Regional vs. Global Finite-State Error Repair*

M. Vilares[1], J. Otero[1], and J. Graña[2]

[1] Department of Computer Science, University of Vigo
Campus As Lagoas s/n, 32004 Ourense, Spain
{vilares,jop}@uvigo.es
[2] Department of Computer Science, University of A Coruña
Campus de Elviña s/n, 15071 A Coruña, Spain
grana@udc.es

Abstract. We focus on the domain of a regional least-cost strategy in order to illustrate the viability of non-global repair models over finite-state architectures. Our interest is justified by the difficulty, shared by all repair proposals, to determine how far to validate. A short validation may fail to gather sufficient information, and in a long one most of the effort can be wasted. The goal is to prove that our approach can provide, in practice, a performance and quality comparable to that attained by global criteria, with a significant saving in time and space. To the best of our knowledge, this is the first discussion of its kind.

1 Introduction

A classic problem in error repair is how far into the string to validate the process. Given that it is not possible to ensure that the correction and the programmer's intention are the same, the goal is to find the least-cost one. This can only be judged in the context of the entire input, and global methods [4, 5] are not necessarily the best option, due to their inefficiency, but are the most commonly used and for this reason considered to be the most appropriate. An alternative consists of examining the non-global context and attempting to validate repairs by tentatively recognizing ahead, following a successful approach on context-free grammars (CFGs) [7].

In this sense, although all proposals on error repair in the Chomsky's hierarchy are guided by some kind of linguistic data, whether grammar or automaton-based, each level strongly conditions the strategy to follow. So, requests on regular grammars (RGs) are different from those dealing with CFGs [8], where parses are not usually performed in depth, but breadth-wise; whilst the number of states in the associated push-down automaton is often small in practice. Our proposal takes this into account by limiting the search space associated to the repair. We explore the alternatives according to the topology of the corresponding finite automaton (FA). This allows us to restrict

* Research partially by the Spanish Government under projects TIN2004-07246-C03-01, TIN2004-07246-C03-02 and HP2002-0081, and the Autonomous Government of Galicia under projects PGIDIT03SIN30501PR and PGIDIT02SIN01E.

the error hypotheses to areas close to the point where the standard recognizer comes to a halt, which translates into a significant reduction in time and space costs in relation to global approaches.

2 The operational model

Our aim is to parse a word $w_{1..n} = w_1 \dots w_n$ according to an RG $\mathcal{G} = (N, \Sigma, P, S)$. We denote by w_0 (resp. w_{n+1}) the position in the string, $w_{1..n}$, previous to w_1 (resp. following w_n). We generate from $\mathcal{G}$ a *numbered minimal acyclic finite automaton* for the language $\mathcal{L}(\mathcal{G})$. In practice, we choose a device [3] generated by GALENA [2]. A *finite automaton* (FA) is a 5-tuple $\mathcal{A} = (\mathcal{Q}, \Sigma, \delta, q_0, \mathcal{Q}_f)$ where: $\mathcal{Q}$ is the set of states, Σ the set of input symbols, δ is a function of $\mathcal{Q} \times \Sigma$ into $2^{\mathcal{Q}}$ defining the transitions of the automaton, q_0 the initial state and $\mathcal{Q}_f$ the set of final states. We denote $\delta(q, a)$ by $q.a$, and we say that $\mathcal{A}$ is *deterministic* iff $\mid q.a \mid \leq 1,\ \forall q \in \mathcal{Q},\ a \in \Sigma$. The notation is transitive, $q.w_{1..n}$ denotes the state $(\overset{n-2}{\dots}(q.w_1)\overset{n-2}{\dots}).w_n$. As a consequence, w is *accepted* iff $q_0.w \in \mathcal{Q}_f$, that is, the *language accepted by* $\mathcal{A}$ is defined as $\mathcal{L}(\mathcal{A}) = \{w,$ such that $q_0.w \in \mathcal{Q}_f\}$. An FA is *acyclic* when the underlying graph is. We define a *path in the* FA as a sequence of states $\{q_1, \dots, q_n\}$, such that $\forall i \in \{1, \dots, n-1\},\ \exists a_i \in \Sigma,\ q_i.a_i = q_{i+1}$.

