

The applications of combinatorics on words

Dominique Perrin

October 8, 2006

Outline: examples on the interplay theory/practice met in the framework of the Lothaire project and involving:

- The Unitex package
- The Burrows Wheeler transformation
- Transducers

A collaborative project run with Jean Berstel and comprising

- Combinatorics on Words, Addison Wesley, 1984, Cambridge 1997
- Algebraic Combinatorics on Words, Cambridge, 2002
- Applied Combinatorics on Words, Cambridge, 2005



Lothaire's Books



This is the third book in the Lothaire's series, following the volumes "Combinatorics on Words" and "Algebraic Combinatorics on Words" already published. Available at [Cambridge University Press](#) since July, 2005

Its objective is to present in a unified manner the various applications of combinatorics on words. The application areas include core algorithms for text processing, natural language processing, speech processing, bioinformatics, and several areas of applied mathematics such as combinatorial enumeration and fractal analysis.

The intended audience is the general scientific community. No special knowledge is needed, and familiarity with the application areas or with the material covered in the previous volumes is not required. In particular, the content, including problems and algorithms, is accessible to anyone working in the area of computer science.

As with the previous volumes, this book is written in collaboration by a group of authors, under the guidance of the editors.

[Content](#)
[Implementation of algorithms](#)
[Solutions to problems](#)


This book covers developments on new topics in the domain of Combinatorics on Words. Available at [Cambridge University Press](#). Available since May, 2002. ISBN: 0521812208

[Content](#)
[Solutions to problems](#)
[Errata](#)


The 1983 edition of M. Lothaire's "Combinatorics on Words", Encyclopedia of Mathematics, Vol. 17, Addison-Wesley has been reprinted in 1997 at [Cambridge University Press](#), with only minor corrections, in the *Cambridge Mathematical Library*.

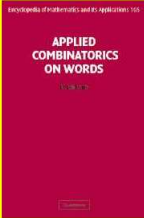
lothaire - Mozilla

Fichier Edition Affichage Aller à Marque-pages Outils Fenêtre Aide

Précédent Suivant Actualiser Arrêter <http://www-igm.univ-mlv.fr/%7Eberstel/Lothaire/AppCWImplementations.ht> Rechercher Imprimer

Accueil Marque-pages

Applied Combinatorics on Words: Implementation of algorithms



Algorithms on words

A set of [computer programs in Java](#) for the algorithms of Chapter 1 is available in a preliminary form. They can be freely copied and used with the mention of their origin. The idea is to present an illustration of a possible effective implementation rather than fine tuned optimal software. No guarantee at all is given for correctness. A [documentation](#) is in progress.

Structures for indexes

[Computer programs in Java and C](#) for the algorithms of Chapter 2 and for other text processing algorithms are available.

Statistical natural language processing

Programs for the algorithms of this chapter are available at:
<http://www.research.att.com/sw/tools/fsm>
<http://www.research.att.com/sw/tools/grm>
<http://www.research.att.com/sw/tools/dcd>

Statistics on words with applications to biological sequences

Computations of words with exceptional frequency in DNA were performed with programs available at: <http://www-mig.jouy.inra.fr/ssb/rmes/>

Periodic structures in words

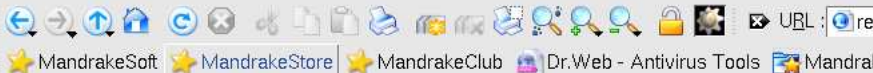
Concerning this chapter, principal algorithms have been implemented in the mreps software <http://www.loria.fr/mreps/>.

