

Un algorithme de composition musicale

Table des matières

1	Présentation	2
2	Le compositeur	2
2.1	Le code PMX	2
2.2	Structures de données utilisées	3
2.3	L'algorithme	4
3	Chaînes de Markov	5
3.1	Modélisation	5
3.2	Nombre moyen de mesures générées par le compositeur	5
3.3	Étude d'un exemple	6
4	Résultats	6
5	Améliorations possibles	6
6	Applications pratiques	6
7	Bibliographie et sites Internet	7

1 Présentation

L'idée de ce TIPE m'est venue d'un algorithme de génération de texte à partir d'une base de livres. Le concept est le suivant : à partir d'un mot m de cette base, l'algorithme affiche un mot m' au hasard suivant m dans l'un des livres, puis fait de même en partant de m' . De proche en proche, une phrase est écrite. Ainsi construite, il y a peu de risques qu'elle figure dans l'un des livres.

J'ai voulu appliquer cet algorithme à une base de partitions de musique, où les "mots" seraient les mesures des partitions, c'est-à-dire des ensembles de notes de musique.

Le processus se stabilise sous certaines conditions, fournissant alors une partition nouvelle. Après avoir détaillé le fonctionnement de ce compositeur, on tentera d'obtenir des informations sur cette partition "limite".

2 Le compositeur

2.1 Le code PMX

Le PMX (Preprocessor for MusiXTeX) est un langage permettant d'éditer des partitions. Il s'agit ici de partitions de piano, composées de plusieurs systèmes de deux lignes : une pour la main droite (celle du haut) et une pour la main gauche. Ces lignes sont elles-mêmes constituées de mesures.

Par exemple, pour cette partition :

Au clair de la lune



Le code PMX correspondant est le suivant :

```
2 1 4 4 4 4 0 0
1 1 29 0.085
Piano
bt
./
It120ipipi
Tt
Au clair de la lune
c03 | g0+ | c2- g | c0 | d | f- | d+ | g- /
c44 c c d | e44 r d r | c44 e d d | c44 r r r | d44 d d d | a43 r a r | d44 c b a | g43 r r r /
```

Ici l'intérêt se porte sur les deux dernières lignes, les autres ne servant qu'à donner des indications de style. Chaque mot représente une note, chaque barre verticale une séparation entre deux mesures. Dans le code, la ligne du bas correspond à la main droite.

On peut compiler un fichier .pmx pour obtenir un fichier .pdf comportant la partition, ainsi qu'un fichier .mid, permettant de l'écouter.