In order to reduce the memory requirements, we apply a minimization process [1]. In this sense, we say that two FA's are *equivalent* iff they recognize the same language. Two states, p and q, are *equivalent* iff the FA with p as initial state, and the one that starts in q recognize the same language. An FA is *minimal* iff no pair in $\mathcal{Q}$ is equivalent. Although the standard recognition is deterministic, the repair one could introduce non-determinism by exploring alternatives associated to possibly more than one recovery strategy. So, in order to get polynomial complexity, we avoid duplicating intermediate computations in the repair of $w_{1..n} \in \Sigma^+$, storing them in a table $\mathcal{I}$ of *items*, $\mathcal{I} = \{[q, i],\ q \in \mathcal{Q},\ i \in [1, n+1]\}$, where $[q, i]$ looks for the suffix $w_{i..n}$ to be analyzed from $q \in \mathcal{Q}$.

We describe our work using *parsing schemata* [6], a triplet $\langle \mathcal{I}, \mathcal{H}, \mathcal{D} \rangle$, with $\mathcal{H} = \{[a, i],\ a = w_i\}$ a set of items called *hypothesis* that encodes the word to be recognized[1], and $\mathcal{D}$ a set of *deduction steps* that allow to items to be derived from previous ones. These are of the form $\{\eta_1, \dots, \eta_k \vdash \xi \,/\, conds\}$, meaning that if all antecedents η_i are present and the conditions *conds* are satisfied, then the consequent ξ is generated. In our case, $\mathcal{D} = \mathcal{D}^{\text{Init}} \cup \mathcal{D}^{\text{Shift}}$, where:

$$\mathcal{D}^{\text{Init}} = \{\vdash [q_0, 1]\} \qquad \mathcal{D}^{\text{Shift}} = \{[p, i] \vdash [q, i+1] \,/\, \exists [a, i] \in \mathcal{H},\ q = p.a\}$$

The recognition associates a set of items S_p^w, called *itemset*, to each $p \in \mathcal{Q}$; and applies these deduction steps until no new application is possible. The word is recognized iff a *final item* $[q_f, n+1]$, $q_f \in \mathcal{Q}_f$ has been generated. We can assume, without loss of generality, that $\mathcal{Q}_f = \{q_f\}$, and that there is only one transition from (resp. to) q_0 (resp. q_f). To get this, it is sufficient to augment the original FA with two states becoming the new initial and final states, and

[1] A word $w_{1\dots n} \in \Sigma^+$, $n \geq 1$ is represented by $\{[w_1, 1],\ [w_2, 2],\ \dots,\ [w_n, n]\}$.

linked to the original ones through empty transitions, our only concession to the notion of minimal FA.

3 The error repair frame

Let us assume that we are dealing with the first error in a word $w_{1..n} \in \Sigma^+$. We extend the item structure, $[p, i, e]$, where now e is the error counter accumulated in the recognition of w at position w_i in state p. We talk about the *point of error*, w_i, as the point at which the difference between what was intended and what actually appears in the word occurs, that is, $q_0.w_{1..i-1} = q$ and $q.w_i \notin \mathcal{Q}$. The next step is to locate the origin of the error, limiting the impact on the analyzed prefix to the context close to the point of error, in order to save computational effort. To do so, we introduce some topological properties. Since we work with acyclic FAs, we can introduce a simple order in $\mathcal{Q}$ by defining $p < q$ iff there exists a path $\rho = \{p, \ldots, q\}$; and we say that q_s (resp. q_d) is a *source* (resp. *drain*) for ρ iff $\exists a \in \Sigma,\ q_s.a = p$ (resp. $q.a = q_d$). In this manner, the pair (q_s, q_d) defines a *region* $\mathcal{R}_{q_s}^{q_d}$ iff $\forall \rho$, source$(\rho) = q_s$, we have that drain$(\rho) = q_d$ and $\mid \{\forall \rho,\ \text{source}(\rho) = q_s\} \mid > 1$. So, we can talk about *paths*$(\mathcal{R}_{q_s}^{q_d})$ to refer to the set $\{\rho/\text{source}(\rho) = q_s,\ \text{drain}(\rho) = q_d\}$ and, given $q \in \mathcal{Q}$, we say that $q \in \mathcal{R}_{q_s}^{q_d}$ iff $\exists \rho \in \text{paths}(\mathcal{R}_{q_s}^{q_d}),\ q \in \rho$. We also consider $\mathcal{A}$ as a global region. So, any state, with the exception of q_0 and q_f, is included in a region.

This provides a criterion to place around a state in the underlying graph a zone for which any change applied on it has no effect on its context. So, we say that $\mathcal{R}_{q_s}^{q_d}$ is the *minimal region in* $\mathcal{A}$ *containing* $p \in \mathcal{Q}$ iff it verifies that $q_s \geq p_s$ (resp. $q_d \leq p_d$), $\forall \mathcal{R}_{p_s}^{p_d} \ni p$, and we denote it as $\mathcal{M}(p)$.