1 2 3 4 perrin KPPP Ration gv: Ap Applic lothaire file/h emac Gimp 8 22

The anonymous french package

A set of (unguaranteed) Java programs comprising:

- Basic algorithms on words
- Automata and regular expressions
- Transducers
- Parsing
- Enumeration



★ MandrakeSoft

★ MandrakeStore

★ MandrakeClub

Dr.Web - Antivirus Tools

★ Mandra

All Classes

[Alphabet](#)
[BMinimizer](#)
[Buffer](#)
[CompileExpression](#)
[DFA](#)
[DFT](#)
[DListInt](#)
[EcoRMinimizer](#)
[Element](#)
[ElementaryAlgorithms](#)
[Entropy](#)
[ExpressionCompiler](#)
[FMinimizer](#)
[FixedArrayTrie](#)
[ForaxTrie](#)
[Grammar](#)
[HalfEdge](#)
[HopcroftMinimizer](#)
[ICFA](#)
[IDFA](#)
[INFA](#)
[InfoDFA](#)
[InfoNFA](#)
[IntList](#)
[IntQueue](#)
[LL](#)
[LinkedNFA](#)
[MinAutomaton](#)
[Minimizer](#)

Package **Class** [Tree](#) [Deprecated](#) [Index](#) [Help](#)[PREV CLASS](#) [NEXT CLASS](#)SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METHOD](#)

Interface Minimizer

All Known Implementing Classes:

[BMinimizer](#), [FMinimizer](#), [HopcroftMinimizer](#), [NbisMinimizer](#), [NMinimizer](#), [RMinimizer](#)

```
public interface Minimizer
```

This interface is implemented by the classes

- [BMinimizer](#) implementing the Brzozowski minimization algorithm.
- [HopcroftMinimizer](#) implementing the Hopcroft minimization algorithm.
- [FMinimizer](#) implementing the fusion minimization algorithm.
- [NMinimizer](#) implementing the naive minimization algorithm.
- [RMinimizer](#) implementing the Revuz minimization algorithm.

The Burrows Wheeler transform

Let $w = abracadabra$. The list of conjugates of w sorted in alphabetical order is represented below.

	1	2	3	4	5	6	7	8	9	10	11
1	<i>a</i>	<i>a</i>	<i>b</i>	<i>r</i>	<i>a</i>	<i>c</i>	<i>a</i>	<i>d</i>	<i>a</i>	<i>b</i>	<i>r</i>
2	<i>a</i>	<i>b</i>	<i>r</i>	<i>a</i>	<i>a</i>	<i>b</i>	<i>r</i>	<i>a</i>	<i>c</i>	<i>a</i>	<i>d</i>
3	<i>a</i>	<i>b</i>	<i>r</i>	<i>a</i>	<i>c</i>	<i>a</i>	<i>d</i>	<i>a</i>	<i>b</i>	<i>r</i>	<i>a</i>
4	<i>a</i>	<i>c</i>	<i>a</i>	<i>d</i>	<i>a</i>	<i>b</i>	<i>r</i>	<i>a</i>	<i>a</i>	<i>b</i>	<i>r</i>
5	<i>a</i>	<i>d</i>	<i>a</i>	<i>b</i>	<i>r</i>	<i>a</i>	<i>a</i>	<i>b</i>	<i>r</i>	<i>a</i>	<i>c</i>
6	<i>b</i>	<i>r</i>	<i>a</i>	<i>a</i>	<i>b</i>	<i>r</i>	<i>a</i>	<i>c</i>	<i>a</i>	<i>d</i>	<i>a</i>
7	<i>b</i>	<i>r</i>	<i>a</i>	<i>c</i>	<i>a</i>	<i>d</i>	<i>a</i>	<i>b</i>	<i>r</i>	<i>a</i>	<i>a</i>
8	<i>c</i>	<i>a</i>	<i>d</i>	<i>a</i>	<i>b</i>	<i>r</i>	<i>a</i>	<i>a</i>	<i>b</i>	<i>r</i>	<i>a</i>
9	<i>d</i>	<i>a</i>	<i>b</i>	<i>r</i>	<i>a</i>	<i>a</i>	<i>b</i>	<i>r</i>	<i>a</i>	<i>c</i>	<i>a</i>
10	<i>r</i>	<i>a</i>	<i>a</i>	<i>b</i>	<i>r</i>	<i>a</i>	<i>c</i>	<i>a</i>	<i>d</i>	<i>a</i>	<i>b</i>
11	<i>r</i>	<i>a</i>	<i>c</i>	<i>a</i>	<i>d</i>	<i>a</i>	<i>b</i>	<i>r</i>	<i>a</i>	<i>a</i>	<i>b</i>

The word $T(w)$ is the last column of the array. Thus $T(w) = rdarcaaabb$.