2.2 Structures de données utilisées

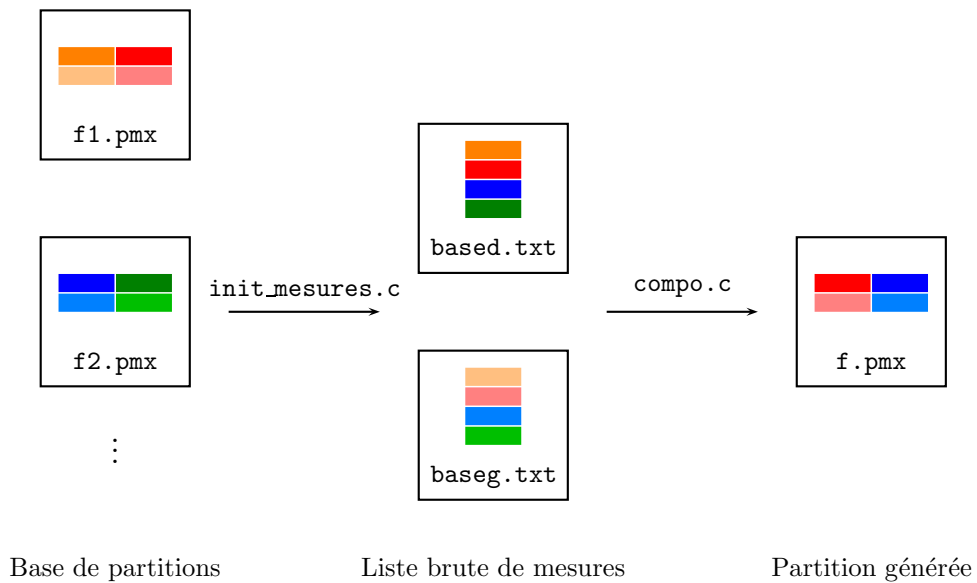


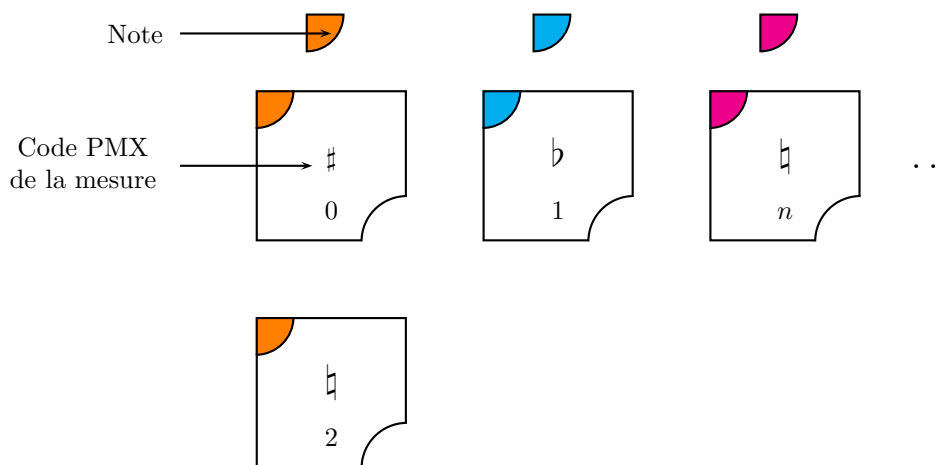
Schéma principal

On représente ici les mesures par des barres de couleur.

Un premier programme codé en C nommé `init_mesures.c` va, à partir d'une liste de fichiers `.pmx`, récupérer toutes les mesures de la partie de la main droite (en conservant leur ordre) et les insérer dans un fichier `based.txt` (de même pour celles de la main gauche, dans `baseg.txt`).

À partir de cette liste brute de mesures, un deuxième programme, `compo.c`, va créer une nouvelle partition. Au fur et à mesure de la composition, on veillera à ce que pour chaque mesure de la main droite, la main gauche qui lui est associée dans la nouvelle partition soit la même que dans la partition initiale. D'où l'intérêt de mettre les mesures de la main gauche et de la main droite dans deux bases distinctes.

Pour que la composition soit effectuée, il faut créer au préalable des classes de mesures, qui pour une note donnée nous permettent de savoir toutes les mesures commençant par celle-ci.



Classes de mesures

Et pour chaque mesure, il faut avoir accès à la première note de la mesure qui la suit dans la partition initiale ; sauf pour la dernière mesure d'une partition, pour laquelle on choisit d'attribuer, comme "première note de la mesure qui la suit", une note appelée "note stop". Cette note mettra fin à l'algorithme, d'où son nom.

2.3 L'algorithme

À partir d'une note `noteInitiale` reçue en paramètre, on applique l'algorithme suivant :

