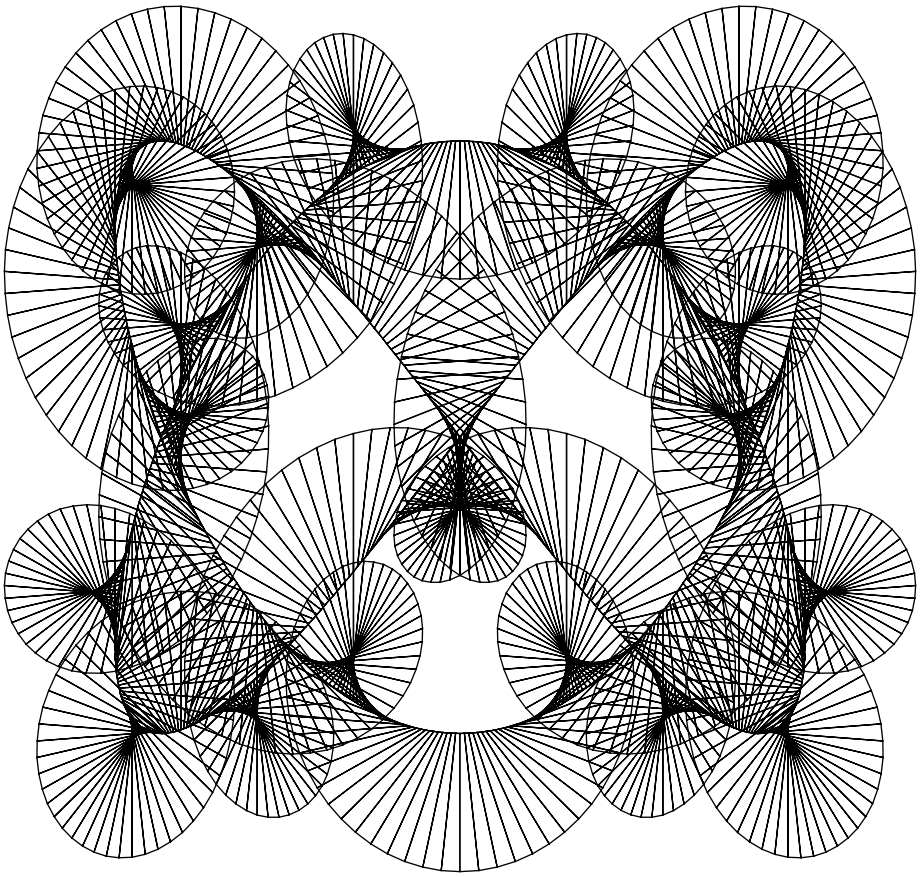
STR

Archiving and viewing mathematical documents

(version 3)

Jean-Marc Drézet



Sommaire

1 Brief Description **3**

2 The STR program in detail **3**

3 The visualisation programs **8**

4 Installation **8**

1 Brief Description

STR is a program for the operating system Linux, designed to produce Web pages describing a set of mathematical documents by giving their list, each of them having a number which can be used to easily display them. Two types of documents are distinguished: preprints from ArXiv, and other documents. For the documents of ArXiv, the description contains the names of the authors, the title of the article and the associated summary given by the authors. The description of the other documents has only the names of the authors and title.

The main page is represented on figure 1.

Each letter of the part **"Textes ArXiv"** is a link to the alphabetical list of articles whose first author has a name beginning with this letter, followed by the list of articles where another author has a name beginiing with this same letter (see figure 2).

The page corresponding to **Autres textes** shows the alphabetical list of the other articles (see figure 3).

The page corresponding to **Auteurs** gives the list of the authors. Each author name is a link to a page giving the list of articles of this author, ArXiv articles and other articles (see figures 4 and 5).

The graphical program **xxw** can be used to view an article, knowing its number. The command line program **See** can do the same.

