Partitions musicales augmentées

D. Fober, C. Daudin, Y. Orlarey, S. Letz

Grame
Centre national de création musicale
Lyon - France

Avril 2010

GRAME CENTRE NATIONAL DE CREATION MUSICALE

Sommaire

Le projet Interlude.

Nouveaux paradigmes numériques pour l'exploration et l'interaction gestuelle expressive avec des contenus musicaux.

Domaines d'applications :

- professionnel (pédagogie, pièces interactives...)
- grand public (jeux musicaux...)

Partenaires :

- Ircam, Grame
- VoxLer, Dafact
- NoDesign, Atelier les Feuillantines

Le projet Interlude.

Nouveaux paradigmes numériques pour l'exploration et l'interaction gestuelle expressive avec des contenus musicaux.

Domaines d'applications :

- professionnel (pédagogie, pièces interactives...)
- grand public (jeux musicaux...)

Partenaires :

- Ircam, Grame
- VoxLer, Dafact
- NoDesign, Atelier les Feuillantines

Le projet Interlude.

Nouveaux paradigmes numériques pour l'exploration et l'interaction gestuelle expressive avec des contenus musicaux.

Domaines d'applications :

- professionnel (pédagogie, pièces interactives...)
- grand public (jeux musicaux...)

Partenaires :

- Ircam, Grame
- VoxLer, Dafact
- NoDesign, Atelier les Feuillantines

Interaction avec des contenus symboliques.

Partition musicale augmentée.

- Une *partition musicale augmentée* est une partition mettant en relation un objet musical symbolique avec différentes représentations de son interprétation.
- La partition musicale est à considérer au sens large, comme un objet graphique permettant de représenter un objet temporel.
- L'interprétation représente une instance sonore ou gestuelle particulière de la partition.

Interaction avec des contenus symboliques.

Partition musicale augmentée.

- Une *partition musicale augmentée* est une partition mettant en relation un objet musical symbolique avec différentes représentations de son interprétation.
- La partition musicale est à considérer au sens large, comme un objet graphique permettant de représenter un objet temporel.
- L'interprétation représente une instance sonore ou gestuelle particulière de la partition.

Interaction avec des contenus symboliques.

Partition musicale augmentée.

- Une *partition musicale augmentée* est une partition mettant en relation un objet musical symbolique avec différentes représentations de son interprétation.
- La partition musicale est à considérer au sens large, comme un objet graphique permettant de représenter un objet temporel.
- L'interprétation représente une instance sonore ou gestuelle particulière de la partition.

Interaction avec des contenus symboliques.

Partition musicale augmentée.

- Une *partition musicale augmentée* est une partition mettant en relation un objet musical symbolique avec différentes représentations de son interprétation.
- La partition musicale est à considérer au sens large, comme un objet graphique permettant de représenter un objet temporel.
- L'interprétation représente une instance sonore ou gestuelle particulière de la partition.

Problématiques

Au coeur de la partition augmentée

- extension de la partition à des objets musicaux arbitraires
- expression de relations entre espaces graphiques et temporels
- représentation de l'interprétation (gestuelle, sonore)

Sommaire

Objets musicaux de première classe

Tous les composants de la partition :

- ont une dimension graphique,
- ont une dimension temporelle,
- sont adressables aussi bien dans l'espace graphique que temporel,
- gèrent les relations entre espace temporel et graphique,
- sont synchronisables dans l'espace graphique et temporel.

Composants

Typologie des ressources graphiques.

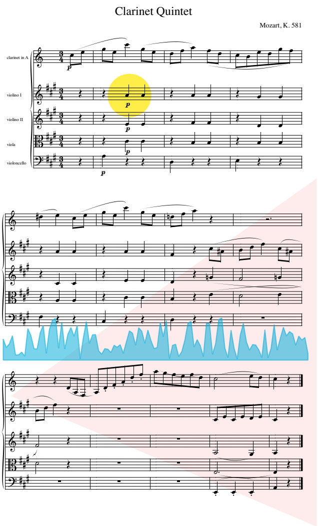
- Partitions musicales
  format GMN (Guido Music Notation format) ou MusicXML
- Eléments textuels
- Graphiques bitmaps (jpg, gif, tiff, png, ...)
- Graphiques vectoriels (rectangles, ellipses, ...)
- Représentations graphiques du son et du geste

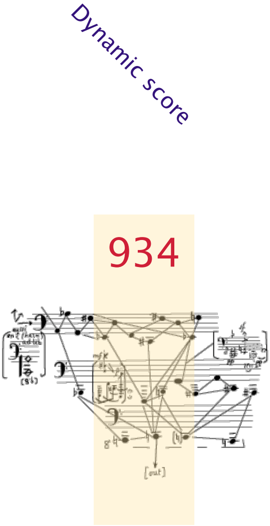
Paramètres de contrôle.