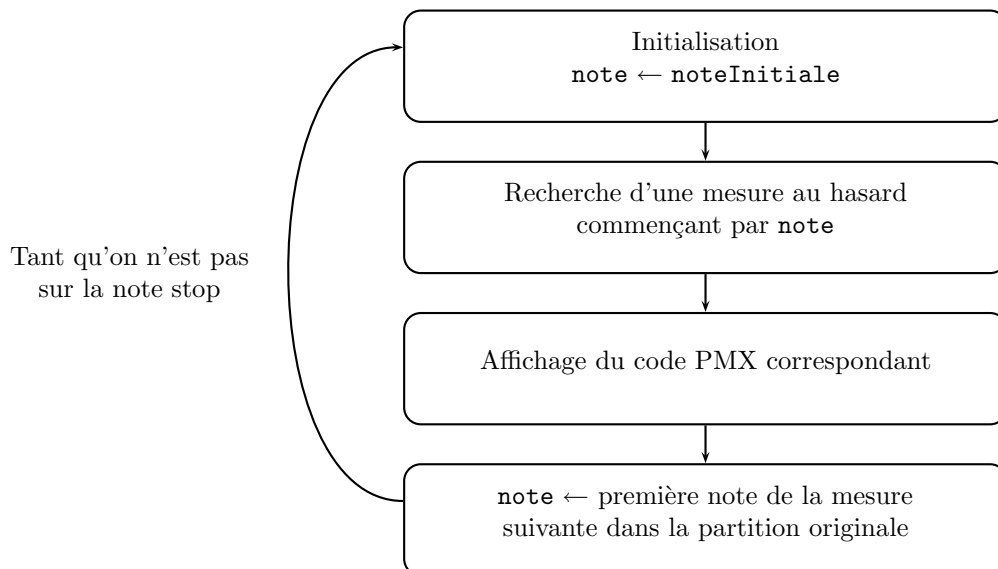
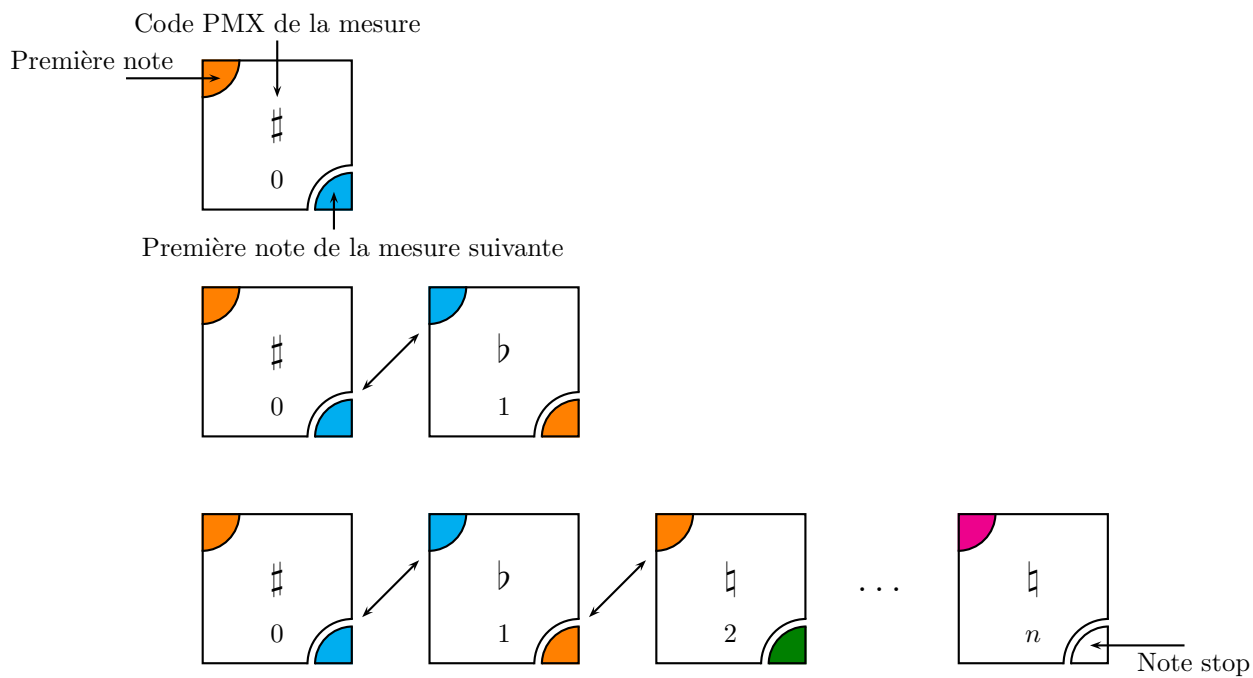


Schéma du fonctionnement de l'algorithme

Sur un exemple, cela donne :



3 Chaînes de Markov

3.1 Modélisation

Le système permettant de composer des partitions peut être représenté par une chaîne de Markov d'ordre 1, à temps discret, définie sur l'espace d'états fini des premières notes des mesures, et homogène dans le temps.

