment of the parameters that guide the use of each operator. Finally, in section V we carry out a comparison between the results from the different alternative that have been analyzed, and we also compare them with the results from the original algorithm. Finally, in section VI we give some remarks and introduce some extensions we are currently working on.

II. PROBLEM FORMULATION

Let \( S = \{S_1, S_2, \ldots, S_n\} \) be a set of BSUs (the territory to be divided into LLMAs) and \( W_{S_i, S_j} \) the number of commuters from BSU \( S_i \) to BSU \( S_j \), that is, the number of residents in \( S_i \) that work in \( S_j \) (thus, \( W_{S_i, S_j} \) is the amount of people who simultaneously live and work within the boundaries of BSU \( S_i \)). The objective is to obtain the set of markets (LLMAs) \( M = \{M_1, M_2, \ldots, M_m\} \), where \( m \) is unknown a priori, so as \( M_i \neq \emptyset \), \( \forall M_i \in M; \bigcup_{i=1}^{m} M_i = S \) and \( M_i \cap M_j = \emptyset, \forall i, j \in [1, m], i \neq j \). That maximizes fitness function \( f \). Let \( II \) be the interaction index between two markets:

\[
II(M_i, M_j) = \frac{W_{M_i, M_j}}{R_i} \times \frac{J_j}{P_{M_i, M_j}} + \frac{W_{M_j, M_i}}{R_j} \times \frac{J_i}{P_{M_j, M_i}}
\]

(1)

where

\[
W_{M_i, M_j} = \sum_{S_i \in M_i} \sum_{S_j \in M_j} W_{S_i, S_j}
\]

(2)

is the total number of commuters residing in the set of BSUs of \( M_i \) that works in any of the BSUs of \( M_j \); \( R_i = W_{(M_i), S} \) the total number of workers residing in \( M_i \); and \( J_k = W_{S, (M_k)} \) the total number of jobs in \( M_k \).

Factor \( PE_{M_i, M_j} \) is the fraction of the employed population residing in \( M_i \) and working in \( M_j \); and \( PJ_{M_i, M_j} \) is the portion of jobs in \( M_j \) that are held by workers residing in \( M_i \).

This interaction index can be the base for different fitness functions. Among them in this exercise we have decided to test our method with

\[
f(M) = \text{card}(M) \times \sum_{S_i \in S} II(S_i, M_S_i - \{S_i\})
\]

(3)

where \( M_S_i \) is the market \( S_i \) belongs to. In our case we calculate the interaction index between a BSU \( S_k \) –which is considered as a mono-BSU market– and the market that would result if that BSU \( S_k \) is substracted from the market \( M_k \) it belongs to. This interaction index between a BSU and the market it belongs to is a generalization of the interaction index used in [2]. The inclusion of the number of LLMAs as a factor allows to reach the highest possible number of independent LLMAs –this is one of the criteria usually applied in practical exercises [9].

Besides, each market \( M_i \in M \) must fulfill two requirements in terms of minimum self-containment percentages \((\beta_1, \beta_2, 0 \leq \beta_1 < \beta_2 \leq 1)\) –i.e. both the proportion of the occupied working locally, and the proportion of jobs filled by local workers must exceed a given threshold–, and minimum size in terms of employed population \((\beta_3, \beta_4, 1 \leq \beta_4 \leq \beta_3)\):

\[
\begin{align*}
\min \left( \frac{W_{M_i, M_i}}{W_{M_i, S}} \times \frac{W_{M_i, M_i}}{W_{S, M_i}} \right) & \geq \beta_1 \\
W_{M_i, S} & \geq \beta_4
\end{align*}
\]

(4)

(5)

Very urbanized environments are in real world characterized by the intensity and complexity of the network of commuting flows, something which makes it difficult to identify isolated groups of BSU. To facilitate the identification of a larger number of separate LLMAs in such environments a trade-off between both constraints (self-containment and minimum size) has been introduced similarly to [2], but using the formulation proposed by Casado-Diaz[9]. According to this proposal, the minimum self-containment requirement is linearly relaxed from \( \beta_2 \) to \( \beta_1 \) for populations sizes from \( \beta_4 \) to \( \beta_3 \). For each market in a given solution, this trade-off is evaluated as follows:

\[
\begin{align*}
\min \left( \frac{W_{M_i, M_i}}{W_{M_i, S}} \times \frac{W_{M_i, M_i}}{W_{S, M_i}} \right) & \geq 1 \\
\frac{\beta_2 - \beta_1}{\beta_4 - \beta_3} & \leq m
\end{align*}
\]

(6)

(7)

We have also included a minimum connectivity requisite to guarantee some degree of territorial contiguity without employing spatial data: a BSU can only belong to a market if it is reachable from any other BSU of that market through the \( \gamma \) largest outgoing/incoming commuting flows of each BSU in the market (we call this functional neighbourhood).

