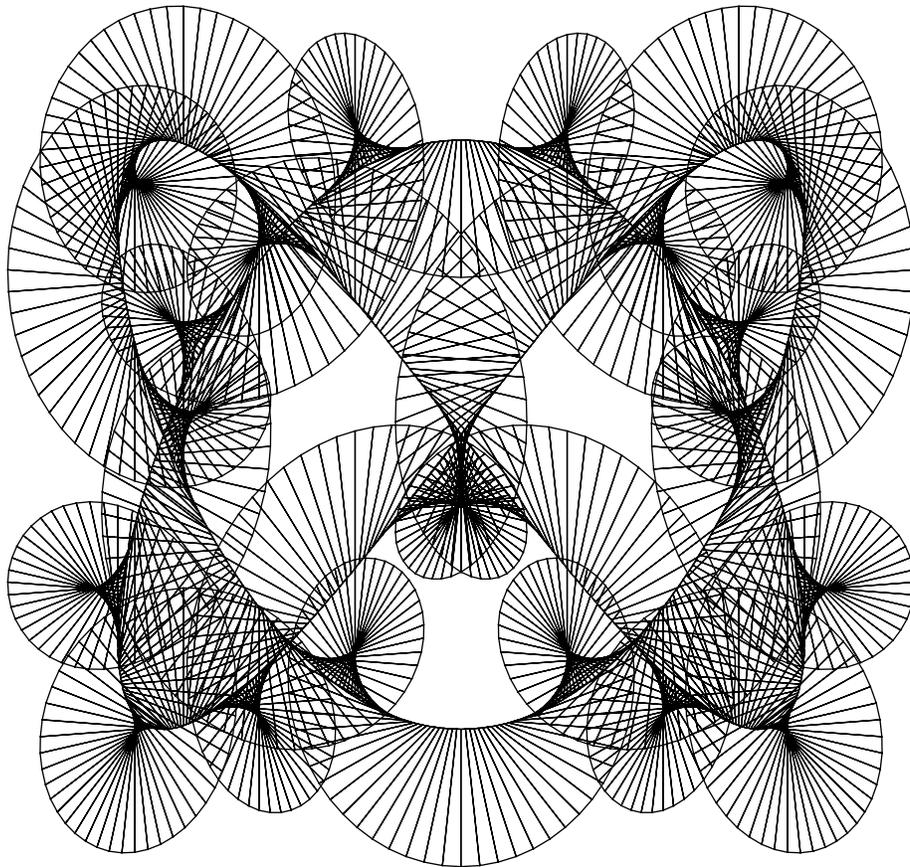


STR

Archivage et consultation de documents mathématiques
(version 3)

Jean-Marc Drézet



Sommaire

1	Description succincte	3
2	Le programme STR en détail	3
3	Les programmes de visualisation	8
4	Installation	8

1 Description succincte

STR est un programme pour le système d'exploitation **Linux**, destiné à produire des pages Web décrivant un ensemble de documents mathématiques en donnant leur liste, et en leur attribuant chacun un numéro permettant de les afficher facilement. Deux types de documents sont distingués : les articles provenant d'ArXiv, et les autres documents. La description des premiers comporte les noms des auteurs, le titre de l'article et le résumé associé donné par les auteurs. La description des seconds ne comporte que le nom des auteurs et le titre.

La page principale (figure 1) permet d'accéder à tous les textes stockés.

Chaque lettre de la rubrique **Textes Arxiv** est un lien vers la liste alphabétique des articles dont le nom du premier auteur commence par la lettre en question, suivie de la liste des articles dont un autre auteur a un nom commençant par cette lettre (voir figure 2).

La page correspondant à **Autres textes** donne la liste alphabétique des autres articles (voir figure 3).

La page correspondant à **Auteurs** donne la liste des auteurs d'articles. En cliquant sur un nom d'auteur on est amené à la liste des articles de cet auteur, aussi bien les textes ArXiv que les autres (voir figures 4 et 5).

Le programme graphique **xxw** permet d'afficher un article connaissant son numéro. Le programme en ligne de commande **See** fait la même chose.