Paramètres de contrôle communs.

- position (x, y, z)
- échelle
- rotation
- couleur
- date
- durée
- visibilité

Exemple


Clarinet Quintet
Mozart, K. 581
clarinet in A
violino I
violino II
viola
violoncello


Dynamic score
934

Implémentation

- sous forme de librairie partagée C++.
- sous forme d'application : afficheur de partition augmentée.
- multi-platform [MacOS X, Linux, Windows].
- basé sur le framework Qt.
- basé sur le moteur Guido et la librairie libMusicXML.
- support du protocole OSC [oscpack].

Sommaire

Segments temporels

- Un *segment temporel* est défini comme un intervalle $i = [t_0, t_1[$ tel que $t_0 \leqslant t_1$.
- $i = [t_0, t_1[$ est dit vide quand $t_0 = t_1$.
  Il sera noté $\emptyset$.
- L'intersection de 2 segments temporels est le plus grand intervalle tel que :

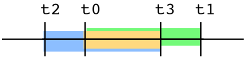
$$\forall i_m, \forall i_n, \ i_m \cap i_n := \{j \mid j \in i_m \ \wedge \ j \in i_n\}$$

Segments temporels

- Un *segment temporel* est défini comme un intervalle $i = [t_0, t_1[$ tel que $t_0 \leqslant t_1$.
- $i = [t_0, t_1[$ est dit vide quand $t_0 = t_1$.
  Il sera noté $\oslash$.
- L'intersection de 2 segments temporels est le plus grand intervalle tel que :

$$\forall i_m,\ \forall i_n,\ i_m \cap i_n := \{j \mid j \in i_m \ \wedge \ j \in i_n\}$$

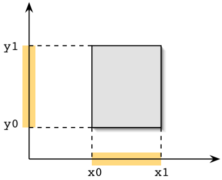
Segments temporels



- Un *segment temporel* est défini comme un intervalle $i = [t_0, t_1[$ tel que $t_0 \leqslant t_1$.
- $i = [t_0, t_1[$ est dit vide quand $t_0 = t_1$.
  Il sera noté $\oslash$.
- L'intersection de 2 segments temporels est le plus grand intervalle tel que :

$$\forall i_m,\ \forall i_n,\ \ i_m \cap i_n := \{j \mid j \in i_m\ \wedge\ j \in i_n\}$$

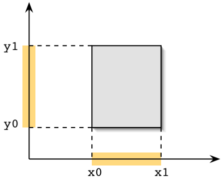
Segments graphiques



- Un *segment graphique g* est défini comme un rectangle donné par deux intervalles $g = (x, y)$ où x est un intervalle sur l'axe des abscisses et y, sur l'axe des ordonnées.
- $g = \{x, y\}$ est dit vide quand$x = \emptyset$ ou $y = \emptyset$
- Intersection $\cap$ entre segments graphiques :

$$\forall g_m = \{x_m, y_m\},\ \forall g_n = \{x_n, y_n\},\ g_m \cap g_n = \{x_m \cap x_n, y_m \cap y_n\}$$

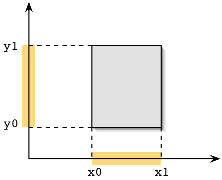
Segments graphiques

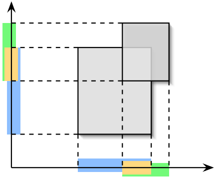


- Un *segment graphique* g est défini comme un rectangle donné par deux intervalles $g = (x, y)$ où x est un intervalle sur l'axe des abscisses et y, sur l'axe des ordonnées.
- $g = \{x, y\}$ est dit vide quand$x = \emptyset$ ou $y = \emptyset$
- Intersection $\cap$ entre segments graphiques :

$$\forall g_m = \{x_m, y_m\},\ \forall g_n = \{x_n, y_n\},\ g_m \cap g_n = \{x_m \cap x_n, y_m \cap y_n\}$$