2 The STR program in detail

The **STR** program is used to generate web pages and numbers of items based on the informations given by the user. In the case of a document from ArXiv, the abstract of the article (i.e. the saved summary web page) should be placed in a particular directory, as well as the (compacted) Postscript file containing the article. For other items, information must be written in a special file (name of the file containing the article, names of the authors, title and type of file).

2.1 Preparation of the directories – The files containing the documents must be in a unique directory, with two different subdirectories named divers and resume. The main directory will contain the compacted Postscript files containing the ArXiv articles. The directory divers will contain the other documents. The directory resume will contain the summaries of ArXiv articles.

Another directory, containing some files produced and used by **STR** must also be created (it may be a subdirectory of the previous main directory). It must contain a subdirectory namedAuteurs. It must also contain the main page liste4.html of figure 1 (which can be easily

modified to use another language).
Finally, another directory must be created to contain the temporary files used when viewing items.

2.2 The initialization file – At startup, **STR** and the visualization program read the file .listerc3 which should be in the main directory of the user. It contains 13 lines giving the following information:

Line 1: *name of the directory containing the files produced by STR and some files used by it* – Example: ~/Articles/DB

Line 2: *name of the directory containing the documents* – Example: ~/Articles

Line 3: – *name of the suffix of the compacted articles from ArXiv* Example: ps.bz2

Line 4: *name of the suffixe of the articles in Postscript from ArXiv, once they have been decompacted* – Example: ps

Line 5: *name of the suffix of the articles written in dvi and compacted* – Example: dvi.bz2

Line 6: *name of the suffix of the files of articles in written dvi, once they are unpacked* – Example: dvi

Line 7: *name of the temporary directory, followed by a prefix* – Example: ~/tmp/XXX . Here "~/tmp" is the name of the directory that must be created (cf. 2.1).

Line 8: *name of the decompression program* – Example: bunzip2

Line 9: *name of the program used to for view ArXiv articles in PostScript* – Example: okular

Line 10: *name of the program used to view the other articles in PostScript* – Example: okular

Line 11: *name of the program used to view the other articles in dvi* – Example: okular

Line 12: *name of the program used to view the other articles in djvu* – Example: djview

Line 13: *name of the program used to view the other articles in pdf* – Example: okular

In the previous examples, the file .listerc3 will contain no the lines
~/Articles/DB
~/Articles
ps.bz2
ps
dvi.bz2
dvi

~/tmp/XXX
bunzip2
okular
okular
okular
djview
okular

2.3 Formats of the documents – The Arxiv articles – They must be in Postscript and compacted, their name must be of the form ZZZZZZ.suffix, where ZZZZZZ is a name given by the user (for example NameAuthor-05), and suffix is given in line 3 of the initialisation file .listerc3 (cf. 2.2, hence in the preceding example, it should be NameAuthor-05.ps.bz2). The file containing the resume of the article and saved from Arxiv should have a name with the same prefix ZZZZZZ, followed with .html (hence in the preceding example it will be NameAuthor-05.html).

2.4 Formats of the documents – The other texts – The other texts (non ArXiv) can be in *compacted Postscript*, in *compacted dvi*, in *pdf* or in *djvu*. The file names can be anything, except the compacted Postscript and dvi files, which must conform to the indications contained in the initialization file.

2.5 Compression and decompression – The same program must be used to compact the files in Postscript or in dvi. It should produce files with suffixes corresponding to what is indicated in the file .listerc3 (cf. 2.2).

2.6 Liste of the other articles – A file containing the of list non-ArXiv articles and information about them must be created and updated each time that a new article is added. The name of this file is liste_div.lst, and it must be located in directory of files produced and used by STR, given in the first line of the initialization file .listerc3 (see 2.2). Each article is described by 4 lines of information, and each of these blocks of 4 rows must be separated from the next by a blank line. The 4 rows contain the following information:

Line 1: *name of the file containing the article*

Line 2: *names of the authors*. Voir plus loin.