We are now ready to characterize the point at which the recognizer detects that there is an error and calls the repair algorithm. We say that w_i is *point of detection* associated to a point of error w_j iff $\exists q_d > q_0.w_{1..j},\ \mathcal{M}(q_0.w_{1..j}) = \mathcal{R}_{q_0.w_{1..i}}^{q_d}$, that we denote by *detection*$(w_j) = w_i$. We then talk about $\mathcal{R}_{q_0.w_{1..i}}^{q_d}$ as the *region defining the point of detection* w_i.

The error is located in the left recognition context, given by the closest source. However, we also need to locate it from an operational viewpoint, as an item in the process. We say that $[q, j] \in S_q^w$ is an *error item* iff $q_0.w_{j-1} = q$; and we say that $[p, i] \in S_p^w$ is a *detection item* associated to w_j iff $q_0.w_{i-1} = p$.

Once we have identified the beginning of the repair region, we introduce a *modification* to $w_{1..n} \in \Sigma^+$, $M(w)$, as a series of edit operations, $\{E_i\}_{i=1}^n$, in which each E_i is applied to w_i and possibly consists of a sequence of insertions before w_i, replacement or deletion of w_i, or transposition with w_{i+1}.

This topological structure can be used to restrict the notion of modification, looking for conditions that guarantee the ability to recover the error. So, given $x_{1..m}$ a prefix in $\mathcal{L}(\mathcal{A})$, and $w \in \Sigma^+$, such that xw is not a prefix in $\mathcal{L}(\mathcal{A})$, we define a *repair of* w *following* x as $M(w)$, so that:

(1) $\mathcal{M}(q_0.x_{1..m}) = \mathcal{R}_{q_s}^{q_d}$ (the minimal region including the point of error, $x_{1..m}$)
(2) $\exists \{q_0.x_{1..i} = q_s.x_i, \ldots, q_s.x_{i..m}.M(w)\} \in \text{paths}(\mathcal{R}_{q_s}^{q_d})$

denoted by *repair*(x, w), and $\mathcal{R}_{q_s}^{q_d}$ by *scope*(M). We now organize the concept around a point of error, $y_i \in y_{1..n}$, in order to take into account all possible repairs, by defining the *set of repairs for* y_i, as *repair*$(y_i) = \{xM(w) \in \text{repair}(x, w)/w_1 = \text{detection}(y_i)\}$. Next, we focus on filtering out undesirable repairs, introducing criteria to select minimal costs. For each $a, b \in \Sigma$ we assume insert, $I(a)$; delete, $D(a)$, replace, $R(a, b)$, and transpose, $T(a, b)$, costs. The *cost of a modification* $M(w_{1..n})$ is given by *cost*$(M(w_{1..n})) = \Sigma_{j \in J_\dashv} I(a_j) + \Sigma_{i=1}^{n}(\Sigma_{j \in J_i} I(a_j) + D(w_i) + R(w_i, b) + T(w_i, w_{i+1}))$, where $\{a_j,\ j \in J_i\}$ is the set of insertions applied before w_i; $w_{n+1} = \dashv$ the end of the input and $T_{w_n, \dashv} = 0$. So, we define the set of *regional repairs* for $y_i \in y_{1..n}$, a point of error, as

$$\text{regional}(y_i) = \{xM(w) \in \text{repair}(y_i) \Big/ \begin{matrix} \text{cost}(M) \leq \text{cost}(M'),\ \forall M' \in \text{repair}(x, w) \\ \text{cost}(M) = \min_{L \in \text{repair}(y_i)}\{\text{cost}(L)\} \end{matrix} \}$$

Before dealing with cascaded errors, precipitated by previous erroneous repairs, it is necessary to establish the relationship between recovery processes. So, given w_i and w_j points of error, $j > i$, we define the set of *viable repairs* for w_i in w_j as $\text{viable}(w_i, w_j) = \{xM(y) \in \text{regional}(w_i)/xM(y) \ldots w_j$ prefix for $\mathcal{L}(\mathcal{A})\}$. Repairs in *viable*$(w_i, w_j)$ are the only ones capable of ensuring the recognition in $w_{i..j}$ and, therefore, the only ones possible at the origin of cascaded errors. In this sense, we say that a point of error $w_k,\ k > j$ is a *point of error precipitated by* w_j iff $\forall xM(y) \in \text{viable}(w_j, w_k),\ \exists \mathcal{R}_{q_0.w_{1..i}}^{q_d}$ defining $w_i = \text{detection}(w_j)$, such that $\text{scope}(M) \subset \mathcal{R}_{q_0.w_{1..i}}^{q_d}$. This implies that w_k is precipitated by w_j when the region defining the point of detection for w_k summarizes all viable repairs for w_j in w_k. That is, the information compiled from those repairs has not been sufficient to give continuity to a process locating the new error in a region containing the preceding ones and therefore depending on these. We then conclude that the origin of the current error could be a wrong study of previous ones.