Segments graphiques





- Un *segment graphique* g est défini comme un rectangle donné par deux intervalles $g = (x, y)$ où x est un intervalle sur l'axe des abscisses et y, sur l'axe des ordonnées.
- $g = \{x, y\}$ est dit vide quand$x = \oslash$ ou $y = \oslash$
- Intersection $\cap$ entre segments graphiques :

$$\forall g_m = \{x_m, y_m\},\ \forall g_n = \{x_n, y_n\},\ g_m \cap g_n = \{x_m \cap x_n, y_m \cap y_n\}$$

Généralisation de la notion de segment

- Un segment de dimension n, noté s^n, est défini comme une liste de n intervalles $s^n = (i_1, ..., i_n)$ où i_j est un intervalle sur la dimension j.
- Un segment s^n de dimension n est dit vide quand $\exists i \in s^n \mid i = \emptyset$.
- L'intersection de segments de dimension n est définie comme la liste des intersections de leurs intervalles :

$$s_1^n \cap s_2^n = (i_1 \cap j_1, ..., i_n \cap j_n)$$

où $s_1^n = (i_1, ..., i_n)$ et $s_2^n = (j_1, ..., j_n)$

Généralisation de la notion de segment

- Un segment de dimension n, noté s^n, est défini comme une liste de n intervalles $s^n = (i_1, ..., i_n)$ où i_j est un intervalle sur la dimension j.
- Un segment s^n de dimension n est dit vide quand $\exists i \in s^n \mid i = \emptyset$.
- L'intersection de segments de dimension n est définie comme la liste des intersections de leurs intervalles :

$$s_1^n \cap s_2^n = (i_1 \cap j_1, ..., i_n \cap j_n)$$

où $s_1^n = (i_1, ..., i_n)$ et $s_2^n = (j_1, ..., j_n)$

Généralisation de la notion de segment

- Un segment de dimension n, noté s^n, est défini comme une liste de n intervalles $s^n = (i_1, ..., i_n)$ où i_j est un intervalle sur la dimension j.
- Un segment s^n de dimension n est dit vide quand $\exists i \in s^n \mid i = \emptyset$.
- L'intersection de segments de dimension n est définie comme la liste des intersections de leurs intervalles :

$$s_1^n \cap s_2^n = (i_1 \cap j_1, ..., i_n \cap j_n)$$

où $s_1^n = (i_1, ..., i_n)$ et $s_2^n = (j_1, ..., j_n)$

Segmentations

- Une ressource R de dimension n est dite *segmentable* quand elle peut être vue comme un segment S^n de dimension n.
- La segmentation d'une ressource R est l'ensemble des segments $Seg(R) = \{s_1^n, ... s_i^n\}$ tels que :

$$\begin{array}{lll} \forall i, j \in Seg(R) & i \cap j = \emptyset & \text{les segments sont disjoints} \\ \forall i \in Seg(R) & i \cap S^n = i & \text{tous les segments sont inclus dans R} \end{array}$$

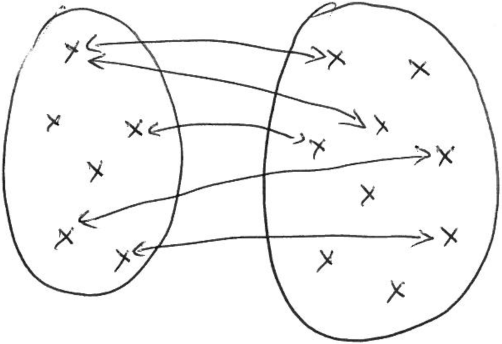
Segmentations

- Une ressource R de dimension n est dite *segmentable* quand elle peut être vue comme un segment S^n de dimension n.
- La segmentation d'une ressource R est l'ensemble des segments $Seg(R) = \{s_1^n, ...s_i^n\}$ tels que :

$$\begin{array}{rll} \forall i, j \in Seg(R) & i \cap j = \emptyset & \text{les segments sont disjoints} \\ \forall i \in Seg(R) & i \cap S^n = i & \text{tous les segments sont inclus dans R} \end{array}$$

Mapping (1)

Un *mapping* est une relation entre 2 segmentations.



