Computer Science Journal of Moldova, vol.**18**, no.1(52), 2010

Developing a derivatives generator

Mircea Petic

Abstract

The article intends to highlight the particularities of the derivational morphology mechanisms that will help in lexical resources extension. Some computing approaches for derivational morphology are given for several languages, inclusively for Romanian. This paper deals with some preprocessing particularities, that are needed in the process of automatic generation. Then, generative mechanisms are presented in the form of derivational formal rules separately for prefixation and suffixation. The article ends with several approaches in automatic new generated words validation.

Key-words: derivatives, word generation, lexicon, word validation

1 Introduction

Romanian derivational morphology represents an important issue in Romanian lexical resources extension. To automate the process of derivation it is necessary: to establish rules that can be applied to stems in order to obtain new derivatives; to establish conditions in which these rules can be applied; if these restrictions do not guarantee the correctness of the generated words - to develop and to implement a validation mechanism.

In consideration of premises, this article pretends to highlight the particularities of the derivational morphology mechanisms that will help in lexical resources extension without any semantic information.

In order to understand the differences and similarities of the approaches used in our research, the article starts with a description

of the known methods used in derivational morphology for Romanian and other languages. Then some preprocessing particularities, that are needed in the process of automatic generation, are described, such as the issues connected with derivatives analysis and the particular features of a lexicon for derivatives generations aims, followed by the short description of the vowel and/or consonant alternation. The generative mechanisms are presented in the form of derivational formal rules separately for prefixation and suffixation. The article ends with several approaches in automatic new generated words validation.

2 The derivational process automatization

This section will be a brief overview of automation methods of derivation for different languages. Note that the automation of the derivation process mechanisms can help to solve other problems, such as: generation of morphological families, generation of derivatives with predictable meanings, expanding dictionaries and lexicons, informational retrieval, machine translation, etc [1]. Below, different approaches of the derivational morphology will be described for the following languages: Russian, Italian, Serbian, Arabian, French and Romanian.

Studying the automatization of the derivational process on the examples from different languages, we came to a conclusion that the obvious elements for processing in derivational morphology are vocabulary, lexicon or dictionary. Though it is not the subject of the present compartment. Beside the list of words, there is also another important moment to be discussed. There are two approaches in finding the corresponding derivatives. The first approach provides, that the derivatives are simply described in the lexicons, and being needed they are extracted with the help of some restrictions. The second approach corresponds to generative mechanisms, that form derivatives using constraint rules.

So, the first approach is used in the description of the Italian derivational morphology where generation is fulfilled only with regard to derivatives included into descriptions of all derived words in terms of finite automata. Another example is the system designed for Arabic

morphology involving two kinds of hierarchies: one for morphological forms and the other – for set of rules. In this case all stems and the corresponding information are stored in the lexicon.

The second approach is found in the description of Russian, Serbian and French languages. **RUSLO** (RUsskoe SLOvoobrazovanie) **derivation system**, works with Russian words. It can analyse both present and not present (as the jargon and neologisms and/or slang) in the dictionary words. RUSLO solved the problem of generation and analysis of derivatives for Russian language through detailed generative mechanisms of derivation [2]. In the case of Serbian language derivatives were generated in a predictable way. The derivation is considered predictable if the word changes gender (profesor → profesorka), or enhances meaning, i.e. generates diminutives (profesorčić) and augments (profesorčina), forms relational adjectives (profesorski) and possessive adjectives (profesorov) as well. The last one in fact is not the case of Romanian language. This process was called **regular derivation** [3].

GeDeriF is a system for French, which automatically analyses unknown in dictionary words and overgenerates derivatives. The system uses derivational rules for suffixes *-able*, *-ité* and *-is (-er)*. These generated derivatives had been checked in Encyclopedia Universalis and terminology review Le Banc de Mots, which was drawn from a variety of sources. Besides this, they made a program that automatically checked the search engine www.yahoo.fr for each of the generated terms. Nevertheless the average of the percentage of correct words is very low [4].

One of the first applications of automatic differentiation system for Romanian language was FAVR in the Mac environment ELU which aimed to complete coverage of the inflectional morphology. Then, prefixes and suffixes were described by means of lexical or grammatical paradigms. In this scope 20 grammar categories have been used. This morphologic description was tested on more than 15.000 lexical entries [5].