The BW compression method

Principle: two passes

- 1 First transform the source file by BW.
- 2 Use a standard compression method (run length or move-to-front) to compress the output.

Complexity: linear (using a suffix tree of the input).

The Parikh vector of the word $w = abracadabra$ is $v = (5, 2, 1, 1, 2)$ and $\rho(v) = \{5, 7, 8, 9\}$.

$$\sigma = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 \\ 1 & 3 & 7 & 11 & 4 & 8 & 5 & 9 & 2 & 6 & 10 \end{pmatrix}$$

$$\pi = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & \mathbf{7} & \mathbf{8} & \mathbf{9} & 10 & 11 \\ 3 & 6 & 7 & 8 & 9 & 10 & 11 & 5 & 2 & 1 & 4 \end{pmatrix}$$

Theorem (Gessel, Reutenauer)

For any positive vector $v = (n_1, n_2, \dots, n_k)$ with $n = n_1 + \dots + n_k$, the map $w \mapsto \pi = P(w)$ is one to one from the set of conjugacy classes of primitive words of length n on an alphabet A with k symbols with Parikh vector v onto the set of cyclic permutations on $\{1, 2, \dots, n\}$ such that $\rho(v)$ contains $\text{des}(\pi)$.



Unitex is a corpus processing system, based on automata-oriented technology. The main functions are:

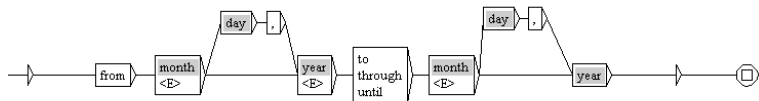
- building, checking and applying electronic dictionaries
- pattern matching
- applying lexicon grammar tables

51 matches

 Enable links Allow concordance edition

armed forces and head of the government [from 1973 to 1990](#), he gave orders to eliminate, torture genocidal group that governed Cambodia [from 1975 to 1979](#) and caused the deaths of as many as 2 presided over Cambodia's Killing Fields [from 1975 to 1979](#). Tens of thousands of their real or 2 Menachem Begin, Israel's Prime Minister [from 1977 to 1983](#). His vision of an Israel which must i states that passed right-to-carry laws [from 1977 to 1992](#). He contends that after more relaxed lable when I did the study on the years [from 1977 to 1994](#). It is likewise false that I did "not ulfed the Paris City Hall that he ruled [from 1977 until 1995](#). A rebellious faction of Gaullist our leader and the country's President [from 1979 to 1990](#), was sexually molesting his adolescen post he filled with skill and judgment [from 1979 to 1990](#). In 1989, in perhaps his most dramati oad because he too wanted to be a hobo. [From 1981 to 1987](#), IBM used the Tramp as the logo to ad e father of judge shows, Joseph Wagner. [From 1981 to 1993](#), his sessions of The People's Court e systematic checks of white Unos dating [from 1983 to 1989](#). One by one, owners were called into s. Says Norden, a chaplain at Pentridge [from 1985 to 1992](#): "Our prisons are places of violence, ear reign of General Ibrahim Babangida, [from 1985 to 1993](#), Abiola himself often operated as a b hich could lead to a rematch.) As mayor [from 1985 to 1993](#), Suarez was known for his thin skin, ouths murdered by firearms went up 153% [from 1985 to 1995](#). In some ways the school-yard killing Californians who died in nursing homes [from 1986 through 1993](#). In more than 7% of the cases, 1 ppens, Lauder was ambassador to Austria [from 1986 to 1987](#) and is a notable Schiele collector.) ic. He was also governor of Connecticut [from 1990 to 1995](#). Currently, he's a professor at the U Number of blacks who moved to the South [from 1990 to 1996](#) 287,400: Number of blacks who left th Number of blacks who left the Northeast [from 1990 to 1996](#) 368,800: Number of blacks who moved t deposited in his personal bank accounts [from 1991 to 1995](#) is what the magistrates are now seeki deposited in his personal bank accounts [from 1991 to 1995](#), much of it in cash, is what the magi end has accelerated in the past decade. [From 1991 to 1996](#), regular ed accounted for just 23% of enefit of travel--namely, foreign cash. [From 1991 to 1996](#), the number of visitors to Laos jumpe ho gyrated as one of her backup dancers [from 1991 to 1996](#); in Tokyo. Christopher Conte, 29, is t about the Clintons and their friends. [From 1993 to 1997](#), two Scaife foundations transferred \$ aine and methamphetamine at town dances [from 1993 to 1997](#). And just in case that wasn't enough nvestors had nearly doubled their money [from 1994 to 1997](#), could have got so deeply in trouble. hose that troopers stopped and searched [from 1995 to 1997](#) were white; 17% of drivers are black, e and natural gas, were market darlings [from 1995 to 1997](#), but now have more detractors than an efold in the past 10 years; it grew 14% [from 1996 to 1997](#) alone. Total Canadian revenues from t of adult trade books slipped nearly 7% [from 1996 to 1997](#), and overall sales dropped 3.4%. Unde as a young Jewish girl and stayed there [from October 1942 to May 1945](#), living in several refuge