Mapping (2)

- Pour un mapping $M \subseteq Seg(R_1) \times Seg(R_2)$ la fonction :

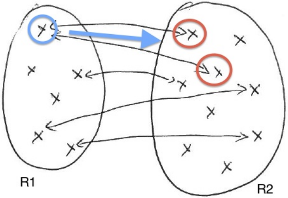
  $$M^+(i) = \{i' \in Seg(R_2) \mid (i, i') \in M\}$$

  donne l'ensemble des segments de R_2 associés au segment i de R_1

- et la fonction inverse :

  $$M^-(i') = \{i \in Seg(R_1) \mid (i, i') \in M\}$$

  donne l'ensemble des segments de R_1 associés au segment i' de R_2.



Mapping (2)

- Pour un mapping $M \subseteq Seg(R_1) \times Seg(R_2)$ la fonction :

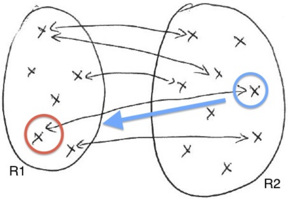
$$M^{+}(i) = \{i' \in Seg(R_2) \mid (i, i') \in M\}$$

donne l'ensemble des segments de R_2 associés au segment i de R_1

- et la fonction inverse :

$$M^{-}(i') = \{i \in Seg(R_1) \mid (i, i') \in M\}$$

donne l'ensemble des segments de R_1 associés au segment i' de R_2.



Mapping (3)

- Ces fonctions sont définies sur un ensemble de segments comme l'union des mappings de chaque segment :

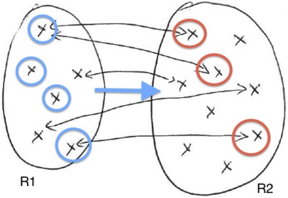
$$M^{+}(\{i_1, ...i_n\}) = M^{+}(i_1) \cup M^{+}(i_2) ... \cup M^{+}(i_n)$$

- Composition de mappings : soit $M_1 \subseteq Seg(R_1) \times Seg(R_2)$ et $M_2 \subseteq Seg(R_2) \times Seg(R_3)$

$$(M_1 \circ M_2)^{+}(i) = M_2^{+}(M_1^{+}(i))$$

- soit la relation :

$$M_1 \circ M_2 \subseteq Seg(R_1) \times Seg(R_3)$$



Mapping (3)

- Ces fonctions sont définies sur un ensemble de segments comme l'union des mappings de chaque segment :

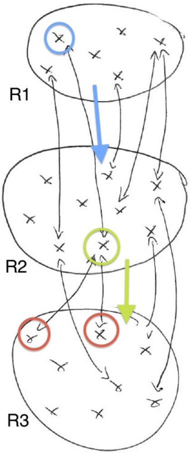
$$M^+(\{i_1, ...i_n\}) = M^+(i_1) \cup M^+(i_2) ... \cup M^+(i_n)$$

- Composition de mappings :
soit $M_1 \subseteq Seg(R_1) \times Seg(R_2)$
et $M_2 \subseteq Seg(R_2) \times Seg(R_3)$

$$(M_1 \circ M_2)^+(i) = M_2^+(M_1^+(i))$$

- soit la relation :

$$M_1 \circ M_2 \subseteq Seg(R_1) \times Seg(R_3)$$



Mapping (3)

- Ces fonctions sont définies sur un ensemble de segments comme l'union des mappings de chaque segment :

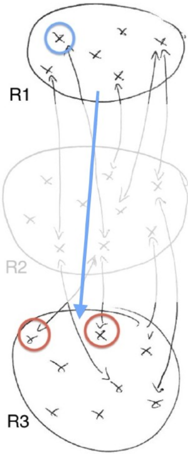
$$M^+(\{i_1, ...i_n\}) = M^+(i_1) \cup M^+(i_2) ... \cup M^+(i_n)$$

- Composition de mappings :
  soit $M_1 \subseteq Seg(R_1) \times Seg(R_2)$
  et $M_2 \subseteq Seg(R_2) \times Seg(R_3)$

$$(M_1 \circ M_2)^+(i) = M_2^+(M_1^+(i))$$

- soit la relation :

$$M_1 \circ M_2 \subseteq Seg(R_1) \times Seg(R_3)$$



Relations entre espaces graphiques et temporels

Segmentations et mappings pour chaque type de composant.

type	segmentations et mappings requis
texte	*graphique* ↔ **text** ↔ **temps relatif**
partition	*graphique*↔ *temps relatifenroulé* ↔ *temps relatif*
image	*graphique* ↔ **pixel** ↔ **temps relatif**
gr. vectoriel	**vectoriel**↔ **temps relatif**
signal	*graphique* ↔ **frame** ↔ **temps relatif**

Démo

Voir :

- Max/sync/sync.maxpat
- PureData/sync/sync.pd
- python/example.py
- lisp/example.lisp

INScoreViewer doit être actif.