2 Le programme STR en détail

Le programme **STR** permet de générer les pages web et les numéros d'articles associés en fonction des informations données par l'utilisateur. Dans le cas d'un article ArXiv, le résumé de l'article (c'est-à-dire la page web du résumé enregistrée) doit être placée dans un répertoire particulier, de même que le fichier en Postscript (compacté) contenant l'article. Pour les autres articles, des informations doivent être écrites dans un fichier spécial (nom du fichier contenant l'article, noms d'auteurs, titre et type de fichier).

2.1 Préparation des répertoires – Les fichiers contenant les documents doivent se trouver dans un répertoire unique, comportant deux sous-répertoires nommés **divers** et **resume**. Le répertoire principal contiendra les fichiers en Postscript des articles d'ArXiv, compactés. Le répertoire **divers** contiendra les autres documents. Le répertoire **resume** contiendra les résumés des articles d'ArXiv.

Un autre répertoire, contenant les fichiers produits et certains fichiers utilisés par **STR** doit être aussi créé (il peut-être un sous-répertoire du répertoire principal précédent). Il doit contenir un sous-répertoire nommé **Auteurs**. On y placera aussi le fichier **liste4.html** qui est la page

principale de la figure 1.

Un autre répertoire enfin, doit être créé, qui contiendra les fichiers temporaires utilisés lors de la visualisation des articles.

2.2 Le fichier d’initialisation – Au démarrage, **STR**, ainsi que les programmes de visualisation, lisent le fichier `.listerc3` qui doit se trouver dans le répertoire de l’utilisateur. Il contient 13 lignes donnant les informations suivantes :

Ligne 1 : *nom du répertoire contenant les fichiers produits par STR et certains fichiers utilisés par lui* – Exemple : `~/Articles/DB`

Ligne 2 : *nom du répertoire contenant les articles* – Exemple : `~/Articles`

Ligne 3 : *nom du suffixe des fichiers d’articles compactés d’ArXiv* – Exemple : `ps.bz2`

Ligne 4 : *nom du suffixe des articles écrits en Postscript, une fois qu’ils sont décompactés* – Exemple : `ps`

Ligne 5 : *nom du suffixe des fichiers d’articles divers en dvi compactés* – Exemple : `dvi.bz2`

Ligne 6 : *nom du suffixe des fichiers d’articles divers en dvi, une fois qu’ils sont décompactés* – Exemple : `dvi`

Ligne 7 : *nom du répertoire temporaire, suivi d’un préfixe* – Exemple : `~/tmp/XXX`. Ici “`~/tmp`” est le nom du répertoire devant être créé (cf. 2.1).

Ligne 8 : *nom du programme de décompression* – Exemple : `bunzip2`

Ligne 9 : *nom du programme de visualisation des articles en Postscript d’ArXiv* – Exemple : `okular`

Ligne 10 : *nom du programme de visualisation des articles divers en Postscript* – Exemple : `okular`

Ligne 11 : *nom du programme de visualisation des articles divers en dvi* – Exemple : `okular`

Ligne 12 : *nom du programme de visualisation des articles divers en djvu* – Exemple : `djview`

Ligne 13 : *nom du programme de visualisation des articles divers en pdf* – Exemple : `okular`

Dans les exemples précédents, le fichier `.listerc3` contiendra donc les lignes

`~/Articles/DB`

`~/Articles`

ps.bz2
ps
dvi.bz2
dvi
~/tmp/XXX
bunzip2
okular
okular
okular
djview
okular

2.3 Format des documents – Les textes d’Arxiv – Les textes provenant d’ArXiv doivent être en Postscript compactés, de la forme *ZZZZZZ.suffixe*, où *ZZZZZZ* est un nom donné par l’utilisateur (par exemple *NomAuteur-05*), et *suffixe* est donné dans la ligne 3 du fichier d’initialisation *.listerc3* (cf. 2.2, donc dans l’exemple précédent, ce sera *NomAuteur-05.ps.bz2*). Le fichier correspondant contenant le résumé provenant d’ArXiv et sauvegardé, aura un nom avec le même préfixe *ZZZZZZ*, suivi de *.html* (dans l’exemple précédent ce sera donc *NomAuteur-05.html*).

2.4 Format des documents – Les autres textes – Les autres textes (non ArXiv) peuvent être en *Postscript compacté*, en *dvi compacté*, en *pdf* ou en *djvu*. Les noms de fichiers sont indifférents, à part les Postscript et dvi compactés, qui doivent se conformer aux indications données dans le fichier d’initialisation.