Line 3: *title*

Line 4: *format of the article*. It is ps, dvi, pdf or djvu.

If the name of an author has multiple parts, they must be separated by the character # instead of spaces, otherwise these parts will be considered as different authors names.

Example:

```
dupond-durand.pdf
Dupond Durand
Surfaces de Riemann
pdf

leroy-lebihan.ps.bz2
Leroy Le#Bihan
Differential geometry
ps

SM.djvu
Smith
Linear Algebra
djvu
```

2.7 Alias – Some names of authors (those of articles from ArXiv automatically recovered from abstracts) can appear written in different ways, especially if they have accents, and be counted as names of distinct authors. To avoid this inconvenience, it is possible to indicate in a file the different names of the same author. This is the file alias.lst, located in the directory of files produced and used by **STR**, and mentioned in the initialization file .listerc3 (see 2.2). This file is made up of lists (one for each author) separated by a line containing the single character # . The first row of a list is the real name of author, others have other names that can appear. *Note:* the main (first) author name of each list must be used by some article.

Exemple :

Dupré

Dupre

Dupr

#

Möppe

Moppe

Moeppe

Mppe

2.8 Practical implementation – With the parameters of the initialization file .listerc3 given in 2.2, one must proceed as follows to add a new article:

To add a new article from ArXiv, download the article in PostScript (and compact it) as well as his summary, their name must have the same prefix, for example Article23.ps.bz2 will be the name of the compacted Postscript file and Article23.html that of its summary. The article will be placed in the directory ~/Articles and the summary in ~/Articles/resume.

To add another article from another source, it will have to be in one of the supported formats (compacted Postscript, compacted dvi, pdf or djVu). It will be located in the directory ~/Articles/divers. The file ~/Articles/DB/liste_div.lst must be then completed with four lines describing the article, as shown in 2.6.

To update the liste of articles, one must then run the program **STR**, and give it the command all, followed by exit:

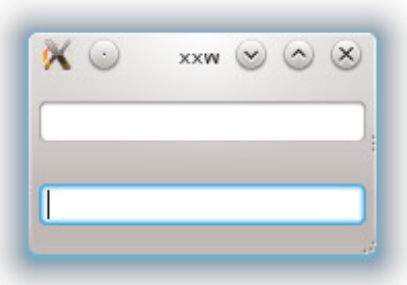
```
--> STR

interpcom-3.2     (2010)
-----> all
 ------------------> ls_dir /d2/Articles//resume ~/Articles/DB/
 ------------------> update_DB /d2/Articles/DB/ ~/Articles//resume
 ------------------> make-html ~/Articles/DB/
 ------------------> lst_autres ~/Articles/DB/
 ------------------> lst_auth_ar ~/Articles/DB/
 ------------------> lst_auth_au ~/Articles/DB/
 ------------------> mix ~/Articles/DB/ ~/Articles/DB/
 ------------------> make-html2 ~/Articles/DB/ ~/Articles/DB//Auteurs/
-----> exit
12 instructions
STR successfully executed
-->
```

2.9 The command 'clean' – It can be given to **STR** to delete all the files produced by it (but not the documents !).

3 The visualisation programs

The program **xxw** can be used to display an article knowing its number that must be given in the top row if it is an article from ArXiv and in the bottom line if it is another article.



The program **See** is used as follows: to view an article of ArXiv, execute the command (in a terminal)

See NNN

where NNN is the number of the article, and to view another article, type

See -a NNN

where NNN is the number of the article.

4 Installation

STR has been developped on a Linux system.

To compile **STR**, type make. The program uses the library readline, which should be present in every Linux distribution.

To compile the visualisation program **xxw**, go in the directory xxw, and do

./compile

The program uses the libraries of X11 and gtk2.