Sommaire

Le problème...

Approche précédente :

Signal(s) (f0, rms...) → calcul de la représentation
Type de représentation → calcul de la représentation

- représentations statiques du signal
- non extensible dynamiquement

Et maintenant ?

- système plus général i.e. permettant de couvrir une grande variété de représentations
- système extensible dynamiquement
- et facile à utiliser...

Le problème...

Approche précédente :

Signal(s) (f0, rms...) → calcul de la représentation
Type de représentation → calcul de la représentation

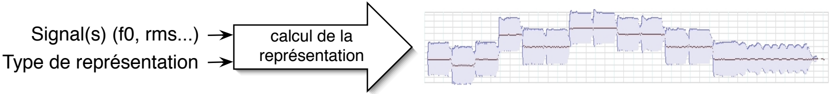
- représentations statiques du signal
- non extensible dynamiquement

Et maintenant ?

- système plus général i.e. permettant de couvrir une grande variété de représentations
- système extensible dynamiquement
- et facile à utiliser...

Le problème...

Approche précédente :



- représentations statiques du signal
- non extensible dynamiquement

Et maintenant ?

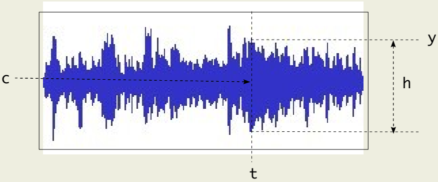
- système plus général i.e. permettant de couvrir une grande variété de représentations
- système extensible dynamiquement
- et facile à utiliser...

Des signaux graphiques...

Le *graphique d'un signal* vu comme un *signal graphique*:

Un signal composite constitué :

- d'un signal d'élévation (les valeurs en y) .
- d'un signal d'épaisseur .
- d'un signal de couleur .



Des signaux graphiques...

Considérons maintenant que nous disposons d'un signal S défini comme une fonction du temps :

$$f(t) : \mathbb{R} \rightarrow \mathbb{R}^3 = (y, h, c) \mid y, h, c \in \mathbb{R}$$

alors ce signal pourrait être dessiné directement.
(i.e. sans calcul supplémentaire)

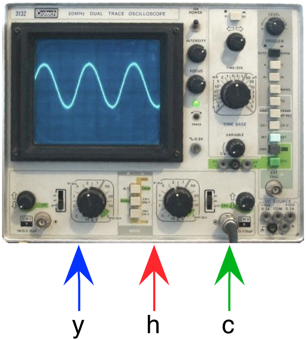
Pour simplifier, on suppose que l'espace de couleurs adressé par c n'a qu'une dimension.

Des signaux graphiques...

Considérons maintenant que nous disposons d'un signal S défini comme une fonction du temps :

$$f(t) : \mathbb{R} \to \mathbb{R}^3 = (y, h, c) \mid y, h, c \in \mathbb{R}$$

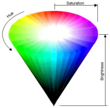
alors ce signal pourrait être dessiné directement.
(i.e. sans calcul supplémentaire)



Pour simplifier, on suppose que l'espace de couleurs adressé par c n'a qu'une dimension.

Types de signaux parallèle

- Type d'un signal de couleur :



(modèle HSBA [hue, saturation, brigthness, transparency])

$$c ::= \overrightarrow{(h, s, b, a)} \mid h, s, b, a \in \mathbb{R}$$

- Type d'un signal graphique :

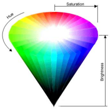
$$g ::= \overrightarrow{(y, th, h, s, b, a)} \mid y, th, h, s, b, a \in \mathbb{R}$$

- Type des signaux graphiques parallèles :

$$g^n ::= \overrightarrow{g} \mid g \in \mathbb{R}^6$$

Types de signaux parallèle

- Type d'un signal de couleur :



(modèle HSBA [hue, saturation, brigthness, transparency])

$$c ::= \overrightarrow{(h, s, b, a)} \mid h, s, b, a \in \mathbb{R}$$

- Type d'un signal graphique :