L'unique état d'absorption de cette chaîne est la note stop.

3.2 Nombre moyen de mesures générées par le compositeur

Soit $M = [m_{i,j}]$ la matrice telle que $m_{i,j}$ est égal à la probabilité de passer d'une mesure commençant par la note n_i à une autre commençant par la note n_j . C'est la matrice de transition.

On peut en réordonner la base, pour obtenir une matrice de transition dite de forme canonique.

$$M = \begin{matrix} & \begin{matrix} n_1 & n_2 & n_3 & n_4 & n_5 \end{matrix} \\ \begin{matrix} n_1 \\ n_2 \\ n_3 \\ n_4 \\ n_5 \end{matrix} & \begin{bmatrix} 0.1 & 0.2 & 0.3 & 0.4 & 0 \\ & & \vdots & & \\ 0 & 0 & 1 & 0 & 0 \\ & & \vdots & & \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix} \end{matrix} \rightarrow \begin{bmatrix} I & 0 \\ R & Q \end{bmatrix}$$

Matrice de transition, sous forme canonique à droite

Soit N la matrice fondamentale associée, donnée par la relation :

$$N = \lim_{n \rightarrow \infty} \sum_{k=0}^n Q^k = (I - Q)^{-1}$$

On peut montrer que :

$$N = \begin{matrix} & \begin{matrix} n_1 & n_2 & n_4 & n_5 \end{matrix} \\ \begin{matrix} n_1 \\ n_2 \\ n_4 \\ n_5 \end{matrix} & \begin{bmatrix} & & & \\ & & & \\ & & & \\ & & & \end{bmatrix} \end{matrix}$$

Matrice fondamentale

Par conséquent, si l'on appelle O la somme des colonnes de la matrice N , alors pour tout i , l'élément $o_{i,1}$ est le nombre moyen de mesures générées par l'algorithme, si la note initiale est n_i .

3.3 Étude d'un exemple

L'annexe 1 est une matrice de transition obtenue à partir d'une base de deux partitions de musique. Cette matrice est en réalité de grande taille, étant donné que, par exemple, on distingue les notes blanches des notes noires. Mais pour plus de clarté, ici on ne représentera que les noms des notes.

On peut voir que si la note initiale est un la, le nombre moyen de mesures générées est de 18.

4 Résultats

L'annexe 2 est un exemple de partition générée par l'algorithme. La base de partitions était constituée de deux partitions, qu'on a représentées en bleu et rose.

- Lorsque deux mesures consécutives font partie de la même zone de couleur, cela signifie qu'elles se suivaient déjà dans la partition initiale.
- Si elles font parties de zones disjointes de même couleur, c'est qu'elles proviennent de la même partition initiale, mais qu'elles n'étaient pas dans cet ordre-là dans celle-ci.
- Enfin, si on passe d'une zone de couleur à une autre, c'est qu'on n'est plus dans la même partition.

5 Améliorations possibles

Pour générer une phrase, on peut partir d'un ensemble de deux mots m_1 et m_2 , en choisir au hasard un troisième m_3 qui les suit dans la base de livres (supposée suffisamment grande pour qu'il y ait plusieurs occurrences de ces deux mots), puis répéter le processus sur les mots m_2 et m_3 , etc.

Par analogie, on peut partir d'un ensemble de deux notes n_1 et n_2 , qui seront les premières notes de deux mesures, puis choisir une troisième mesure au hasard parmi celles qui suivent une mesure commençant par n_1 et une mesure commençant par n_2 , et ainsi de suite. Ce serait alors une chaîne de Markov d'ordre 2.

La partition générée sera alors plus jolie à écouter, mais si la base de partitions n'est pas très grande, on risque de se retrouver avec une mélodie qui figure déjà dans l'une des partitions initiales.