4 The algorithm

Although most authors appeal to global methods to avoid distortions due to unsafe error location [4, 5], our proposal applies a dynamic estimation of the repair region, guided by the linguistic knowledge present in the underlying FA. Formally, we extend the item structure, $[p, i, e]$, where now e is the error counter accumulated in the recognition of w at position w_i in state p. Once the point of error has been located, we apply all possible transitions beginning at both, the point of error and the corresponding point of detection, which corresponds to the following deduction steps in error mode, $\mathcal{D}_{\text{error}} = \mathcal{D}_{\text{error}}^{\text{Shift}} \cup \mathcal{D}_{\text{error}}^{\text{Insert}} \cup \mathcal{D}_{\text{error}}^{\text{Delete}} \cup \mathcal{D}_{\text{error}}^{\text{Replace}} \cup \mathcal{D}_{\text{error}}^{\text{Transpose}}$:

$$\mathcal{D}_{\text{error}}^{\text{Shift}} = \{[p, i, e] \vdash [q, i+1, e],\ \exists [a, i] \in \mathcal{H},\ q = p.a\}$$
$$\mathcal{D}_{\text{error}}^{\text{Insert}} = \{[p, i, e] \vdash [p, i+1, e + I(a)]\}$$
$$\mathcal{D}_{\text{error}}^{\text{Delete}} = \{[p, i, e] \vdash [q, i, e + D(w_i)] \Big/ \begin{matrix} \mathcal{M}(q_0.w_{1..j}) = \mathcal{R}_{q_s}^{q_d} \\ p.w_i = q_d \in \mathcal{R}_{q_s}^{q_d} \textbf{ or } q = q_d \end{matrix} \}$$

$$\mathcal{D}_{\text{error}}^{\text{Replace}} = \{[p,i,e] \vdash [q,i+1,e+R(w_i,a)], \Big/ \begin{array}{l}\mathcal{M}(q_0.w_{1..j}) = \mathcal{R}_{q_s}^{q_d} \\ p.a = q \in \mathcal{R}_{q_s}^{q_d} \textbf{ or } q = q_d\end{array} \}$$

$$\mathcal{D}_{\text{error}}^{\text{Transpose}} = \{[p,i,e] \vdash [q,i+2,e+T(w_i,w_{i+1})] \Big/ \begin{array}{l}\mathcal{M}(q_0.w_{1..j}) = \mathcal{R}_{q_s}^{q_d} \\ p.w_i.w_{i+1} = q \in \mathcal{R}_{q_s}^{q_d} \textbf{ or } q = q_d\end{array} \}$$

where $w_{1..j}$ looks for the current point of error. Observe that, in any case, the error hypotheses apply on transitions behind the repair region. The process continues until a repair covers the repair region.

When dealing with an error which is not the first one in the word, it could condition a previous repair. This arises when we realize that we come back to a detection item for which some recognition branch includes a previous recovery process. The algorithm re-takes the error counters, adding the cost of new error hypotheses to profit from the experience gained from previous repairs. This permits us to deduce that if w_l is a point of error precipitated by w_k, then:

$$q_0.w_{1..i} < q_0.w_{1..j},\ \mathcal{M}(q_0.w_l) = \mathcal{R}_{q_0.w_{1..i}}^{q_d},\ w_j = y_1,\ xM(y) \in \text{viable}(w_k, w_l)$$

which proves that the state associated to the point of detection in a cascaded error is strictly smaller than the one associated to the source of the scope in the repairs precipitating it. So, the minimal possible scope of a repair for the cascaded error includes any scope of the previous ones, that is,

$$\max\{\mathit{scope}(M),\ M \in \text{viable}(w_k, w_l)\} \subset \max\{\mathit{scope}(\tilde{M}),\ \tilde{M} \in \text{regional}(w_l)\}$$

This allows us to get an asymptotic behavior close to that obtained by global repair methods, and with a comparable quality, but in practice at the cost of a local one.

5 Asymptotic behavior

Our aim now is to validate the practical interest of our proposal in relation to classic global ones, putting into evidence the theoretical results previously advanced. We think that it is an objective criterion to measure the quality of a repair algorithm, since the point of reference is a technique that guarantees the best quality for a given error metric when all contextual information is available.