2.5 Compression et décompression – Il faut utiliser toujours le même programme pour compacter les articles en Postscript ou en dvi. Il doit produire des fichiers ayant des suffixes correspondant à ce qui est indiqué dans le fichier d’initialisation *.listerc3* (cf. 2.2).

2.6 Liste des articles – Un fichier contenant la liste des articles ne provenant pas d’ArXiv et des informations les concernant doit être créé et mis à jour à chaque fois qu’un nouvel article est ajouté. Le nom de ce fichier est *liste_div.lst*, et il doit être situé dans le répertoire des fichiers produits par **STR**, mentionné dans la première ligne du fichier d’initialisation *.listerc3* (cf. 2.2). Chaque article est décrit par 4 lignes d’informations, et chacun de ces blocs de 4 lignes doit être séparé du suivant par une ligne vide. Les 4 lignes contiennent les informations suivantes :

Ligne 1 : *nom du fichier contenant l’article*

Ligne 2 : *noms des auteurs de l’article*. Voir plus loin.

Ligne 3 : *titre de l'article*

Ligne 4 : *format de l'article*. C'est `ps`, `dvi`, `pdf` ou `djvu`.

Si le nom d'un auteur a plusieurs parties, elles doivent être séparées par le caractère `#` au lieu d'espaces, sinon ces différentes parties seront considérées comme des noms d'auteurs différents.

Exemple :

```
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 – Certains noms d'auteurs (ceux d'article provenant d'ArXiv, récupérés automatiquement à partir des résumés) peuvent apparaître écrits de différentes façons, en particulier s'ils ont des accents, et être comptés comme des noms d'auteurs différents. Pour éviter cet inconvénient, il est possible d'indiquer dans un fichier les différents noms d'un même auteur. C'est le fichier `alias.lst`, situé dans le répertoire des fichiers produits et utilisés par **STR**, et mentionné dans le fichier d'initialisation `.listerc3` (cf. 2.2). Ce fichier est constitué de listes (une par auteur) séparées par une ligne contenant le caractère unique `#`. La première ligne d'une liste est le vrai nom d'auteur, les autres comportent les autres noms pouvant apparaître. *Note:* le principal (premier) nom d'auteur de chaque liste doit être présent au moins une fois dans un des articles.

Exemple :

```
Dupré
Dupre
Dupr
#
Möppe
Moppe
Moeppe
```

Mppe

2.8 Mise en oeuvre pratique – Avec les paramètres du fichier d’initialisation `.listerc3` donné en 2.2, il faut procéder de la manière suivante pour ajouter un nouvel article :

Pour ajouter un nouvel article provenant d’ArXiv, télécharger l’article en Postscript (et le compacter) ainsi que son résumé, leur nom devant avoir le même préfixe, par exemple `Article23.ps.bz2` sera le nom de l’article compacté et `Article23.html` celui de son résumé. L’article sera placé dans le répertoire `~/Articles` et le résumé dans `~/Articles/resume`.

Pour ajouter un autre article d’une autre provenance, il faudra qu’il soit dans un des formats supportés (Postscript compacté, dvi compacté, pdf ou djvu). Il sera placé dans le répertoire `~/Articles/divers`. Le fichier `~/Articles/DB/liste.div.lst` devra être complété de quatre lignes décrivant l’article, comme indiqué en 2.6.

La mise à jour sera effectuée en exécutant le programme **STR**, à qui on donnera la commande `all`, suivie de `exit` :

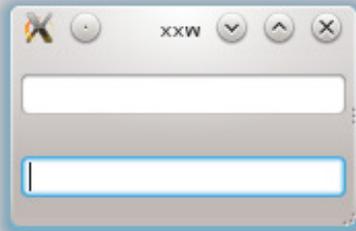
```
--> STR
```

```
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 La commande 'clean' – Elle peut être donnée à **STR** pour supprimer tous les fichiers qu’il a produits (mais pas les documents !).

3 Les programmes de visualisation

Le programme **xxw** permet d'afficher un article connaissant son numéro, qu'il faut donner dans la ligne du haut s'il s'agit d'un article d'ArXiv et dans la ligne du bas s'il s'agit d'un autre article.