The graph of durations



The DELAF dictionary

The files used are all parts of the delaf dictionary. It is a dictionary of about 800000 words representing all words in french at all grammatical forms. A typical part of the delaf looks like

activassent

activasses

activassiez

activassions

activeur

activeur

activeur biologique

activeur tissulaire du plasminogène

Compound words (like 'activeur biologique') are also entries of the dictionary. Some words are repeated because the file is a stripped version of one associating with each word some information giving its grammatical category.

	nbWords	alphabet	Trie	IDFA	Minimal
delafAb.txt	39	29	97	73	64
delafA.txt	731	53	2526	2085	1202
delafB.txt	1556	58	5182	4254	2243
delafC.txt	2550	63	9310	7742	4086
delafa.txt	60324	72	175546	136886	36499
delaf.txt	802009	90	2203261	1683979	273716

Figure: Parts of the delaf french dictionnaire.

	Fusion	Naive	NaiveBis	Brzozo	Hopcroft	Revuz
delafAb.txt	33	30	29	126	58	30
delafA.txt	7188	398	281	202	297	116
delafB.txt	31733	2012	1213	341	693	159
delafC.txt		7427	4135	507	1199	210
delafa.txt		428487	6759	6922		2240
delaf.txt						78550

Figure: Computation time in ms. on a Dell portable

	Naive	NaiveBis	Brzozo	Hopcroft	Revuz
delafA.txt	288	211			
delafa.txt		325681	344149		2179
delaf.txt					19135

Figure: Computation time in ms. on Monge

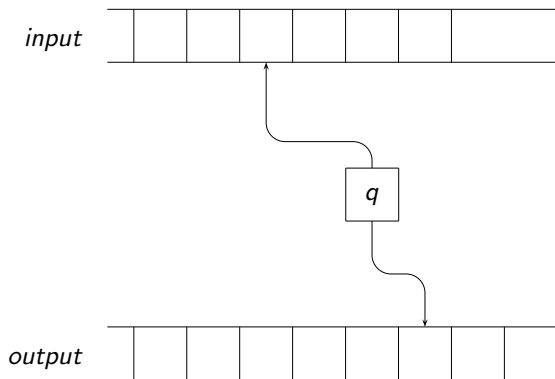


Figure: A transducer reads the input and writes the output.

Main features of transducers

- Composition
- Determinization
- Minimization

If properly presented, these algorithms are essentially the same as those for finite automata.

Rational relations are closed under composition (basic property).
Surprise: the simplest proof is a direct construction of a transducer for the composition (instead of using morphisms, inverse morphisms and intersections with rational sets as usually done).