III. EVOLUTIONARY PROPOSAL

Let \( O \) denote the number of different operators that can be applied to generate new individuals. The algorithm maintains a probability vector \( \xi^t = (\xi_i)_{i=1,O} \) that changes with time (generation) \( t \).

The structure of the evolutionary algorithm for the delineation of a given territory follows the next steps:

**Step 1.** Set time \( t = 0 \). Initialize the operators’ probability vector \( \xi^0 \).

**Step 2.** Produce an initial population consisting of \( n_p \) individuals. At least one of the individuals in the population must be valid -i.e. it must meet all the constraints. To assure this the first individual generated consists of a single market covering all the territory. Complete the initial population with \( n_p - 1 \) randomly generated individuals (in practice, all of these are usually invalid solutions).

**Step 3.** Evaluate fitness of all individuals and sort them accordingly.
Step 4. Repeat \( n_o \) times: select an operator, following probability \( \xi^t \), to be applied to generate a new individual. Depending if it is a crossover or a mutation operator, select two or one, respectively, individuals from the current population by fitness-proportional probability. Generate a new individual, and evaluate its fitness.

Step 5. Sort the whole population, composed of \( n_p + n_o \) individuals, by their fitness value.

Step 6. Select the population for the next generation choosing the \( n_p \) best individuals (truncation scheme).

Step 7. Update \( \xi^{t+1} \) according to \( \xi^t \) and the success of that probability vector.

Step 8. Set \( t = t + 1 \).

Step 9. Stop condition: if evolution time \( t_{max} \) has been accomplished, finish. Otherwise, return to step 4.

A. Genetic representation

The individuals which constitute the population represent feasible solutions, that is, the aggregation of all the BSUs composing territory \( S \), into non-overlapping LLMAs. We have used a group-number encoding \([?)\] where each individual is represented by a vector of \( n \) components, each of which corresponds to a BSU of \( S \), and takes the value of the identifier of the market the BSU belongs to (Figure 1). This representations ensures the non-overlapping constraint is fulfilled.

B. Selection

The selection of the individuals to be affected by recombination and mutation operations is performed following a ranking method, according to which those individuals scoring higher in the fitness function have a larger probability of being selected. We use truncation for the selection of the survival individuals that compose the population in the next generation. So for every generation, the population is composed by the \( n_p \) better solutions in the previous generation.

C. Genetic operators

Due to the large number of constraints that the individuals must meet, the usual operators of recombination and mutation seldom lead to valid solutions. This makes the evolution difficult or even unable to progress. For this reason, in addition to the usual operators of recombination and mutation, we designed four ad-hoc crossover operators and eleven mutation operators (see [8] for a detailed description). Then, \( O = 17 \) operators.

Specialized crossover operators consider the codification of both parents when the offspring is generated avoiding discrepancies between all of them. On the other hand, mutation operators have four main functions: division of markets, fusion of markets, reassignment of single BSUs, and reassignment of groups of BSUs. The goal of division operators is to increase the number of markets—i.e. \( \operatorname{card}(M) \)—in the regionalization, so as to improve the detail of the result. Fusion operators eliminate markets to go back in the process of division. Reassignment operators try to improve the solution by reassigning specific BSUs between markets in a local search procedure.

D. Summary of the evolutionary process

To summarize, from an individual in which all the BSUs are merged to conform a single LLLM, successive applications of division and aggregation of markets, reassignment of single BSUs or groups of BSUs between markets, and recombinations, allow increasing the number of LLMAs, assigning the basic geographical units to the relevant LLLM so that the fitness function is maximised.

E. Study of the operators

To test our proposal we use a case study: the delineation of a set of LLMAs in the Region of Valencia, Spain. Travel-to-work data derived from the Spanish Census of Population [10] allowed us to build a \( 541 \times 541 \) origin-destination commuting matrix (where 541 is the number \( n \) of municipalities that integrate the territory), where each cell represents \( W_{s_i,s_j} \).