5.1 The running language

We choose to work with a lexicon for Spanish built from GALENA [2], which includes 514,781 different words, to illustrate this aspect. The lexicon is recognized by an FA containing 58,170 states connected by 153,599 transitions, of sufficient size to allow us to consider this automaton as a representative starting point for our purposes. Although Spanish is a non-agglutinative language, it shows a great variety of morphological processes, making it adequate for our description. The most outstanding features are to be found in verbs, with a

highly complex conjugation paradigm, including nine simple tenses and nine compound tenses, all of which have six different persons. If we add the present imperative with two forms, the infinitive, the compound infinitive, the gerund, the compound gerund, and the participle with four forms, then 118 inflected forms are possible for each verb. In addition, irregularities are present in both stems and endings. So, very common verbs, such as `hacer` (*to do*), have up to seven different stems: `hac-er`, `hag-o`, `hic-e`, `har-é`, `hiz-o`, `haz`, `hech-o`. Approximately 30% of Spanish verbs are irregular, and can be grouped around 38 different models. Verbs also include enclitic pronouns producing changes in the stem due to the presence of accents: `da` (*give*), `dame` (*give me*), `dámelo` (*give it to me*). We have considered forms with up to three enclitic pronouns, like `tráetemelo` (*bring it for you and me*). There exist some highly irregular verbs that cannot be classified in any irregular model, such as `ir` (*to go*) or `ser` (*to be*); and others include gaps in which some forms are missing or simply not used. For instance, meteorological verbs such as `nevar` (*to snow*) are conjugated only in third person singular. Finally, verbs can present duplicate past participles, like `impreso` and `imprimido` (*printed*).

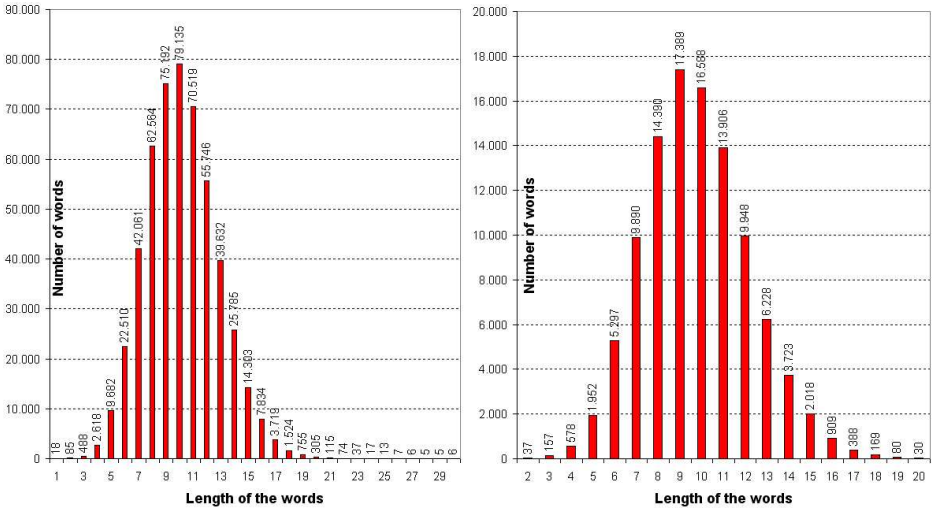

Fig. 1. Statistics on the general and error lexicons.

This complexity extends to gender inflection, with words considering only one gender, such as `hombre` (*man*) and `mujer` (*woman*), and words with the same form for both genders, such as `azul` (*blue*). In relation to words with separate forms for masculine and feminine, we have a lot of models: `autor`, `autora` (*author*); `jefe`, `jefa` (*boss*); `poeta`, `poetisa` (*poet*); `rey`, `reina` (*king*) or `actor`, `actriz` (*actor*). We have considered 20 variation groups for gender. We can also refer to number inflection, with words presenting only the singular

form, as `estrés` (*stress*), and others where only the plural form is correct, as `matemáticas` (*mathematics*). The construction of different forms does not involve as many variants as in the case of gender, but we can also consider a certain number of models: `rojo`, `rojos` (*red*); `luz`, `luces` (*light*); `lord`, `lores` (*lord*) or `frac`, `fraques` (*dress coat*). We have considered 10 variation groups for number.

5.2 The operational testing frame

From this lexicon, we select a representative sample of morphological errors for practical evaluation. This can be verified from Fig. 1, which shows the equitative distribution of both the original lexicon and the running sample, in terms of lengths of the words dealt with. For each length-category, errors have been randomly generated in a number and position for the first error in the input string as is shown in Fig. 3. This is of some importance since, as the authors claim, the performance of previous proposals depend on these factors, which makes no practical sense. No other dependencies, for example in terms of lexical categories, have been detected at morphological level and, therefore, they have not been considered.