COMPOSETRANSDUCERS(\mathcal{G}, \mathcal{T})

```
1  ▷  $\mathcal{G}$  and  $\mathcal{T}$  are literal transducers
2   $\mathcal{U} \leftarrow \text{NEWTRANSDUCER}()$ 
3  for each edge  $(p, a, b, q)$  of  $\mathcal{G}$  do
4      for each edge  $(r, b, c, s)$  of  $\mathcal{T}$  do
5          add  $((p, r), a, c, (q, s))$  to the edges of  $\mathcal{U}$ 
6  for each edge  $(p, a, \varepsilon, q)$  of  $\mathcal{G}$  do
7      for each state  $r$  of  $\mathcal{T}$  do
8          add  $((p, r), a, \varepsilon, (q, r))$  to the edges of  $\mathcal{U}$ 
9  for each edge  $(r, \varepsilon, c, s)$  of  $\mathcal{T}$  do
10     for each state  $p$  of  $\mathcal{G}$  do
11         add  $((p, r), \varepsilon, c, (p, s))$  to the edges of  $\mathcal{U}$ 
12   $\text{INITIAL}_{\mathcal{U}} \leftarrow \text{INITIAL}_{\mathcal{G}} \times \text{INITIAL}_{\mathcal{T}}$ 
13   $\text{TERMINAL}_{\mathcal{U}} \leftarrow \text{TERMINAL}_{\mathcal{G}} \times \text{TERMINAL}_{\mathcal{T}}$ 
14  return  $\mathcal{U}$ 
```

Example

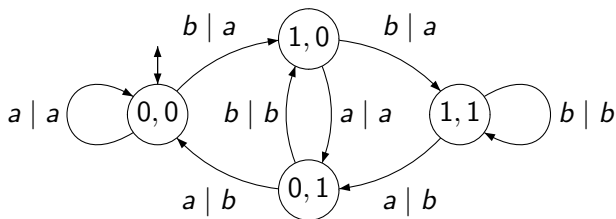


Figure: The right 2-shift.

Sequential Transducers

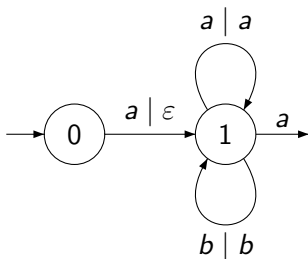


Figure: A sequential transducer for the circular left shift on words beginning with a .

Determinization of Transducers

In a first step, we build the set of states and the next state function of the resulting sequential transducer \mathfrak{B} .

EXPLORE($\mathcal{T}, S, \mathfrak{B}$)

- 1 $\triangleright \mathcal{T}$ is a collection of sets of half-edges
- 2 $\triangleright S$ is an element of \mathcal{T}
- 3 **for** each letter a **do**
- 4 $(v, U) \leftarrow \text{LCP}(\text{NEXT}(S, a))$
- 5 $\text{NEXT}_{\mathfrak{B}}(S, a) \leftarrow (v, U)$
- 6 **if** $U \neq \emptyset$ and $U \notin \mathcal{T}$ **then**
- 7 $\mathcal{T} \leftarrow \mathcal{T} \cup U$
- 8 $(\mathcal{T}, \mathfrak{B}) \leftarrow \text{EXPLORE}(\mathcal{T}, U, \mathfrak{B})$
- 9 **return** $(\mathcal{T}, \mathfrak{B})$

We can finally write the function realizing the determinization of a transducer into a sequential one.