We have applied a \((10 + 1) – EA\). Then, parameters were set in these values: population size \( n_p = 10 \), offspring size \( n_o = 1 \). The condition of termination, i.e. deadline \( t_{max} \) was set to 1800 s. Parameter \( \gamma \) of minimum flow connectivity or functional neighbourhood was set to 5. For the rest of parameters to include self-containment as well as minimum size constraints, (6) and (7); the values were in this case taken from the official method applied in the UK [2] for the delineation of the official sets of Travel-to-Work Areas (TTWAs), i.e. \( \beta_1 = 0.7, \beta_2 = 0.75, \beta_3 = 20000 \) and \( \beta_4 = 3500.0 \).

In our first approach [7], the probability of application of each one of the operators was equal for all of them, that is \( \xi = \frac{1.0}{7} = \frac{1}{7} \), and they stay fixed throughout all the evolution.

Extensive experimentation showed that the degree of “success” of the diverse operators varies notably. The frequency of their use depends on the specific form of the fitness function used and on the algorithm parameters, and we have also tested that their relative success within one solution is also affected by the territory over which the procedure is applied. The experimentation stage reveals that the final solution is reached departing from the original individual through 430.56 operations on average. Studies reveal that the number of operations that on average. Most of these operations are recombinations, reassignments of groups of BSUs and divisions (Table 1). In terms of the stage in which the contribution of the diverse types of operators allows the improvement of the best individual, division operators are more successful. However, once the solution approaches the maximum number of possible markets, improvements are
basically produced by recombinations and reassignment of BSUs.

Similar results are obtained when monitoring which operators generate good individuals that remain between generations (Figure 2).

<table>
<thead>
<tr>
<th>Type of operator</th>
<th>Appearances</th>
<th>Appearances by interval</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>0-20%</td>
</tr>
<tr>
<td>Recombinations</td>
<td>42%</td>
<td>35%</td>
</tr>
<tr>
<td>Divisions</td>
<td>15%</td>
<td>18%</td>
</tr>
<tr>
<td>Fusions</td>
<td>1%</td>
<td>1%</td>
</tr>
<tr>
<td>Reassignments of BSUs</td>
<td>10%</td>
<td>9%</td>
</tr>
<tr>
<td>Reassignments of groups of BSUs</td>
<td>32%</td>
<td>35%</td>
</tr>
</tbody>
</table>

We have also measured the contribution of each operator in terms of the average improvement in the fitness function when the best individual of the population changes (Table II) as a result of its action. Division operators produce high increases in the evaluation because the increase in the number of markets is directly reflected in the proposed fitness function (3). As the evolution process takes place the magnitude of the improvements declines as only slight readjustments in the regionalization occur.

Table III shows the time cost of applying each of the operators. Recombination operators as well as the standard mutation (Reassignment1) are the fastest ones. Introduction of knowledge in the rest of the other mutation operators makes them expensive in terms of temporal complexity. However, their specific contribution in order to reach the best solution, by significantly improving the fitness function, balances this higher temporal cost.

In the last stage of the evolution, once the territory has been broadly divided, experimentation stage shows that the operators of recombination perform a local search in the space of solutions, contributing with small but repeated improvements in the value of the fitness function. On the other hand, mutations for the reassignment of individual BSUs -or groups of them- result in local searches as well as in diversification which allow to escape from local minima. These are the reasons why we consider all of the operators necessary, despite the fact that some of them hardly ever yield valid solutions.

### IV. Adaptive Proposals

Each one of the operators has a different influence in the evolutionary process to obtain the final solution. There are differences in the stage of greater success, in the improvement obtained and in their temporal cost. For that reason, the equivalent and/or static probability of application of each one of the operators does not seem to be the suitable way. Therefore, we have considered to adapt these probabilities throughout the evolution process.

The dynamic adjustment of parameters has been broadly used when a huge number of settings should be performed in order to avoid the drawbacks of static preestablished values [11].
In this paper, we have studied different possibilities to adequately the probability vector $\xi$ of application of the operators in realtime:

- Reward to the operator that generates an individual. This reward is different depending on if it improves its parents or the best or the last individuals of the population.
- Reward based on the improvements in the fitness obtained when applying an operator.
- Reward based on improvements in the fitness with respect to the time consumed in generating the offspring.