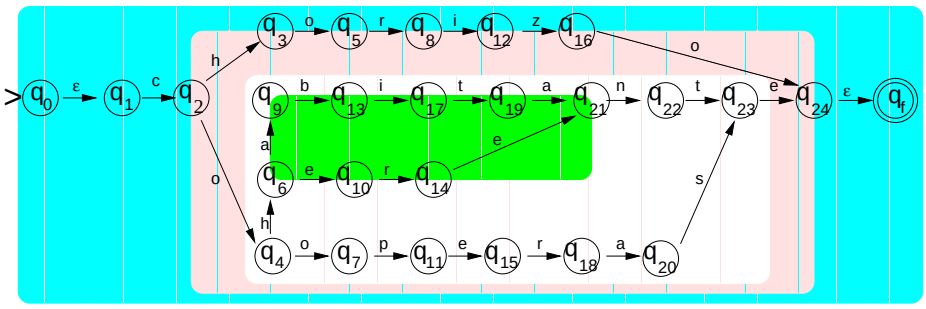

Fig. 2. The concept of region in error repair.

In this context, our testing framework seems to be well balanced, from both an operational and linguistic viewpoint, in order to estimate the practical performance of error repair algorithms on FA architectures. It only remains to decide which repair algorithms will be tested. We choose to compare our proposal with the Savary's global approach [5], an evolution of the Oflazer's algorithm [4] and, to the best of our knowledge, the most efficient method of error-tolerant look-up in finite-state dictionaries. The comparison has been made from three complementary viewpoints: the size of the repair region considered, the computational cost and the quality achieved. We consider the editing distance [5] as error metric, the same proposed by Savary.

5.3 The error repair region

We focus on the evolution of this region in relation to the location of the point of error, in opposition to static strategies associated to global repair approaches. To illustrate it, we take as running example the FA represented in Fig. 2, which recognizes the following words in Spanish: *"chorizo"* (sausage) , *"cohabitante"* (a person who cohabits with another one) , *"coherente"* (coherent) and *"cooperase"* (I cooperated). We consider as input string the erroneous one *"coharizo"*, resulting from transposing *"h"* with *"o"* in *"chorizo"* (sausage), and later inserting the character *"a"*. We shall describe the behavior from both viewpoints, the Savary's [5] algorithm and our proposal, proving that in the worst case, when precipitated errors are present, our proposal can retake the repair process in order to recover the system from cascaded errors.

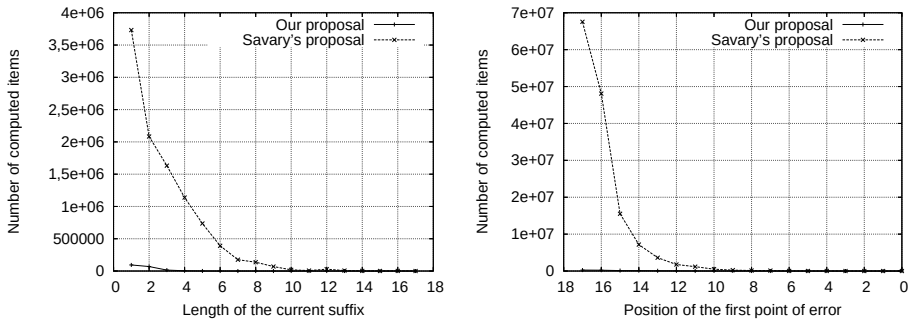

Fig. 3. Number of items generated in error mode.

In this context, the recognition comes to an halt on state q_9, for which $\mathcal{M}(q_9) = \mathcal{R}_{q_6}^{q_{21}}$ and no transition is possible on *"r"*. So, our approach locates the error at q_6 and applies from it the error hypotheses looking for the minor editing distance in a repair allowing the state q_{21} to be reached. In this case, there are two possible regional repairs consisting in first replacing *"a"* by *"e"* and later inserting an *"e"* after *"r"* (resp. replace *"i"* by *"e"*), to obtain the modification on the entire input string *"coherezo"* (resp. *"cohereizo"*), which is not a word in our running language.

