**Contrôle de l’objet *Jgrain7****~*

João Svidzinski

**Description générale de l’objet Jgrain7~**

L’objet *Jgrain7~* est un ensemble de 7 piste comprenant un même traitement granulaire quasi-synchrone en parallèle[[1]](#footnote-1). Cet objet reprend le principe de la réinjection matricielle proposée par Alain Bonardi[[2]](#footnote-2), dont chaque sortie des 7 canaux peuvent être réinjecté vers n’importe quelle entrée du même objet. Le code faust du traitement granulaire de l’objet *Jgrain7~*  l est fondé sur le code original de Curtis Roads[[3]](#footnote-3) et appliqué dans le domaine de l’ambisonie par Pierre Guillot pour la bibliothèque Hoa[[4]](#footnote-4).

***Réinjection matricielle***

La méthode de réinjection matricielle dans l’objet *Jgrain7~* partage le même principe de l’objet *mTDelHarmo16*~ d’Alain Bonardi. Les blocs *inputBlock* et *outputBlock* (Figure 1) sont des ensembles de 7 gains linéaires permettant de varier les niveaux en entrée et en sortie. Le bloc *grainG* contient deux grands blocs (Figure 2) : *grainBlock* et *fdToMatrixBlock.* Le premier correspond aux 7 instances du traitement granulaire en parallèle, le deuxième correspond au module destiné a assingé les entrées et sorites pour la réinjection. Le code faust du *process* (Figure 3), ainsi que le code de la réinjection matricielle (Figure 4) se trouvent ci-dessous.