A. Fixed reward

The first alternative is a self-adaptive method because the offspring inherits the vector of probabilities of its parent, when it is generated by mutation, or the best of its parents, when it comes from recombination. So,

$$\xi_{\text{offspring},i}^{t+1} = \xi_{\text{parent},i}^t, \forall i \in [1, O]$$

Then, the probability of application of the operator $i$ that has generated the offspring is rewarded based on the place that the new individual takes in the population:

- $\xi_{\text{offspring},i}^{t+1} = \xi_{\text{offspring},i}^t + 0.15$, if the fitness of the offspring is greater than that of the best individual of the population.
- $\xi_{\text{offspring},i}^{t+1} = \xi_{\text{offspring},i}^t + 0.05$, if its fitness is greater than that of its parent if it has been generated by mutation or that of both parents if it comes from a recombination.
- $\xi_{\text{offspring},i}^{t+1} = \xi_{\text{offspring},i}^t + 0.025$, if its fitness is greater than that of the worse individual in the population, which gives it possibilities of surviving for the next generation.

This process of reward would not allow that the operators whose probability of application is 0 in some instant of time could be selected and, therefore, to obtain some reward to increase its probability. Therefore, whenever an operator is applied, we introduce a small random modification in all the components of the vectors $\xi$ of all the individuals in the population. That is, $\xi_{p,i} = \xi_{p,i} + \text{noise}, \forall i \in [1, O], \forall p \in [1, n_p]$, where $\text{noise}$ takes a random value between -0.01 and 0.01.

Once these updates are carried out, the values are normalized so as the sum of all the probabilities is 1:

$$\xi_i = \frac{\xi_{\text{offspring},i}^t}{\sum_{j=1}^{O} \xi_{\text{offspring},j}^t}$$

B. Reward depending on the accumulated improvement

This method maintains just one vector of probabilities common to all the population. The purpose of this alternative is to increase in a greater extent the use of those operators allowing to reach offspring that is better than its parents. For it, let $\theta^t = (\theta_i)_{i=1,O}$ be a vector that stores the accumulated improvement by each one of the operators. The procedure of adjustment of the vector of probabilities is as follows:

- In the beginning of the evolution, set all the components of the vector of improvements to 0: $\theta_i^0 = 0, \forall i \in [1, O]$.
- If after applying operator $i$, $f(\text{offspring}) > f(\text{parent})$, then $\theta_i^{t+1} = \theta_i^t + [f(\text{offspring}) - f(\text{parent})]$.
- In order to avoid that an operator that had obtained in the past huge improvements but that no longer should be used, their values are decreased by a factor $\epsilon$; $\theta_i = \epsilon \cdot \theta_i, \forall i \in [1, O]$. We have taken $\epsilon = 0.99$.
- The vector of probabilities is updated depending on the vector of accumulated improvements:

$$\xi_{i}^{t+1} = \frac{\theta_i}{\sum_{j=1}^{O} \theta_j}$$

- Add noise, $\xi_i = \xi_i + \text{noise}, \forall i \in [1, O]$, where $\text{noise}$ takes a random value between -0.01 and 0.01.
- Normalize vector $\xi$ so as the sum of probabilities is equal to 1.

C. Reward depending on the accumulated improvement per time

A question that arises when assigning a larger weight to the operators that allow reaching larger improvements is the fact that such improvements may involve obtaining the offspring in a longer lapse of time. This is the reason why, similarly to the previous method, we have a vector $\chi_i^t = (\chi_{i,j})_{j=1,O}$ to store the accumulated improvements. The procedure to update the vector of probabilities is:

- In the beginning of the evolution, all the components of the vector of improvements are set to 0: $\chi_i^0 = 0, \forall i \in [1, O]$.
- If after applying operator $i$, $f(\text{offspring}) > f(\text{parent})$, then $\chi_i^{t+1} = \chi_i^t + \frac{f(\text{offspring}) - f(\text{parent})}{\text{time to calculate it}}$.
- $\chi_i = 0.99 \cdot \chi_i, \forall i \in [1, O]$.
- Vector of probabilities is updated depending on the vector of improvements:

$$\xi_i = \frac{\chi_i}{\sum_{j=1}^{O} \chi_j}$$

- Add noise, $\xi_i = \xi_i + \text{noise}, \forall i \in [1, O]$, where $\text{noise}$ takes a random value between -0.01 and 0.01.
- Normalize vector $\xi$.

V. RESULTS

An exhaustive study of the application of the different adaptive proposals has been made. We have performed 50 tests for each alternative using as case study the problem of division of the Region of Valencia into local labour markets and employing the same values for the parameters of the algorithm than in section III.E ($n_p = 10$, $n_o = 1$,}
Fig. 3. Probability of application of each operator

$t_{max} = 1800s$. Table IV shows how the average results of all the adaptive approaches are better than those resulting from the original EA and, of course, much more better than the ones obtained with the official procedure used by the public administrations of many countries. In addition, variance does not rise in these best results.