As a result, although we return to the standard recognition in q_{21}, the next input character is now *"i"* (resp. *"z"*), for which no transition is possible and we come back to error mode on the region $\mathcal{M}(q_{21}) = \mathcal{R}_{q_4}^{q_{23}}$ including $\mathcal{M}(q_9) = \mathcal{R}_{q_6}^{q_{21}}$. We then interpret that the current error is precipitated by the previous one, possibly in cascade. As result of this new process none of the regional repairs generated allow us to retake the standard recognition beyond the state q_{23}. At this point, $\mathcal{M}(q_{23}) = \mathcal{R}_{q_2}^{q_{24}}$ become the new region, and the only regional repair is now defined as the transposition of the *"h"* with *"o"*, and the deletion of *"a"*; which agrees with the global repair proposed by Savary, although the repair

region is not the total one, as is the case for the latter algorithm. This repair finally allows acceptance by the FA.

The process described demonstrates that we do not need to extend the repair region to the entire FA in order to get the least-cost correction and, secondly, the risk of errors in cascade can be efficiently solved in the context of non-global approaches. Also, in the worst case, our running example illustrates the convergence of our regional strategy towards the global one from both viewpoints, that of computational cost and that of quality of the correction.

5.4 Computational cost

These practical results are compiled in Fig. 3, using the concept of item previously defined as a unit for measuring the computational effort. We here consider two complementary approaches illustrating the dependence on both the position of the first point of error in the word and the length of the suffix from it. So, in any case, we ensure that we take into account the degree of penetration in the FA at that point, which determines the effectiveness of the repair strategy. In effect, working on regional methods, the penetration determines the number of regions in the FA including the point of error and, as a result, the possibility of considering a non-global resolution.

In order to clearly show the detail of the tests on errors located at the end of the word, which is not easy to observe from the decimal scale of Fig. 3, we include in Fig. 4 the same results using a logarithmic scale. So, both graphics perfectly illustrate our contribution, in terms of computational effort saved, from two viewpoints which are of interest in real systems. Firstly, our proposal shows in practice a linear-like behavior, in contrast to the Savary's one, which seems to be of the exponential type. In particular, this translates into an essential property in industrial applications, the independence of the time of response from the initial conditions for the repair process. Secondly, in any case, the number of computations is significantly reduced when we apply our regional criterion.

5.5 Performance

However, statistics on computational cost only provide a partial view of the repair process, which must also take into account data related to the performance from both the user's and the system's viewpoint. In order to get this, we have introduced the following two measures, for a given word, w, containing an error:

$$performance(w) = \frac{useful\ items}{total\ items} \qquad recall(w) = \frac{proposed\ corrections}{total\ corrections}$$

that we complement with a global measure on the *precision* of the error repair approach in each case, that is, the rate reflecting when the algorithm provides the correction needed by the user. We use the term *useful items* to refer to the number of generated items that finally contribute to obtaining a repair, and *total items* to refer to the number of these structures generated during the process.

We denote by *proposed corrections* the number of corrections provided by the algorithm, and by *total corrections* the number of possible ones, absolutely.

These results are shown in Fig. 5, illustrating some interesting aspects in relation with the asymptotic behavior we want to demonstrate in the regional approach. So, considering the running example, the performance in our case is not only better than Savary's, but the difference existing between them also increases with the location of the first point of error. Intuitively this is due to the fact that the closer this point is to the beginning of the word, the greater is the number of useless items generated in error mode, a simple consequence of the higher availability of different repair paths in the FA when we are working in a region close to q_0. In effect, given that the concept of region is associated to the definition of corresponding source and drain points, this implies that this kind of region is often equivalent to the total one since the lay-out of these regions is always concentric. At this point, regional and repair approaches apply the same error hypotheses not only on a same region, but also from nearby states given that, in any case, one of the starting points for these hypotheses would be q_0 or a state close to it. That is, in the worst case, both algorithms converge.

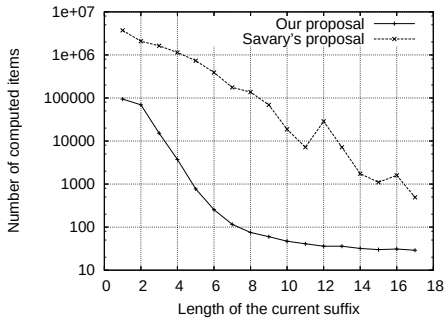

Fig. 4. Number of items generated in error mode. Logarithmic scale.

The same reasoning could be considered in relation to points of error associated to a state in the recognition that is close to q_f, in order to estimate the repair region. However, in this case, the number of items generated is greater in

the case of the global technique, which is due to the fact that the morphology of the language often leads to the generation of regions which concentrate near q_f, a simple consequence of the common derivational mechanisms applied on suffixes defining gender, number or verbal conjugation groups. So, it is possible to find a regional repair by just implying some error hypotheses from the state associated to the point of error or from the associated detection point and, although this regional repair may be different from the global one, its computational cost would usually be lower.