To compile the visualisation program **See**, go in the directory See and do

gcc See.c -o See

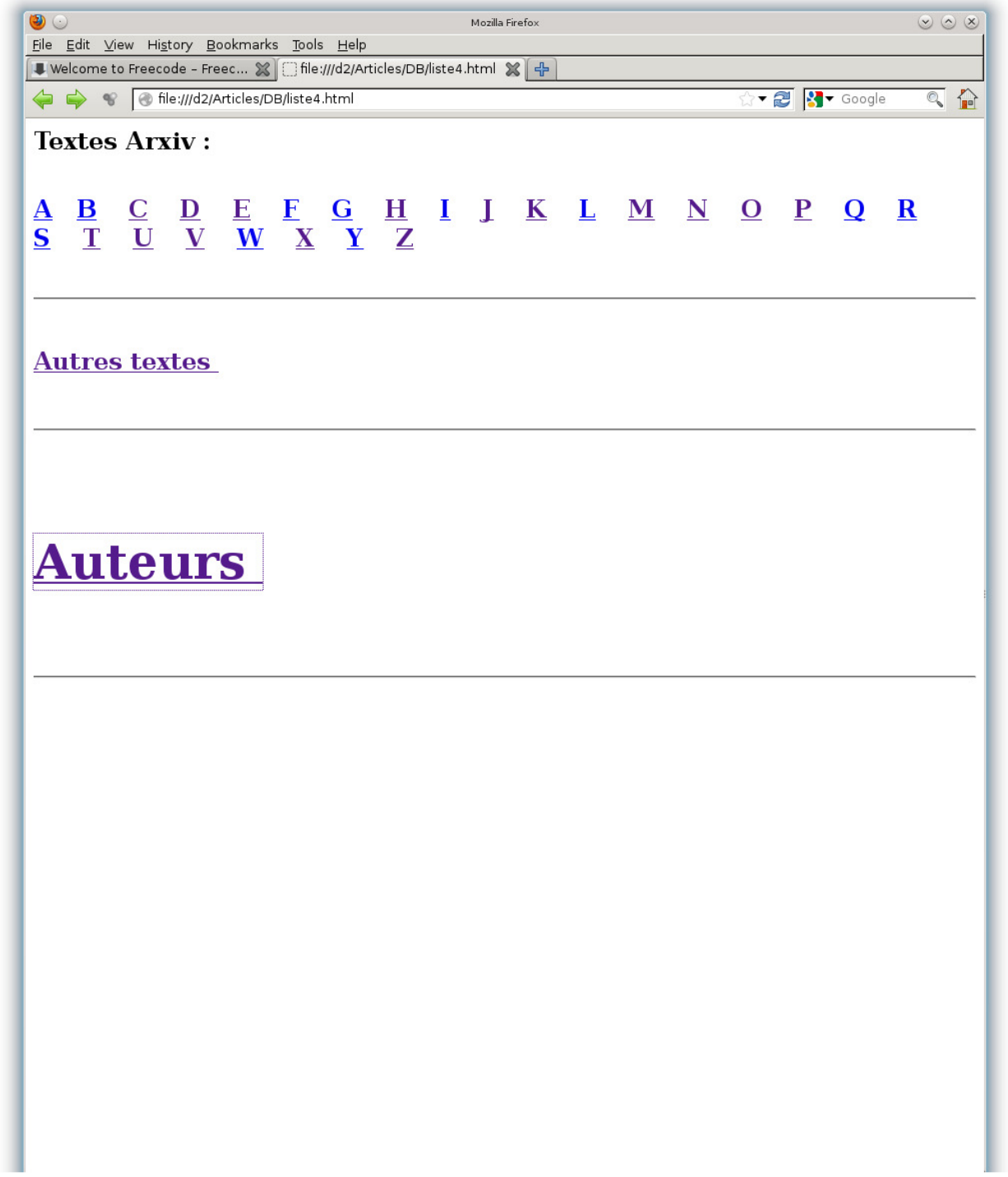


figure 1 – Initial web page

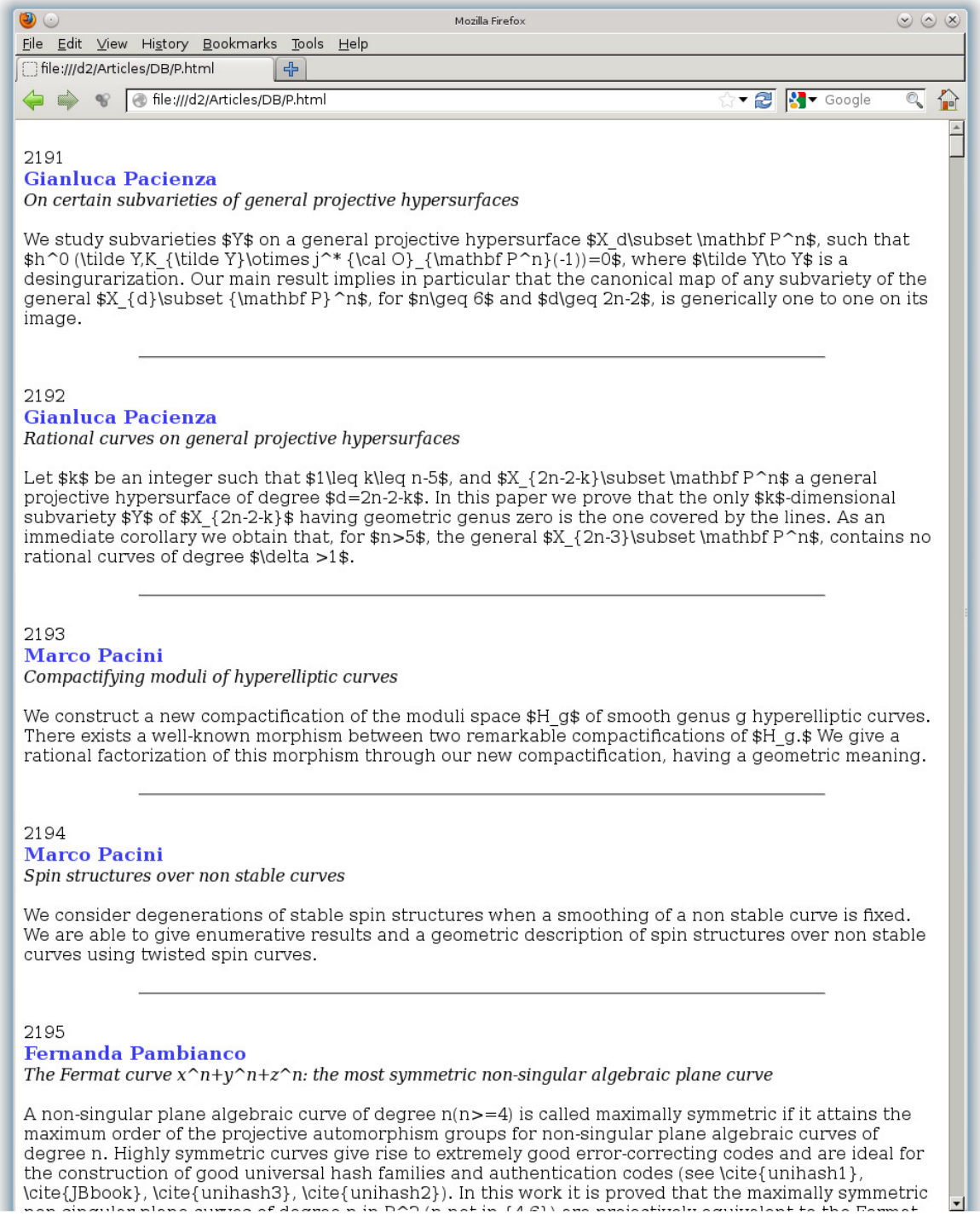