3 Preprocessing in derivational morphology

As it was emphasized above, the lexicon plays an important role in the process of automatization of the derivational morphology. That is why some particularities concerning lexicons are given below. Another important issues in derivational morphology are the derivatives recognition. In addition the problem of vowel and/or consonant alternations is described by presenting a short picture of the type of alternations with concrete examples.

3.1 Lexicon for derivatives generation

Lexicon represents one of the main elements in the process of new derivatives generation. In this case the lexicon is not simply a repository for input of words with syntactic and semantic information (or lemma level), but also prefixes and suffixes are described in it [6].

Another point of view supposes that lexicons should contain not only dictionaries of simple words and their inflections, but also the dictionary of compound words, and dictionary of the finite-state transducers used to recognise unregistered words in the dictionaries [3].

Although for our purposes the best solution is the Dictionary of derivatives [7] containing only the graphical representation and constituent morphemes without any information about their part of speech, though the vast majority are nouns, verbs and adjectives. Electronic version of the dictionary [7] was obtained after it was scanned, the original input OCRized and the corrections made. This electronic version of the dictionary [7] becomes important as it is difficult to establish criteria for validation of new generated derivatives. In addition, it allows detection of derivatives with the appropriate type morphemes (prefix, root and suffix) and is an important electronic resource for research derivational morphology. Basically, the entries in the dictionary [7] are being built based on an uncertain schedule. In this scheme it is not clear where the affixes and the root are. In order to exclude the uncertainty of the electronic version of the dictionary entries, a regular expression representing the structure of derivatives was developed:

```
derivative = (+ morpheme)*.morpheme(-morpheme)*
```

where `+morpheme` is a prefix, `.morpheme` is a root and `-morpheme` is a suffix. An example of an entry in the lexicon is:

```
antistatal=+anti.stat-al

reprogramabil=+re.programa-bil
```

3.2 Automatic derivatives recognition

The majority of the derivational rules are taking into account the consequence of letters referring to words endings or suffixes. Moreover, it is not a good thing to generate several times derivatives with the same prefix. That is why it is important to have a mechanism for derivatives recognition.

As a source for automatic derivatives recognition, a lexicon serves, containing not only graphic representation of the words, but also their part of speech. The lexicon consists of approximately 100000 of words bases, and words can have several entrances for different parts of speech. Besides the lexicon, lists of prefixes with their phonological forms and suffixes were used.

Since not all the words end (begin) with the same suffixes (prefixes), some algorithms were elaborated for enabling the automatic extraction of the derivatives from the lexicon. The elaborated algorithms took into account the fact that being $x, y \in \Sigma^+$, where Σ^+ is the set of all possible roots, and if $y = xv$ then v is the suffix of y and if $y = ux$ then u is the prefix of y. In this context both y and x must be valid words in Romanian language, and u and v are strings that can be affixes for Romanian language. The problem of consonant and/or vowel alternations was neglected in the case of the algorithm of derivatives extraction. This fact does not permit the exact detecting of all derivatives [8].

Being more precise, the following word formation scheme expresses the particularities of prefixation:

$$[prefix\,[stem]_x]_x$$

where x represents part of speech for stem and derivative. Note that in the process of prefixation the part of speech does not change. In the process of suffixation there are cases of part of speech changing, as it is presented in the following word formation scheme:

$$[[stem]_x\,suffix]_y$$

Taking into consideration the peculiarities of the Romanian affixes and derivatives the algorithm for automatic derivatives recognition was elaborated that lately was implemented in a program written in Java programming language. This program allows us to follow at every step of the algorithm the partial results listed in the corresponding textual files.

3.3 Classification of affixes attachment

We examine some classes of affixes attachment. The situation is that there are more derivatives without alternations, especially in the case of the prefix derivation. The lack of vowel and/or consonant alternations in the process of derivation is observed with the following most frequent prefixes: *ne-*, *re-*, *pre-*, *anti-*, *auto-*, *supra-*, and *de-* [8].