TOSEQUENTIALTRANSDUCER(\mathfrak{A})

```
1  ▷  $\mathfrak{A}$  is a transducer
2   $\mathfrak{B} \leftarrow \text{NEWSEQUENTIALTRANSDUCER}()$ 
3   $l \leftarrow \text{CLOSURE}(\{\varepsilon\} \times \text{INITIAL}_{\mathfrak{A}})$ 
4   $\text{INITIAL}_{\mathfrak{B}} \leftarrow l$ 
5  ▷  $\mathcal{T}$  is a collection of sets of half-edges
6   $\mathcal{T} \leftarrow l$ 
7   $(\mathcal{T}, \mathfrak{B}) \leftarrow \text{EXPLORE}(\mathcal{T}, l, \mathfrak{B})$ 
8  for  $S \in \mathcal{T}$  do
9      for  $(u, q) \in S$  do
10         if  $q \in \text{TERMINAL}_{\mathfrak{A}}$  then
11              $\text{TERMINAL}_{\mathfrak{B}}(S) \leftarrow u$ 
12  return  $\mathfrak{B}$ 
```

Example

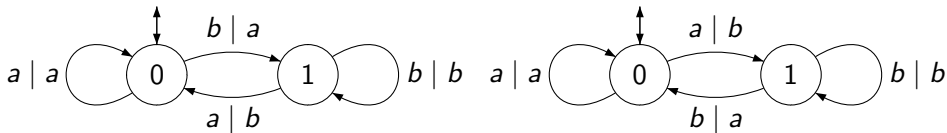


Figure: The circular right shift on words ending with a and its inverse.

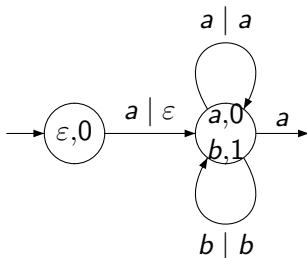


Figure: A sequential transducer for the circular left shift on words

Minimization of transducers

Performed in two steps:

- normalization
- (ordinary) minimization

LONGESTCOMMONPREFIXARRAY(\mathfrak{A})

- 1 $\triangleright P, P'$ are arrays of strings initially null
- 2 $\triangleright M$ is the matrix of transitions of \mathfrak{A} and N the vector of terminals
- 3 **do** $P \leftarrow P'$
- 4 $P' \leftarrow MP + N$
- 5 **while** $P \neq P'$
- 6 **return** P

NORMALIZETRANSDUCER(\mathcal{A})

```
1  $P \leftarrow \text{LONGESTCOMMONPREFIXARRAY}(\mathcal{A})$ 
2  $(\lambda, i) \leftarrow \text{INITIAL}$ 
3  $\text{INITIAL} \leftarrow (\lambda P[i], i)$ 
4 for  $(p, a) \in Q \times A$  do
5      $(u, q) \leftarrow \text{NEXT}(p, a)$ 
6      $\text{NEXT}(p, a) \leftarrow P[p]^{-1} u P[q]$ 
7 for  $p \in Q$  do
8      $T[p] \leftarrow P[p]^{-1} T[p]$ 
```

Example

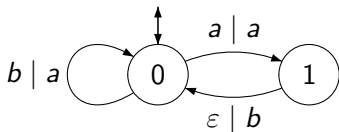
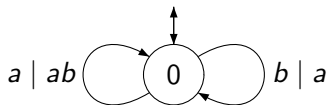


Figure: The Fibonacci morphism.

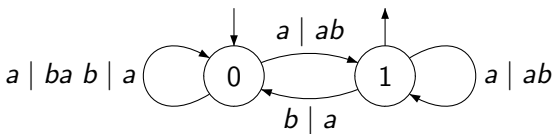
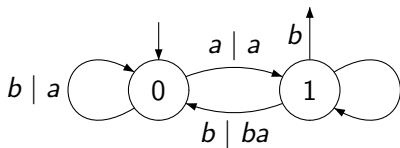


Figure: The normalization algorithm.

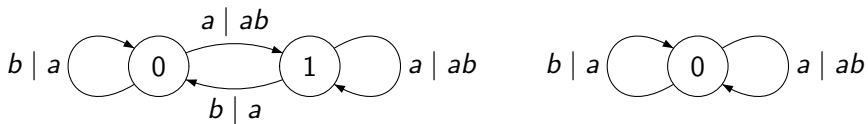


Figure: The minimization algorithm.