figure 2 – Articles from ArXiv

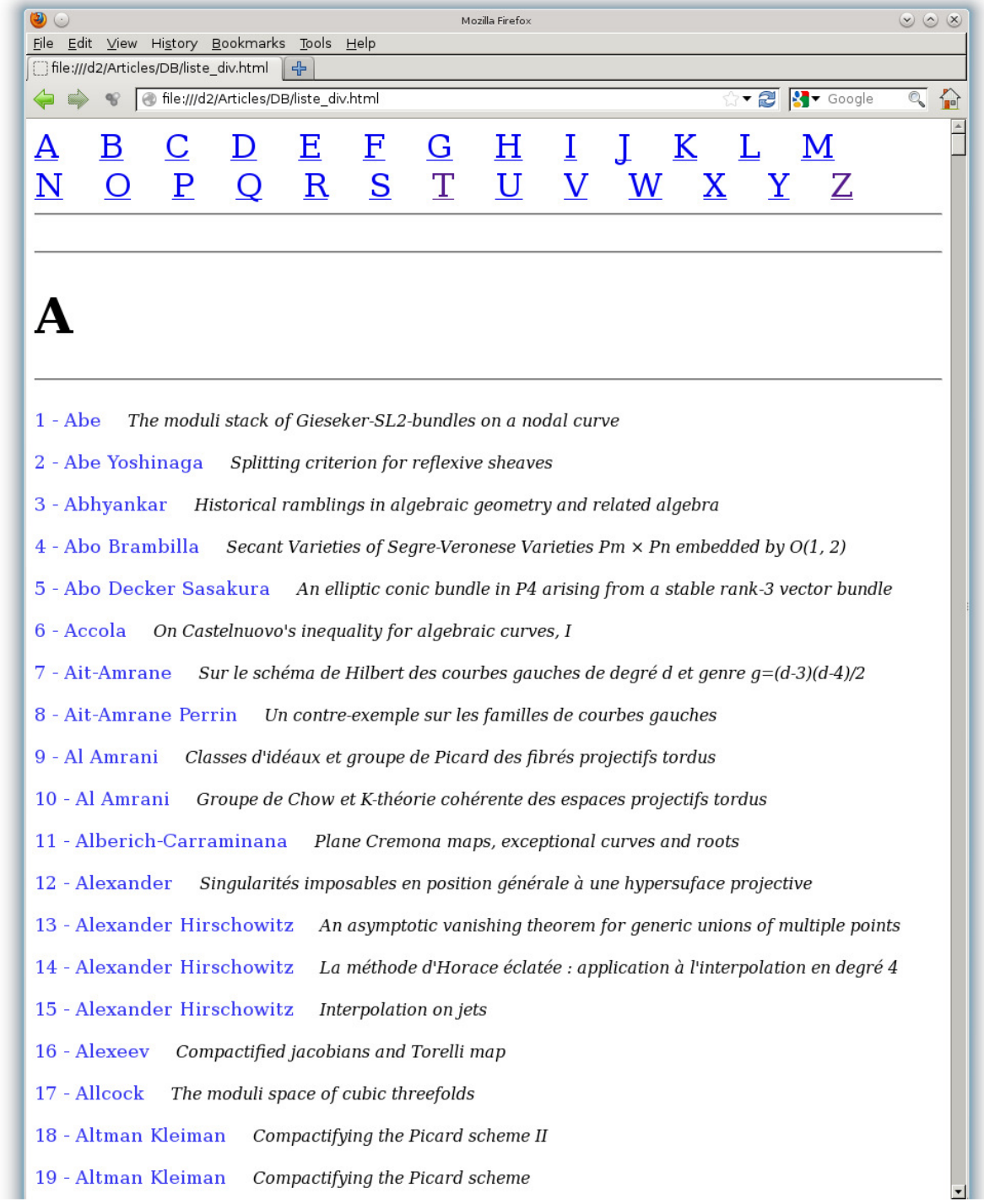