There are cases when affixes do not need vowel and/or consonant alternations in the process of derivation. Below we will present some of these cases. The attachment of the affixes to the words is done by means of:

– addition of a letter to the end of the root, for example, *şurub* → *înşurub**a***, *bold* → *îmbold**i***, *plin* → *împlin**i***;

– deleting of the final letter in the root, for example, *lîn**ă*** → *dezlîn**a***, *purpur**ă*** → *împurpur**a***, *puşc**ă*** → *împuşc**a***;

– changing in the prefix, for example, *şoca* → ***de(s)****şoca* → ***de****şoca*, *pat* → ***su(b)****pat* → ***su****pat*;

– avoiding of the double consonant, for example, *spinteca* → ***de(s)s****pinteca* → ***des****pinteca*, *braţ* → ***su(b)b****raţ* → ***sub****raţ*;

– changing of two final letters in the root, for example, *zeflem**ea*** → *zeflem**itor**, încăp**ea*** → *încăp**ătoare***,

– changing of the final letter in the root, for example, *alini**a*** → *alini**ere**, aşchi**a*** → *aşchi**etor**, cumpăr**a*** → *cumpăr**ător**, curăţ**i*** → *curăţ**ător**, delăs**a*** → *delăs**ător**, depun**e*** → *depun**ător**, faianţ**ă*** → *faianţ**ator**, fărîm**a*** → *fărîm**ător**, împing**e*** → *împing**ător**, transcri**e*** → *transcri**itor***, *c**a**nă* → *c**ăneală**, atr**a**g**e*** → *atr**ăgătoare**, b**a**t**e*** → *b**ătătoare***;

– removing of the last vowel in the root, for example, *răşchi**a*** → *răşchi**tor**, acr**i*** → *acr**eală**, adun**a*** → *adun**ătoare***.

3.4 Problem of vowel and/or consonant alternation

The problem of derivation consists not only in the detection of the derivational rules for separate affixes, but also in the examination of the concrete consonant and/or vowel alternations for the affixes. It is important that not all affixes need vowel and/or consonant alternations in the process of derivation. The vowel and/or consonant alternations are a subject for research not only in derivational morphology but also in inflectional morphology. Though there are some similarities, for example, *ean* → *en* (moldov*ean* – moldov*en*i – moldov*en*esc), *o* → *u* (s*o*ră – s*u*rori – s*u*rioară), *oa* → *o* (pl*oa*ie – pl*oi* – pl*oi*ţă), *t* → *ţ* (bărba*t* – bărba*ţ*i – bărbă*ţ*ie), etc. But the derivational alternations differ from those inflectional, for example: *at* →ăţ (argat – argăţesc), *ar* → *er*, (adevăr – adeveri), *g* → *s* (împunge – împunsătură), etc.

There are no cases with consonant and/or vowel alternations in the process of derivation with suffixes. It means that there are situations when the derivation is made up with minimum number of alternations and with maximum cases of changes in the root, for example: *a* →ă *a* →ă (balsam – îmbălsăma), *a* →ă *a* →ă *a* →ă (caimacam – căimăcămie) etc. Possible vowel and/or consonant alternations are so varied that it is difficult to describe them all in a chapter, but it is possible, at least, to classify them:

– removing of final vowel and changing of final consonant, for example, *descre**şte*** → *descre**scătoare**, închi**de*** → *închi**zătoare**, încre**de***

→ *încre**zătoare**, promi**te*** → *primi**ţătoare***;

– changing of the vowels in the root, for example, *c**a**t**a**r**a**mă* → *înc**ătărăm**a**, primăv**a**r**ă*** → *desprimăv**ăr**a**, rădăcin**ă*** → *dezrădăcin**a**, pl**a**toş**ă*** → *împl**ă**toş**i***;

– changing in the root, for example, *rîde* → *rîzătoare, recunoaşte* → *recunoscătoare, roade* → *rozătoare, sta* → *stătătoare, şedea* → *şezătoare, vedea* → *văzătoare, şti* → *ştiutor*.

On purpose of precision which affixes have alternations in the process of derivation, the digital variant of the derivatives dictionary has been studied. Some of them are illustrated in the Table 1.

Taking into consideration all these observations, it is easier to understand the derivative structure, namely the prefixes, stem and the suffixes of the derivatives. It represents a starting point for the process of automatic derivatives generation.

4 Derivatives generation

Besides the problem of derivatives analysis there is a wish to have the possibility to generate new derivatives, taking into account the stem and affix peculiarities. In the process of linguistic resources completion by automatic derivation appear a natural tendency to use the most frequent affixes. In reality, the most productive affixes prove to be problematic because of their irregular behaviour. That is why for the research there have been chosen those affixes that have allowed to establish simpler behaviour rules, as not to appeal to too much exceptions [9]. That is why the examples of prefixation with *re-*, *ne-*, *in-*/*im-*, and suffixation with *-re*, *-bil*, *-tor*, *-toare*, *-esc*/*-ească*, *-iza* are described below.