A similar behavior can be observed with respect to the recall relation. Here, Savary's algorithm shows a constant graph since the approach applied is global and consequently the set of corrections provided is always the entire one for a fixed error counter. In our proposal, the results prove that the recall is smaller than that for Savary's, which illustrates the gain in computational efficiency in comparison with the global method. With regard to the convergence between both approaches, we must again search around points of detection close to the beginning of the word, which also often implies repair regions being equivalent to the total one and repairs starting around q_0, as is illustrated in Fig. 5.

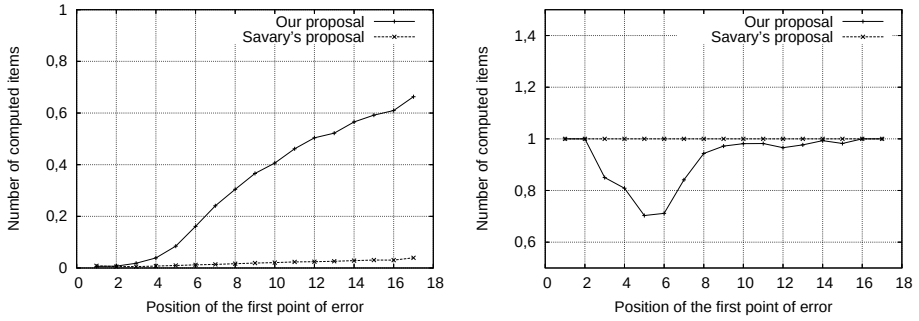

Fig. 5. Performance and recall results.

However, in contrast to the case of performance, it can be seen that for recall the convergence between global and regional proposals also seems to extend to processes where the point of error is associated to states close to q_f, that is, when this point is located near the end of the word. To understand this, it is sufficient to take into account that we are not now computing the number of items generated in the repair, but the number of corrections finally proposed. So, given that the closer to the end of the word we are, the smaller is the number of alternatives for a repair process, both global and regional approaches also converge towards the right of the graph for recall.

Finally, the regional (resp. the global) approach provided as correction the word from which the error was randomly included in 77% (resp. 81%) of the cases. Although this could be interpreted as a justification for using global methods, it is necessary to remember that we are now only taking into account morphological

information, which has an impact on precision for a regional approach, but not for a global one, which always provides all the repair alternatives without exclusion. So, the consideration of the precision concept represents, in the exclusive morphological context considered, a clear disadvantage for our proposal since it bases its efficiency in the limitation of the search space. We expect that the integration of linguistic information from both syntactic and semantic viewpoints will significantly reduce this gap of less than 4% in precision, or may even eliminate it.

6 Conclusion

We have illustrated how a least-cost error repair method can be applied to a finite-state architecture in order to recover the recognition at the point of each error, to avoid the possibility of non-detection of any subsequent errors. So, although the correctness of a symbol can only be judged in the context of the entire string, which can be extremely time-consuming, our proposal minimizes the impact by dynamically graduating the size of the repair zone on the basis of underlying grammatical structure. In this sense, the practical results seem promising, demonstrating as they do a significant reduction in time and space costs with no apparent loss of quality.

References

1. J. Daciuk, S. Mihov, B.W. Watson, and R.E. Watson. Incremental construction of minimal acyclic finite-state automata. *Computational Linguistics*, 26(1):3–16, 2000.
2. J. Graña, F.M. Barcala, and M.A. Alonso. Compilation methods of minimal acyclic automata for large dictionaries. *Lecture Notes in Computer Science*, 2494:135–148, 2002.
3. C.L. Lucchesi and T. Kowaltowski. Applications of finite automata representing large vocabularies. *Software-Practice and Experience*, 23(1):15–30, January 1993.
4. K. Oflazer. Error-tolerant finite-state recognition with applications to morphological analysis and spelling correction. *Computational Linguistics*, 22(1):73–89, 1996.
5. A. Savary. Typographical nearest-neighbor search in a finite-state lexicon and its application to spelling correction. *Lecture Notes in Computer Science*, 2494:251–260, 2001.
6. K. Sikkel. *Parsing Schemata.* PhD thesis, Univ. of Twente, The Netherlands, 1993.
7. M. Vilares, V.M. Darriba, and M.A. Alonso. Searching for asymptotic error repair. *Lecture Notes in Computer Science*, 2608:276–281, 2003.
8. M. Vilares, V.M. Darriba, and F.J. Ribadas. Regional least-cost error repair. *Lecture Notes in Artificial Intelligence*, 2088:293–301, 2001.