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Fitness value</th>
<th>Number of LLMAs</th>
</tr>
</thead>
<tbody>
<tr>
<td>TTWAs method</td>
<td>120.23</td>
<td>44</td>
</tr>
<tr>
<td>Original EA [7]</td>
<td>Best 192.02</td>
<td>63</td>
</tr>
<tr>
<td></td>
<td>Mean 183.27</td>
<td>59.84</td>
</tr>
<tr>
<td></td>
<td>$\sigma$ 3.70</td>
<td>1.81</td>
</tr>
<tr>
<td>Self-Adaptive EA (fixed reward)</td>
<td>Best 190.78</td>
<td>62</td>
</tr>
<tr>
<td></td>
<td>Mean 184.58</td>
<td>60.22</td>
</tr>
<tr>
<td></td>
<td>$\sigma$ 3.69</td>
<td>1.67</td>
</tr>
<tr>
<td>Adaptive EA (reward by improvement)</td>
<td>Best 191.68</td>
<td>63</td>
</tr>
<tr>
<td></td>
<td>Mean 184.66</td>
<td>60.38</td>
</tr>
<tr>
<td></td>
<td>$\sigma$ 3.69</td>
<td>1.70</td>
</tr>
<tr>
<td>Adaptive EA (reward by (improvement/time)</td>
<td>Best 190.48</td>
<td>62</td>
</tr>
<tr>
<td></td>
<td>Mean 184.73</td>
<td>60.44</td>
</tr>
<tr>
<td></td>
<td>$\sigma$ 3.70</td>
<td>1.72</td>
</tr>
</tbody>
</table>

Figure 3 shows the probability of application of each type of operators based on the number of generations when using an adaptive method (reward by improvement/time). In the beginning the most used operators are the divisions. Later, a finer delineation is obtained by applying recombinations as well as reassignments of BSUs and groups of BSUs. The probabilities of the different operators in each time are very similar to the data of success of operators when the probabilities are static (figure 2).

At any point of time, this fact causes the probability of obtaining good individuals and proceeding in the evolution being larger, since the algorithm gives greater importance to most successful operators. This is clear in figure 4 where a comparison between the original proposal's fitness and that of the adaptive alternative (reward by improvement/time) are compared. The adaptive method converges faster to good results. In fact, figure 5 shows great differences in the best individual when only 10 seconds of evolution have elapsed. The adaptive proposal offers a result close to the final one, whereas the original algorithm has still not arrived to a sufficient division of the territory.

VI. CONCLUSIONS

We have presented several adaptive versions of our previous evolutionary algorithm for delimitation of functional areas (an unsupervised multi-constrained graph partitioning problem), and compared both results.

Given the complexity of the problem, when the requirements associated to such a procedure are applied in real case studies (where the number of base spatial units is frequently very high), conventional genetic operators hardly ever lead to valid solutions. We tried to avoid this problem by designing ad-hoc operators. These specialized operators allowed to obtain good final delineations.

From exhaustive studies we have issued that the behavior of each operator changes throughout the evolution in terms of success to improve fitness, quality if success and time consumed to generate the offspring. For this reason we have considered the application of adaptive approaches to perform a realtime update of the probabilities of application of each operator.

Different rewards have been proposed in order to give more weight to the operators that can produce a better improvement in the population. We have tested these methods to the delineation of the Region of Valencia, achieving improvements over the original evolutionary algorithm.

Although it is not presented in this paper, these adaptive methods behave correctly when they are applied to other territories or when other fitness functions are used, without having to make a fine-tuning of the parameter and without determining what operators must be applied in each case.

ACKNOWLEDGMENT

José M. Casado-Díaz has received financial support from the Spanish Ministry of Education and Science (Ministerio de Educación y Ciencia) for the project SEI2007-67767-C04-02/GEOG "Local Labour Markets. New Methods of Delineation and Analysis" (Spanish National Plan of R+D+i). Lucas Martínez-Bernabeu acknowledges financial support from the Spanish Ministry of Education and Science, the European Social Fund (ESF) and the University of Alicante.
Fig. 5. Delineation with the original EA [left] and an adaptive EA (reward by improvement/time) [right] when evolution has recently started

REFERENCES