s

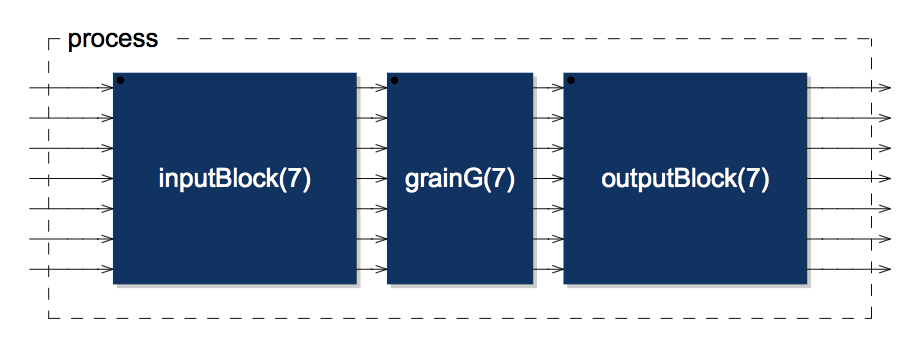


Figure 1 Process du traitement Granulateur Matriciel

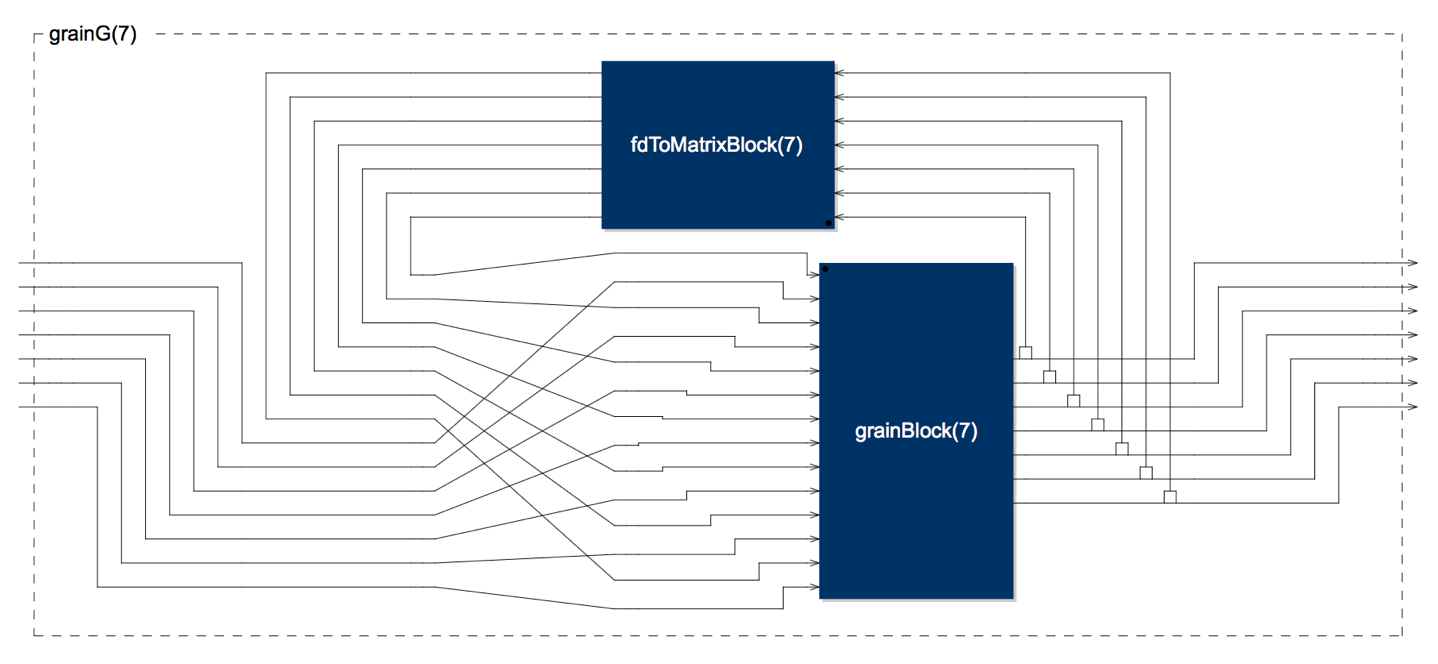


Figure 2 Shèma du principe matriciel du **Jgrain7.**

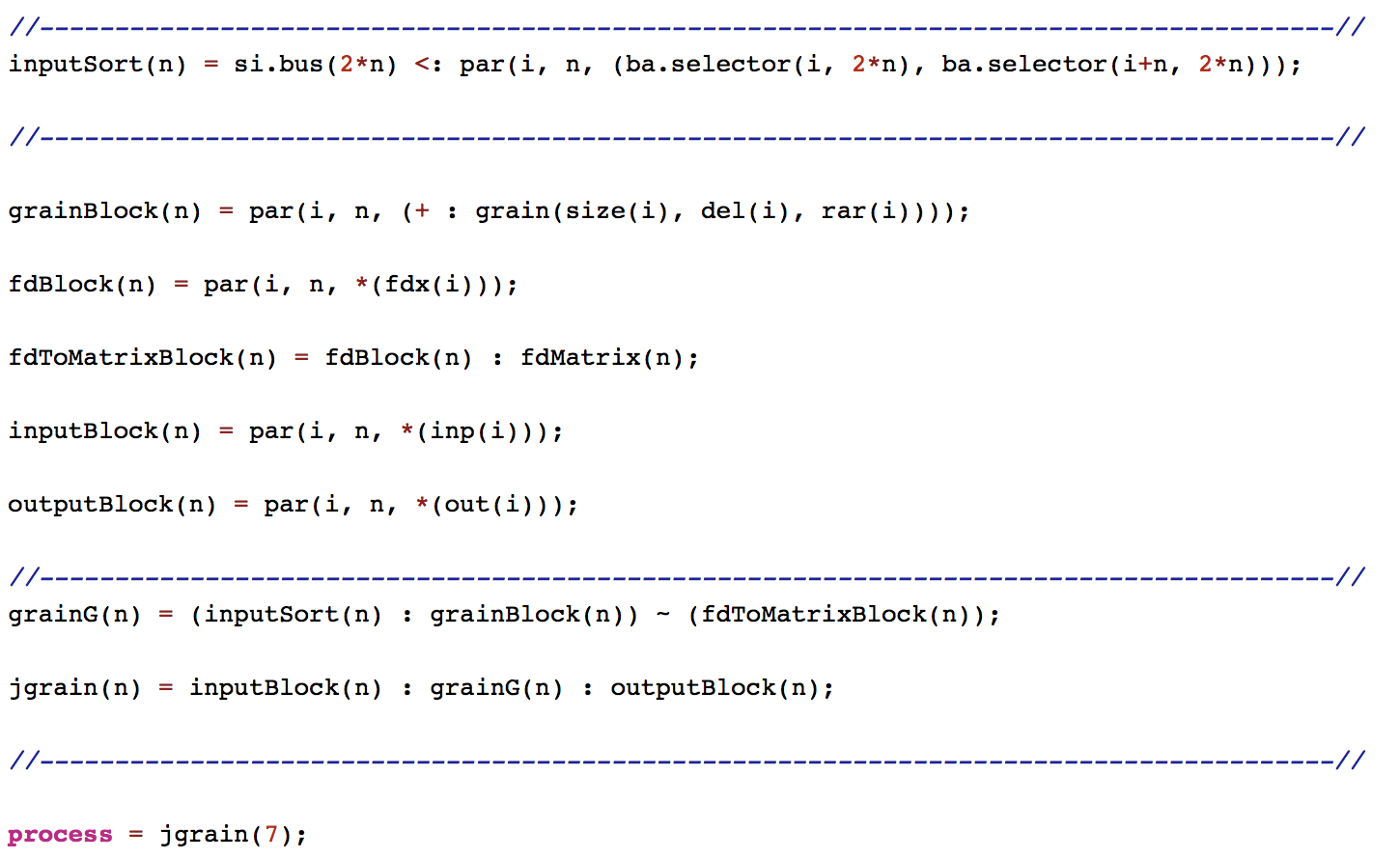


Figure 3 code Faust du process de l'objet Jgrain7.

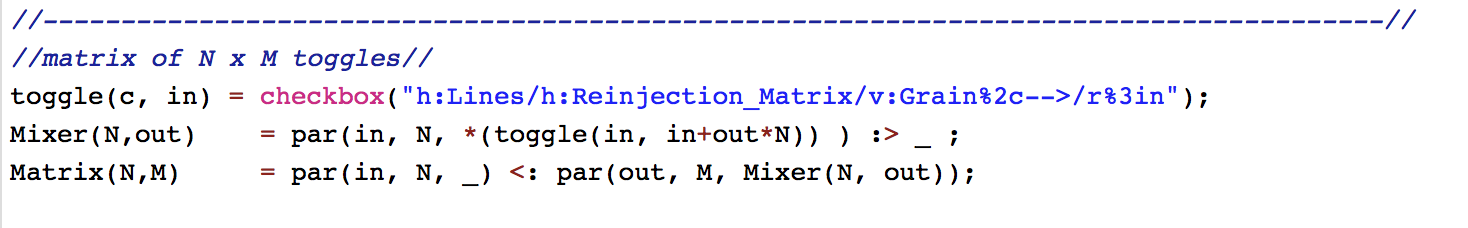


Figure 4 code Faust de la réinjection matricielle.

***Traitement granulaire***

Le traitement granulaire dans l’objet *Jgrain7~* est fondé sur la synthèse granulaire synthèse granulaire quasi-synchrone (QSGS) développé par Curtis Roads et mis en œuvre dans le domaine des harmoniques circulaires par Pierre Guillot. Il s’agit d’un traitement temps réel fondée sur des lignes à retard avec une enveloppe d’amplitude (type gaussienne) dont les grains sonores se succèdent à des intervalles irréguliers grâce à une déviation aléatoire qui détermine l’irrégularité (). Cela permet de générer des densités sonores diverses. Par exemple, si la densité de génération de grain est de l’ordre de 0.1 à 20 grains par seconde, des rythmes métriques seront produits, alors que si la densité́ change dans le temps, on entendra des *accelerandi* et *rallentandi*. Les densités plus enlevées génèrent des continuums.

Il y a donc quatre paramètres pour chacune des 7 instances du traitement granulaire :

* la taille moyenne des grains (en millisecondes, entre 0 et 1000) ;
* le temps de délai (en millisecondes, entre 0 et 5000) ;
* le pourcentage de réinjection des grains (entre 0 et 1) ;
* le pourcentage de raréfaction des grains (entre 0 et 1) ;

Il est toujours possible d’ajouter des taux de réinjection -pour augmenter la densité́ du rendu (voir réinjection matricielle ci-dessus). Cette matrice interne comporte 7 x 7 = 49 (chacun valant 1 ou 0). Le code faust et le schéma ci-dessous (Figure 5 et Figure 6) correspondent au traitement granulaire de l’objet *Jgrain7~.*