figure 3 – Other articles

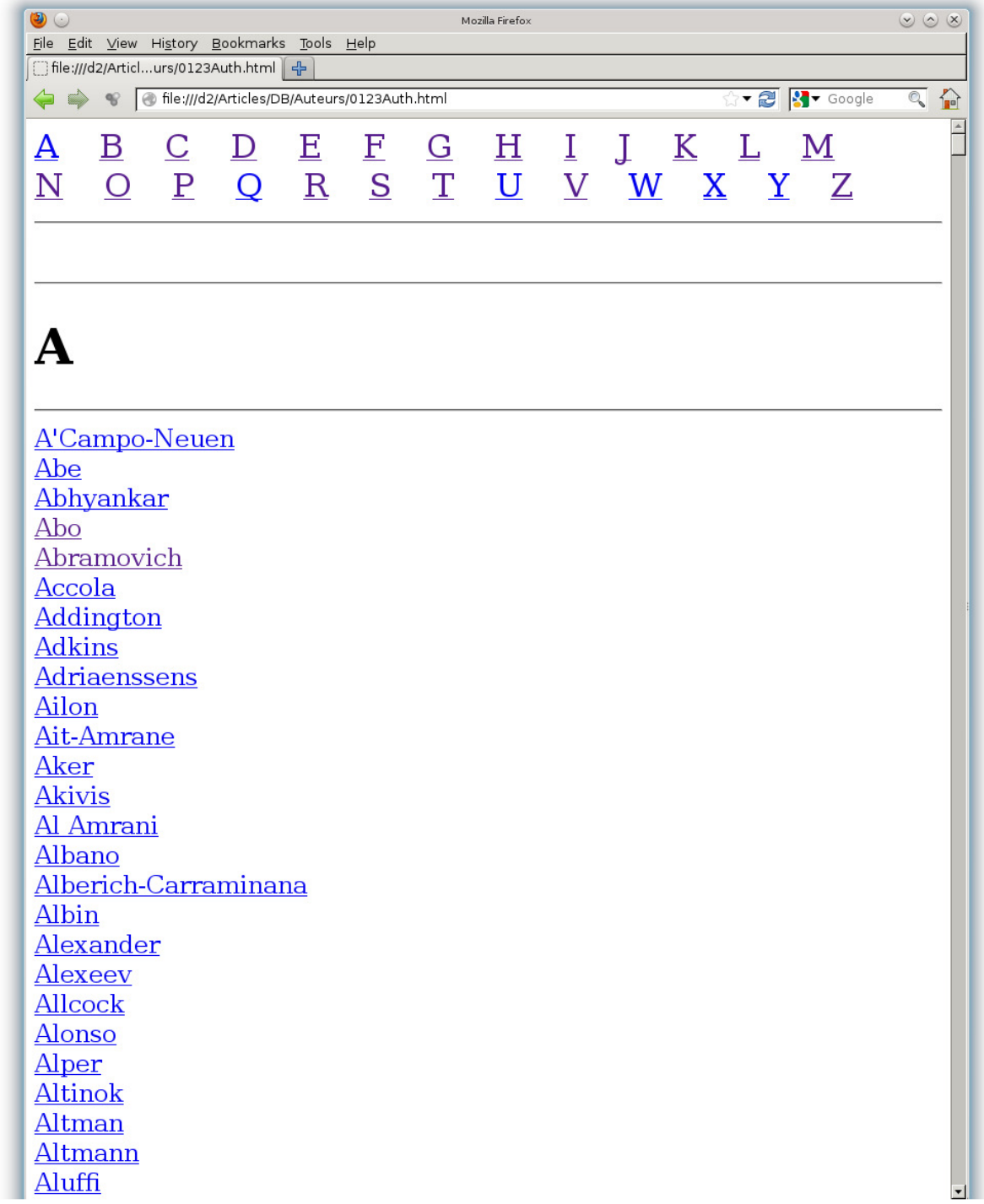


figure 4 – List of the authors