4.1 Automatic prefixation

The rule of derivation with the prefix *re-* is the following, let ω be the infinitive of the verb, then the word of the form $\omega\prime = re\omega$ is also the infinitive of the verb, namely

Table 1. Vowel and/or consonant alternation

Alter. vow/cons	Root/ Stem	Context of alt. vow/cons	Pref/ Suf	Word Examples
a →ă	albastru	bas – băs	el	albăstrel
	arab	rab – răb	ească	arăbească
	cărare	rar– răr	uşă	cărăruşă
	dalb	dal – dăl	ior	dălbior
a → ăr	gustare	tar – tăr	ică	gustărică
a → e	iarbă	iar – ier	ăluă	ierbăluă
a → ă	dandana	dan – dăn		dăndănaie
a → ă		dan – dăn	ie	
	balsam	bal – băl	îm	
		sam – săm	a	îmbălsăma
a → ă	calafat	cal – căl	ui	
a → ă		laf – lăf		
a → ă		fat – făt		călăfătui
a → ă	Banat	ban – băn	ean	
at → ăţ		nat – năţ		bănăţean
a → ă	baniţă	ban – băn	ioară	
ţ –c		niţ – nic		bănicioară
a → e	iatac	iat - iet	el	
a → ă	iatac	tac - tăc		ietăcel
at → ieţ	băiat	ăiat – ăieţ	andru	băieţandru

$$[\omega]_{inf} \rightarrow \left[re\,[\omega]_{inf}\right]_{inf}.$$

In this case there are derivatives like (a) filma → (a) *re*filma, (a) genera → (a) *re*genera, etc. As in the previous case, there are no any vowel and/or consonant alternations in this case.

In this context it is observed that the root for the derivative with the prefix *re-* and suffix *-re* is the infinitive of the verb. So, let ω be the infinitive of a verb, then $\omega\prime = re\omega re$ is a noun, namely

$$[\omega]_{inf} \rightarrow \left[re\,[\omega]_{inf}\,re\right]_{Nn}.$$

The derivatives would be: (a) întîlni → *re*întîlni*re*, (a) verifica → *re*verifica*re*. There are no vowel and/or consonant alternations.

Another known affix, which will permit to generate many derivatives, is the prefix *ne-*. Thus, let ω be an adjective of the form $\omega\prime = \omega\beta$, where $\beta \in \{$ *-tor, -bil, -os, -at, -it, -ut, -ind, -ind*$\}$, then the derivatives of the form $\omega'' = ne\omega\beta$ are possible to generate and the resulted derivatives will be also adjectives.

$$[\omega\beta]_{Adj} \rightarrow \left[ne\,[\omega\beta]_{Adj}\right]_{Adj}.$$

In this case the obtained derivatives would be: conductor → *ne*conductor, nobil → *ne*nobil, invidios → *ne*invidios, iubit → *ne*iubit, născut → *ne*născut. In the process of derivation with the prefix *ne-* the vowel and/or consonant alternations are not observed. Though a question appears, what the endings β represent. If in some cases it is clear that they are forms of participle or gerund, then the strings *tor* and *bil* are lexical suffixes. So an interest appears to the process of derivation with these suffixes.

The derivatives with the prefixes $im-/in-$, as a rule, are adjectives, rarely nouns and verbs. The most numerous derivatives with prefix $in-/im-$ are adjectives formed with the suffix *bil*, for example, *in*cura*bil*, *in*estima*bil*, etc. So, being the adjectives of the form $\omega\prime = \omega bil$, they form derivatives of the form $\omega'' = \omega bil$, where $\omega \in \{in-, im-\}$ [1].

Another well contoured group is that of adjectives derivated with the suffixes $-ent$ and $-ant$: inader*ent*, incoer*ent*, independ*ent*, etc. Similar, being the adjectives $\omega\prime = \omega\gamma$, they form derivatives $\omega'' = \beta\omega\gamma$, where $\beta \in \{in-, im-\}$ and $\gamma \in \{-ent, -ant\}$. In both cases the choice of the β depends on the first letter of the adjective ω, and namely in the case when the letter is b or p then $\beta = im-$, in other cases it is $in-$.