Pour obtenir une musique qui paraît totalement nouvelle, on pourrait, au lieu de créer la partition mesure par mesure, l'écrire note par note. Mais dans ce cas, il faudrait définir des règles plus précises de composition, sinon la musique ne serait pas agréable à écouter. Il faudrait aussi s'assurer que la partition ainsi construite soit correcte syntaxiquement (c'est-à-dire que les règles de solfège de base soient respectées), sinon le fichier PMX ne pourra être compilé.

6 Applications pratiques

Si on choisit de remplacer la note stop par une note au hasard, on peut composer de la musique indéfiniment (en effet, la note stop est la seule condition d'arrêt du programme), ce qui serait intéressant pour créer une musique d'ambiance dans un lieu public, un film ou un jeu vidéo.

Par ailleurs, on pourrait définir un critère de ressemblance entre deux partitions à partir de leurs matrices de transition. En effet, si deux partitions de même tonalité se ressemblent, leurs matrices de transition sont similaires. En exploitant davantage cet élément statistique qu'est cette matrice, on pourrait peut-être dépister un éventuel plagiat.

7 Bibliographie et sites Internet

Édition musicale avec PMX

Luigi Cataldi, traduction par Olivier Vogel.

PSTricks User's Guide

Timothy Van Zandt.

<http://apprendre-en-ligne.net/graphes/markov/limite.html>

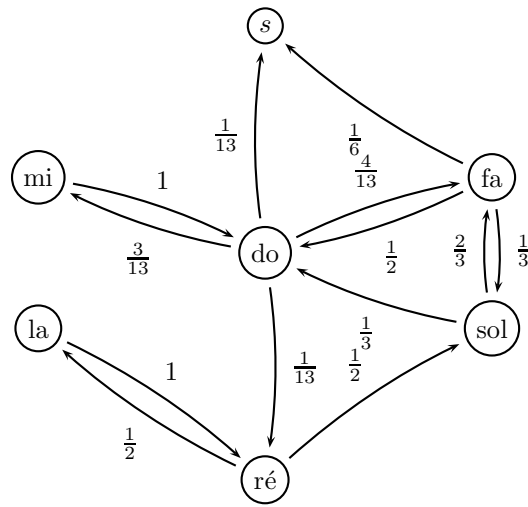
“Introduction aux chaînes de Markov” par Didier Müller.

<http://www.ump.ma/mounir/anits/proc%20stock/markov.pdf>

Cours sur les chaînes de Markov.



Annexe 1 - Étude d'un exemple



Diagramme

	s	do	ré	mi	fa	sol	la	
$M =$	1	0	0	0	0	0	0	
do	$\frac{1}{13}$	$\frac{4}{13}$	$\frac{1}{13}$	$\frac{3}{13}$	$\frac{4}{13}$	0	0	do
ré	0	0	0	0	0	$\frac{1}{2}$	$\frac{1}{2}$	ré
mi	0	1	0	0	0	0	0	mi
fa	$\frac{1}{6}$	$\frac{1}{2}$	0	0	0	$\frac{1}{3}$	0	fa
sol	0	$\frac{1}{3}$	0	0	$\frac{2}{3}$	0	0	sol
la	0	0	1	0	0	0	0	la

Matrice de transition, somme des colonnes de la matrice N

Annexe 2 - Exemple de partition

The image displays a musical score for piano, organized into six systems. Each system consists of a treble and bass staff. The measures are numbered as follows:

- System 1:** Measures 53-58. Measure 53 is highlighted in pink.
- System 2:** Measures 59-63. Measure 59 is highlighted in pink.
- System 3:** Measures 64-68. Measure 64 is highlighted in pink.
- System 4:** Measures 69-74. Measure 69 is highlighted in pink.
- System 5:** Measures 75-80. Measure 75 is highlighted in pink.
- System 6:** Measures 81-86. Measure 81 is highlighted in pink.

The score includes various musical notations such as notes, rests, and dynamic markings. A '8va' marking is present in the final system, indicating an octave shift. The background of the score is divided into blue and pink vertical sections.