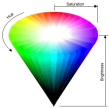
$$g ::= \overrightarrow{(y, th, h, s, b, a)} \mid y, th, h, s, b, a \in \mathbb{R}$$

- Type des signaux graphiques parallèles :

$$g^n ::= \overrightarrow{g} \mid g \in \mathbb{R}^6$$

Types de signaux parallèle

- Type d'un signal de couleur :



(modèle HSBA [hue, saturation, brigthness, transparency])

$$c ::= \overrightarrow{(h, s, b, a)} \mid h, s, b, a \in \mathbb{R}$$

- Type d'un signal graphique :

$$g ::= \overrightarrow{(y, th, h, s, b, a)} \mid y, th, h, s, b, a \in \mathbb{R}$$

- Type des signaux graphiques parallèles :

$$g^n ::= \overrightarrow{g} \mid g \in \mathbb{R}^6$$

Parallélisation de signaux

Soit $\mathbb{S}$, l'ensemble des signaux $s : \mathbb{N} \rightarrow \mathbb{R}$.
Nous définissons une opération *parallèle* '/' comme :

$$s_1 / s_2 / ... / s_n : \mathbb{S} \rightarrow \mathbb{S}^n \mid s_i \in \mathbb{S}$$

Fonction du temps d'un signal parallèle $s^n \in \mathbb{S}^n : \mathbb{N} \rightarrow \mathbb{R}^n$

$$f(t) = (f_0(t), f_1(t), ...f_n(t)) \mid f_i(t) : \mathbb{N} \rightarrow \mathbb{R}$$

Parallélisation de signaux

Soit $\mathbb{S}$, l'ensemble des signaux $s : \mathbb{N} \rightarrow \mathbb{R}$.
Nous définissons une opération *parallèle* '$/$' comme :

$$s_1/s_2/.../s_n : \mathbb{S} \rightarrow \mathbb{S}^n \mid s_i \in \mathbb{S}$$

Fonction du temps d'un signal parallèle $s^n \in \mathbb{S}^n : \mathbb{N} \rightarrow \mathbb{R}^n$

$$f(t) = (f_0(t), f_1(t), ...f_n(t)) \mid f_i(t) : \mathbb{N} \rightarrow \mathbb{R}$$

Exemples

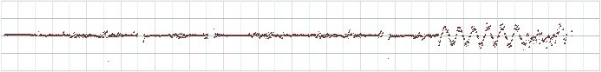
$$g = S_{f0} \ / \ k_t \ / \ k_c$$

S_{f0} : fréquence fondamentale

k_t : signal d'épaisseur constant

k_c : signal de couleur constant

Exemples



$$g = S_{f0} - S_{fr} \;/\; k_t \;/\; k_c$$

S_{f0} : fréquence fondamentale

S_{fr} : fréquence de référence

k_t : signal d'épaisseur constant

k_c : signal de couleur constant

Exemples

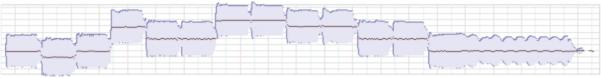
$$g = k_y \;/\; S_{rms} \;/\; k_c$$

S_{rms} : signal RMS

k_y : signal y constant

k_c : signal de couleur constant

Exemples



$$g = S_{f0} \ / \ S_{rms} \ / \ k_c$$

S_{rms} : signal RMS

S_{f0} : fréquence fondamentale

k_c : signal de couleur constant

Exemples

$$g0 = S_{f0} \ / \ S_{rms0} \ / \ k_c0$$

S_{f0} : fréquence fondamentale

S_{rms0} : valeurs RMS de f0

$$g1 = S_{f0} \ / \ S_{rms1} + S_{rms0} \ / \ k_c1$$

S_{rms1} : valeurs RMS de f1

$$g2 = S_{f0} / \ S_{rms2} + S_{rms1} + S_{rms0} \ / \ k_c2$$

S_{rms2} : valeurs RMS de f2

...

$$g = g2 \ / \ g1 \ / \ g0$$

Démo

Voir :

- Max/sinus/sinus.maxpat
- PureData/sinus/sinus.pd
- Max/siggraph/siggraph.maxpat
- PureData/siggraph/siggraph.pd

InterludeScoreViewer doit être actif.

INScore

Partitions augmentées interactives

http://inscore.sourceforge.net/

open source

GRAME CENTRE NATIONAL DE CREATION MUSICALE