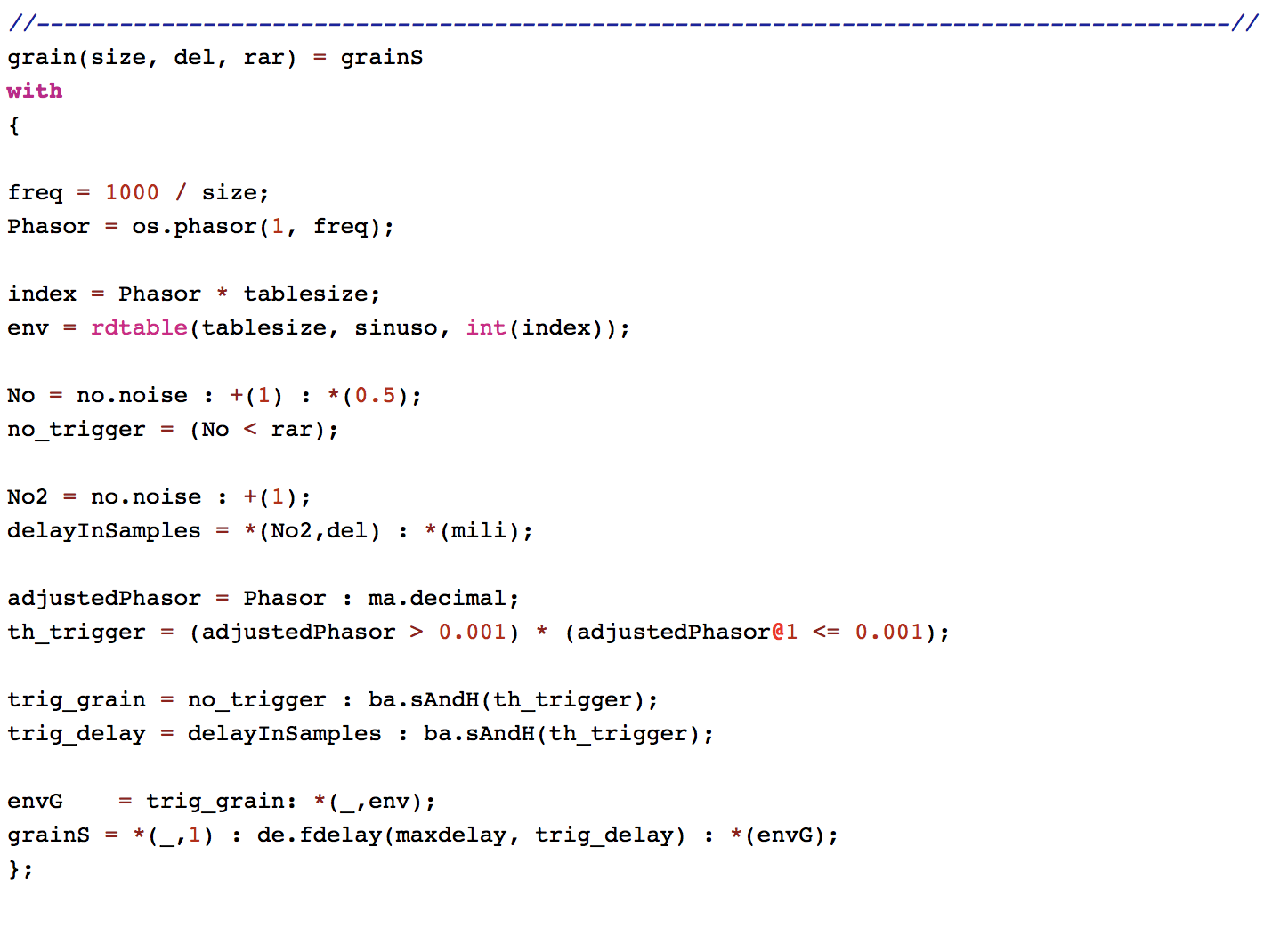


Figure 5 Code faust du traitement granulaire.



Figure 6 Schèmea d'une instance du traitement granulaire.

1. [↑](#footnote-ref-1)
2. L’objet faust [↑](#footnote-ref-2)
3. [↑](#footnote-ref-3)
4. Hoa library. [↑](#footnote-ref-4)