4.2 Automatic suffixation

For the suffix *-re* there is the following rule: let ω be the infinitive of a verb, then the word of the form $\omega\prime = \omega re$ is a noun, namely

$$[\omega]_{inf} \rightarrow \Big[[\omega]_{inf}\, re\Big]_{Nn}.$$

This formal model can generate derivatives such as: citi → citi*re*, mînca → mînca*re*, etc. In the process of derivation there are no vowel and/or consonant alternations.

That is because the suffixes *-tor* and *-bil* have been studied. Both of them have the same origin. Thus, let ω be the infinitive of the verb of the form $\omega\prime = \omega\beta$, where $\beta \in \{-a, i\}$, then it is possible to form the derivatives of the form $\omega'' = \omega\beta\gamma$, where $\gamma \in \{-tor, -bil\}$ is adjective/noun.

This examination includes the verbal lexical simple suffix *-iza*, which has neologic origin and nowadays is very productive and has very strong relation with the lexical suffixes *-ism* and *-ist*. Thus, let ω be an adjective/noun of the form $\omega\prime = \omega\beta\gamma$, where $\gamma \in \{-ism, -ist\}$, then it is possible to say about the derivatives the following:

1. if $\beta \in \{-an, -ian\}$, then the word of the form $\omega\prime\beta = \omega iza$ is a verb;

2. of $\beta \in \{-ean\}$, then the word of the form $\omega\prime e\boxed{\text{a}}n = \omega iza$ is a verb, where $\boxed{\text{a}}$ represents the cut out of the vowel a;

3. if $\beta = \mu ic$, where:

- $\mu \in \{-at, -et, -ot, -if\}$, then the word of the form $\omega\prime = \omega\mu iza$ is a verb;
- $\mu \notin \{-at, -et, -ot, -if\}$, then the word of the form $\omega\prime = \omega\beta iza$ is a verb;

4. if $\beta \in -ur$ă , then the word of the form $\omega\prime = \omega ur\boxed{\breve{a}}iza$ is a verb, where $\boxed{\breve{a}}$ represents the cut out of the vowel *a*.

Thus, the examples of such derivatives are: alcan → alcan*iza*, european → european*iza*, dramatic → dramat*iza*, cosmetic → cosmet*iza*, patriotic → patriot*iza*, ştiinţific → ştiinţif*iza*, caricatură → caricatur*iza*, friptură → fripturiza [1].

The word gender changing can be achieved by switching to other corresponding suffixes, for example, $-tor \rightarrow -toare$, $-esc \rightarrow$ *-ească*, etc. Thus, it was observed that the gender changing is made with the help of suffixation, not of prefixation one. The lexicon, mentioned above, consists of suffixed derivatives only with *-tor*, only with *-toare* and with *-tor* and *-toare* at the same time. There are 148 words (nouns and/or adjectives) of the form $\omega' = \omega tor$, which could change into the words of the form $\omega'' = \omega toare$. Similarly, there are 42 words (nouns and/or adjectives) of the form $\omega' = \omega toare$ which could change into the words of the form $\omega'' = \omega tor$. Nevertheless, these 190 words generated in an automatic way should be validated. First of all, words were checked on their presence in RRTLN. 122 from all generated words were present there. The remaining words were checked in the electronic documents of the Internet, and 49 of 68 derivatives have been validated. Thus 95% of generated words were valid.

The same situation is with the pair of the suffixes *-esc* and *-ească*. According to the same lexicon, it consists of 274 of derivatives with suffixes *-esc*, and 249 with the suffix *-ească*. Note, that 229 of the derivatives are suffixes both with *-esc* and *-ească*. It is natural to assume that the words (nouns and/or adjectives) of the form $\omega' = \omega esc$, could change into the words of the form $\omega'' = \omega easc\breve{a}$. Similarly, the words (nouns and/or adjectives) of the form $\omega' = \omega easc\breve{a}$ could change into the words of the form $\omega'' = \omega esc$. Generating in an automatic way

those derivatives which lack in the case of gender and checking them in an automatic way in the electronic documents, it was established that with the help of RRTLN there were validated 43 words of all 65 generated words. Another 12 of 22 remaining derivatives were validated using a web application based on Google search engine opportunities. So, 84% of obtained words were validated.