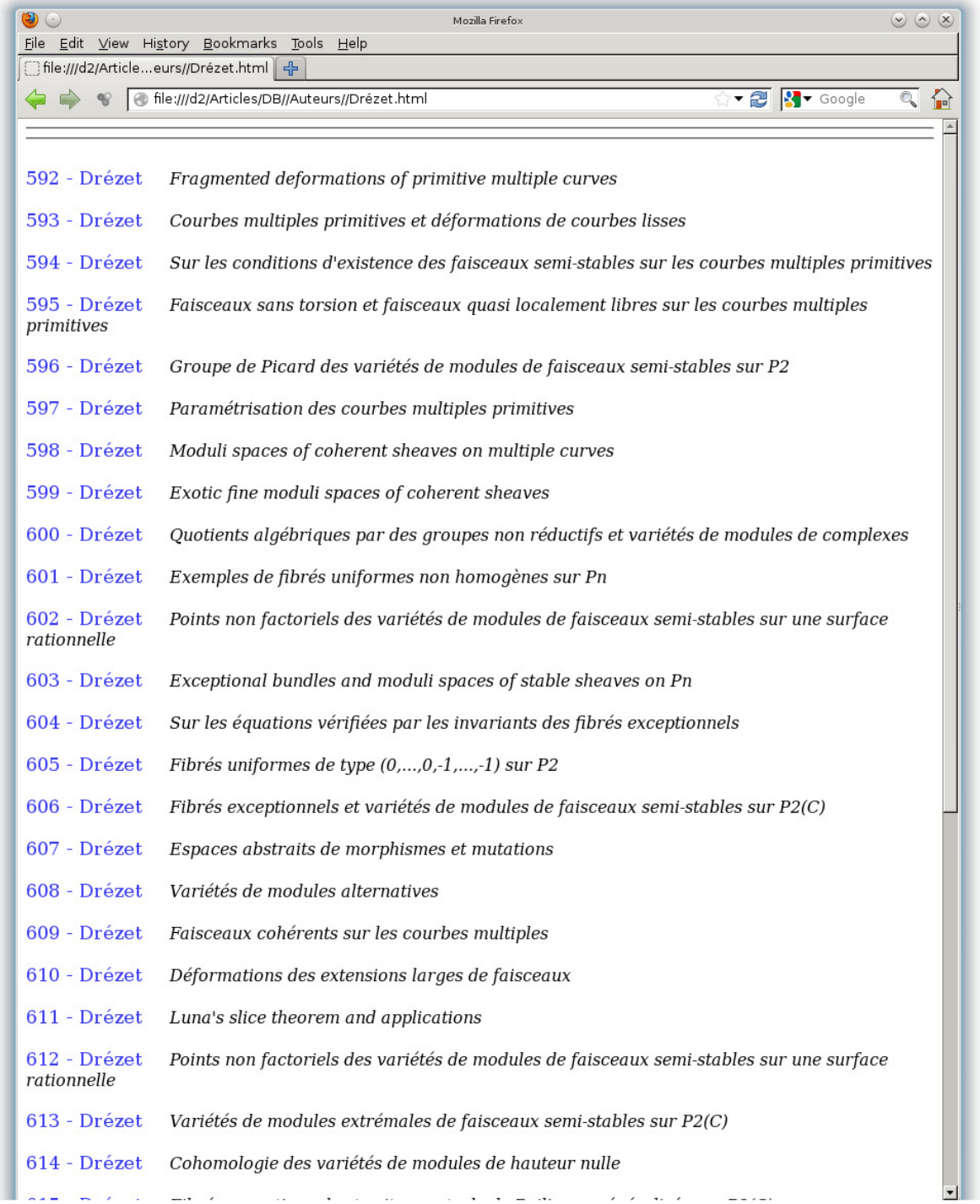


figure 5 – Liste of the articles of an author