Le programme **See** s'utilise de la façon suivante : pour visualiser un article d'ArXiv, exécuter la commande

```
See NNN
```

où NNN est le numéro de l'article, et pour visualiser un autre article, exécuter la commande

```
See -a NNN
```

où NNN est le numéro de l'article.

4 Installation

STR a été mis au point sur un système Linux.

Pour compiler **STR** taper **make**. Le programme nécessite la librairie **readline** qui se trouve normalement dans toutes les distributions.

Pour compiler le programme de visualisation **xxw**, se placer dans le répertoire **xxw**, et faire **./compile**

Le programme utilise les librairies de X11 et **gtk2**.

Pour compiler le programme de visualisation **See**, se placer dans le répertoire **See**, et faire **gcc See.c -o See**



figure 1 – Page web initiale

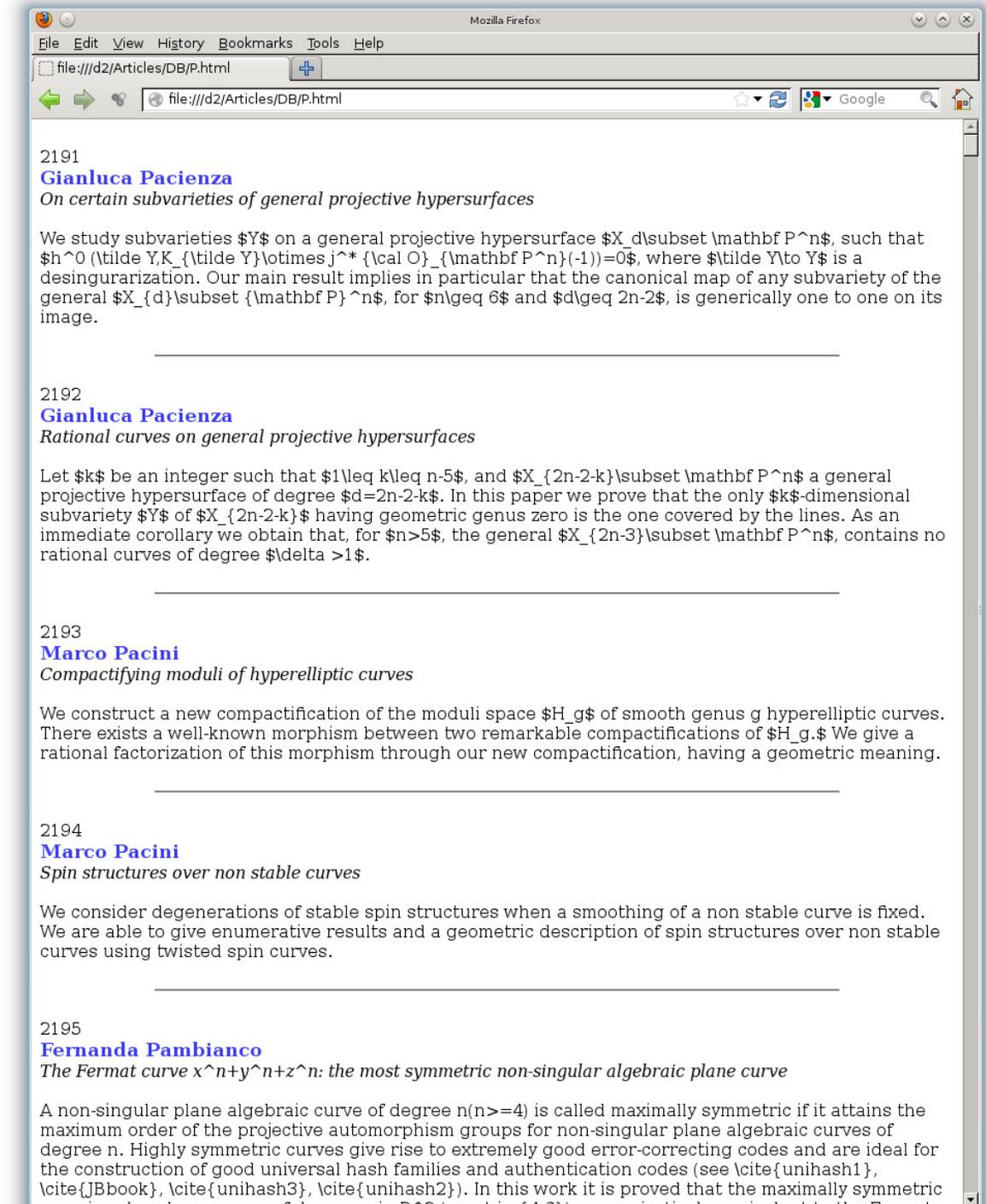


figure 2 – Articles d'ArXiv

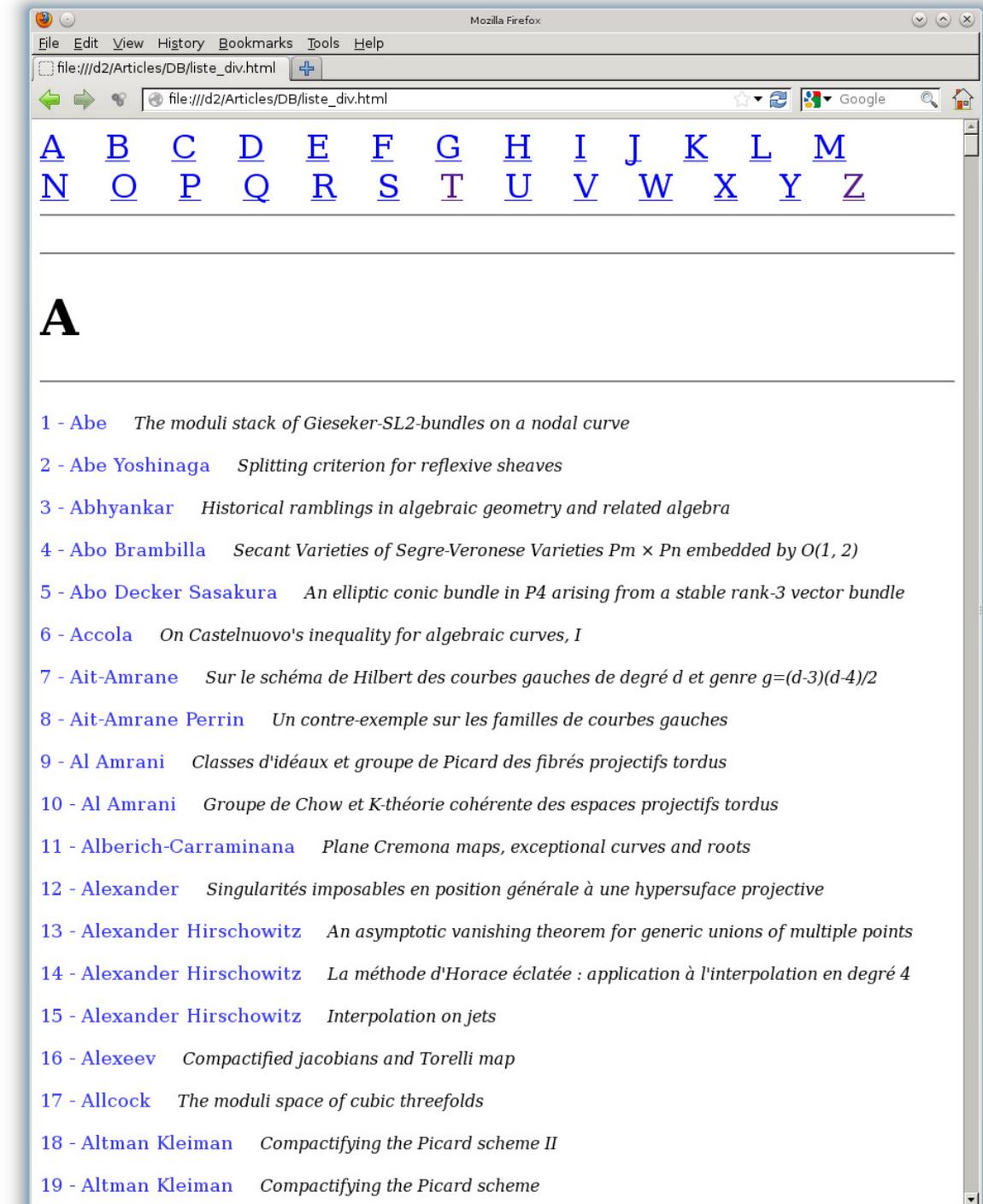


figure 3 – Autres articles



figure 4 – Liste des auteurs

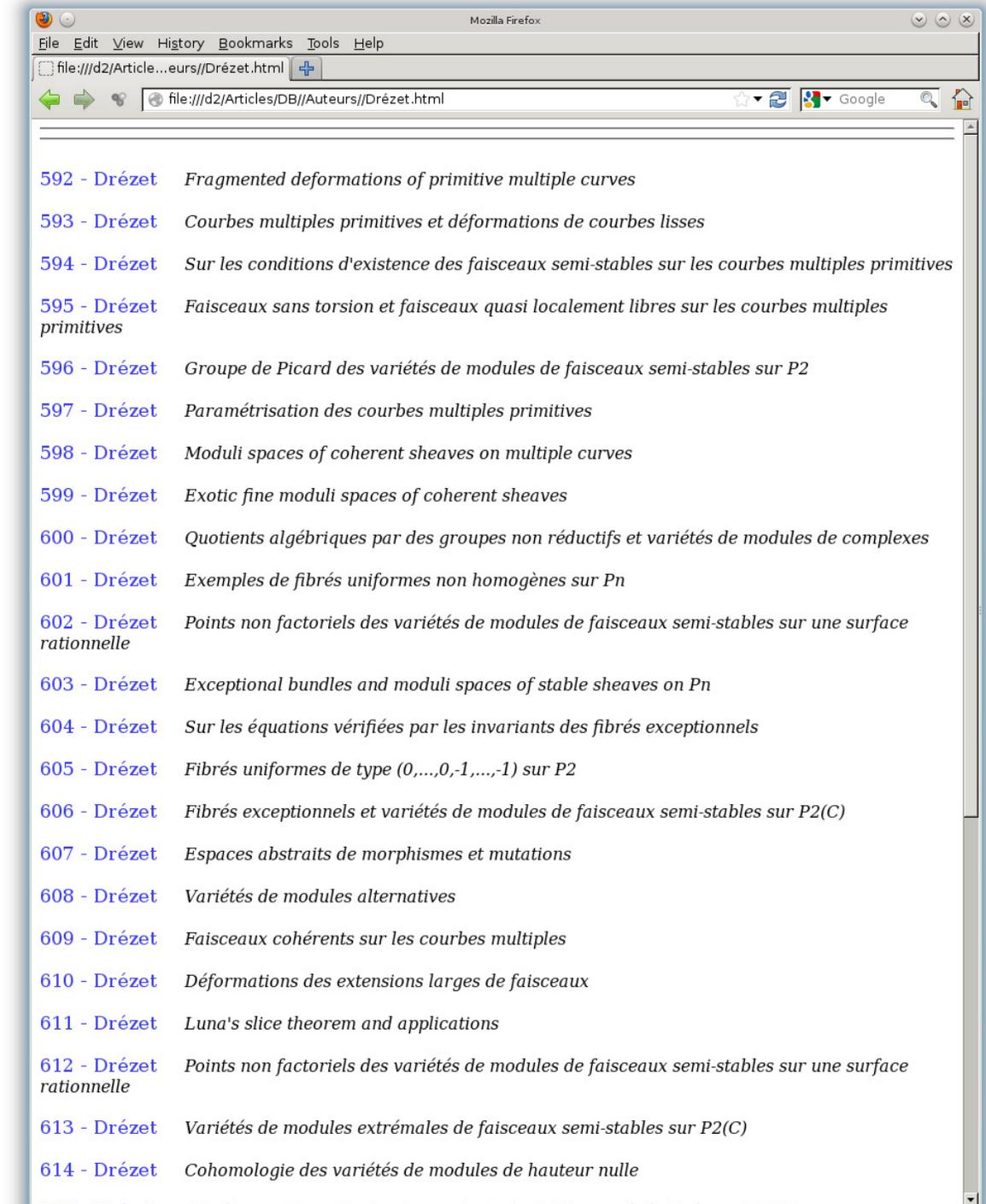


figure 5 – Liste des articles d'un auteur