5 Problem of derivatives validation

Automatic derivation represents an overgenerating mechanism. That is why validation of generated words is needed.

5.1 Models of validation

One of the methods of new word validation consists in manual verification of every new generated derivative as to correspond to semantic and morphologic rules. In the case of the proceeding is performed by a specialist in domain, the specific disadvantages of a manual work appear: considerable resources of time and the possibility to make mistakes. So, this method of validation becomes inefficient [1].

Another method of validation consists of the verification of the derivatives in the existent electronic documents.

5.2 Automatic validation

There are different types of electronic documents.

The first idea that appears – to validate words using existent corpora, that represent verified documents – seems to be the best solution. The condition for being the panacea in the new word validation is a representative corpus, with a big number of words from different domains. As there are no representative Romanian corpora, it is not possible to consider it a good idea.

On the other hand there are documents on Internet, that are not verified, that is why they are not credible. In order to make it more precise, the searching on the Internet should be made for the documents

typed only in Romanian language. Besides this, it is necessary that the following be assured: the possibility to exclude word segmentation; the part of speech of the derivatives [9].

This validation tool divides the generated derivatives in three categories. The first one contains words that are not found in Internet. The second consists of the derivatives that appear less than a frequency limit of n, in our case $n = 1000$. Derivatives that are more frequent that limit n, are registered in the third group. This classification pretends that the words, that are listed more than frequency limit of n, are surely valid. Those, that are from the second group, can be valid but should be verified by specialists in linguistics. The derivatives, that are not present, could not be valid.

The idea of classification pretends to be a mixt method of validation, because needs only the manual verification for the words from the second category.

6 Conclusions

Generation of derivatives is not a trivial problem, because the process does not have a regular mechanism. The solution to store all derivatives of a dictionary is a reasonable one, because these derivatives still will not cover the full diversity of language, being in continuous evolution. From the other hand, the approach to generate constraint derivatives according to constraint rules for derived groups is a mechanism of overgeneration, when the validation phase excludes many wrong formed words. Well defined rules will increase the level of the correct words generation.

References

[1] S. Cojocaru, E. Boian, M. Petic. *Stages in automatic derivational morphology processing*, KEPT2009, Knowledge Engineering, Principles and Techniques, Selected Papers, Cluj-Napoca, July 2 - 4, 2009, pp.97-104.

[2] N. Percova. *RUSLO: An Automatic System for Derivation in Russian - http : //lcl.srcc.msu.ru/library/pertsova_ruslo.pdf* - 22.04.09

[3] V. Duško, C. Krstev. *Derivational Morphology in a E-Dictionary of Serbian*, In Zygmunt Vetulani (ed.), Proceedings of the 2nd Language & Technology Conference, Poznan, Poland, 2005, pp. 139–143.

[4] F. Namer, G. Dall. *GeDeriF: Automatic Generation and Analysis of Morphologically Constructed Lexical Resources*, In LREC: 2nd International Conference on Language Resources & Evaluation,

[5] D. Tufiş, L. Diaconu, A. M. Barbu, C. Diaconu. *Romanian language morphology, a reversible and reusable linguistic resource*, Language and Technology, Publishing House of Romanian Academy, Bucureşti, 1996, pp. 59–65. (in Romanian)

[6] F. Carota. *Derivational Morphology of Italian: Principles of Formalization*, Literary and Linguistic Computing, Vol. 21, Suppl. Issue, 2006.

[7] S. Constantinescu. *The dictionary of derivated words*. Editura Herra, Bucureşti, 2008. (in Romanian)

[8] M. Petic. *Automatic derivational morphology contribution to Romanian lexical acquisition*. Special issue: Natural Language Processing and its Application. Research in Computing Science, Mexico, vol. 46, 2010, pp. 67–78.

[9] M. Petic. *Automatic extention of Romanian linguistic resources*, Romanian Workshop for Linguistic Tools and Resources Volume, Publishing House of the University "Al. I. Cuza", Iaşi, România, 2008, pp. 151–160. (in Romanian)

M. Petic, Received June 25, 2010

M. Petic
Institute of Mathematics and Computer Science,
Academy of Sciences of Moldova
Academiei 5, Chişinău MD-2028 Moldova
E–mail: *mirsha@math.md*