mix-calcul

Sylvain Huet (2006)

1 Introduction au mix-calcul

Le mix-calcul est un calcul permettant de résoudre un ensemble d'observables sous contraintes éventuellement contradictoires en effectuant des sélections aléatoires dans le respect de probabilités définies à l’avance.

Appliqué à la musique, il permet de générer une intelligence de mixage multi-pistes en choisissant pour chaque piste une brique musicale à jouer, et en tenant compte de contraintes fixées par le compositeur et de souhaits formulés par l’auditeur.

Le mix-calcul trouve en partie son inspiration dans les notions préliminaires de mécanique quantique.

2 Formalisation quantique du mix-calcul

2.1 Définitions

L’espace est composé d’observables « ***S*** ».
Chaque observable est associée à un vecteur d’états. Les états sont notés « ***E*** ».

A tout moment, une observable ***S*** est soit suspendue, soit dans un état ***E***. Dans ce dernier cas, on dit que l’état ***E*** est actif. On le note ***E(S).***

Les observables interagissent via des relations non symétriques « γ » et « τ ».

$S'\gamma S$: signifie que l’état de ***S*** dépend de l’état de ***S’***. Les cycles de la relation γ ne sont pas autorisés : on ne peut avoir $S_1\gamma S_2\gamma... S_n\gamma S_1$

$S'\tau S$: signifie que l’état de ***S*** dépend de l’état « précédent » de ***S’***. On note ***E’(S)*** l’état précédent d’une observable ***S***. La relation τ peut être réflexive.

Les relations γ ou τ et les observables peuvent être liées à des états par une relation α :

$E\ \alpha\ S$: si ***E*** est inactif, alors ***S*** est suspendu

$E\ \alpha\ \gamma$: si ***E*** est inactif, alors γ est suspendue

Une relation suspendue perd toute influence.

Lorsque deux observables ***S*** et ***S’*** sont en relation γ ou τ, on définit une matrice de probabilité des états de ***S’*** vers les états de ***S***. On écrit alors : $a\ \gamma_p\ b$, pour indiquer qu’un état ***a*** de ***S’*** contribue avec une probabilité ***p*** à l’état ***b*** de ***S***. On notera également $p_{S'\gamma S}(a,b)$ cette contribution, et même ***p(a,b)*** lorsqu’il n’y a pas d’ambiguïté possible. Cette contribution est un réel positif (éventuellement nul).

Une observable suspendue peut continuer d’influer via une relation γ ou τ : la matrice de probabilité est étendue à l’état « suspendu » de l’observable source.

On note qu’une observable n’ayant qu’un seul état et sans relation α ne peut activer que celui-ci. On parle d’ « observable absolument contrainte », car son état est toujours connu.

On définit une contrainte comme étant la manière de forcer une observable à se trouver dans un certain état.

Comme des contraintes peuvent être contradictoires, on doit les ordonner en leur affectant une importance. De ce fait, on attribue aux relations γ et τ, ainsi qu'aux contraintes, un niveau d'importance.

Ce niveau d'importance peut éventuellement être infini pour les relations γ. Il doit être fini pour les relations τ et pour les contraintes ; cela se justifie par le fait que :

- on doit pouvoir maintenir l'espace dans un état donné, ce qui peut nécessiter de geler des relations τ .
- les contraintes appliquées doivent être considérées comme des souhaits.

2.2 *Résolution (ou réduction)*

2.2.1 Résolution et espace librement calculable

La réduction d'une observable S consiste à déterminer la probabilité de chacun de ses états, puis à faire une sélection aléatoire en tenant compte de ces probabilités. Cette sélection détermine l'état du observable S.

La probabilité, avant normalisation, d'un état b de S vaut :

$$p(b)= \Pi_{S'\gamma S}\, p(E(S),b).\ \Pi_{S'\tau S}\, p(E'(S),b)$$

Cette probabilité se calcule sur les relations γ ou τ non suspendues.

La probabilité normalisée d'un état b de S vaut :

$$P(b)=p(b)/\ \Sigma_{a\in S}\, p(a)$$

Cette probabilité n'existe que si la somme située en diviseur n'est pas nulle, c'est-à-dire s'il existe au moins un état avec une probabilité non nulle avant normalisation.

La résolution de l'espace consiste à déterminer l'état de tous les observables de telle manière que les relations éventuelles soient satisfaites.

Un espace est « librement calculable » s'il existe une résolution en ne tenant compte que des relations d'importance infinie.

On ne s'intéresse dans la suite qu'aux espaces « librement calculables ».

2.2.2 Résolution sous contrainte externe

La résolution sous contrainte externe consiste à imposer l'état de certaines observables.

La contrainte externe consiste à poser $E(S)=b$.

Les contraintes sont associées à un critère d'importance, qui définit un ordre total (cette notion d'importance dépend de l'applicatif qui utilise le mix-calcul).

La résolution sous contrainte consiste à déterminer l'état de toutes les observables, de telle sorte que toutes les relations et toutes les contraintes soient respectées, y compris les relations d'importance finie.

2.2.3 Résolution faible optimale

La résolution faible consiste à identifier une solution en supprimant éventuellement quelques contraintes ou relations. Une résolution faible est caractérisée par cet ensemble des contraintes et relations supprimées.

La résolution faible optimale consiste à déterminer la « meilleure » résolution faible en appliquant cette règle :

- soit A et B deux résolutions faibles (A et B sont les ensembles des contraintes et relations supprimées)
- soit c la contrainte d'importance la plus forte présente dans A et absente de B, ou présente dans B et absente de A.
- si c est présent dans B et absent de A, on considère que A est meilleur que B
- si c est présent dans A et absent de B, on considère que B est meilleur que A

Il est évident qu'un espace « librement calculable » peut toujours être résolu faiblement : dans le pire des cas on peut le résoudre en supprimant toutes les contraintes et toutes les relations d'importance finie.

2.3 Observables numériques et relations arithmétiques

On définit des observables numériques, qui sont des observables particulières dont les états sont les nombres réels. On écrira $\boldsymbol{S_a}$. Ce sont donc des observables dont les états sont en nombre infini et en congruence avec l'espace des réels.

On définit des relations arithmétiques. Au lieu de définir des relations gamma et tau entre des observables $\boldsymbol{S_1, S_2, ..., S_n}$ et une observable $\boldsymbol{S}$, on représente ces relations sous la forme d'une expression arithmétique entre les observables $\boldsymbol{S_1, S_2, ..., S_n}$ et l'observable $\boldsymbol{S}$.
Cette expression s'appuie sur les états présents ***E(…)*** ou passés ***E'(…)*** des observables $\boldsymbol{S_1, S_2, ..., S_n}$ et fournit l'état actif de $\boldsymbol{S}$.
Si une observable est numérique, son état est un nombre réel (par convention : 0 si l'observable est suspendue).
Si une observable n'est pas numérique, ses états sont des éléments d'un ensemble.
Par exemple :

$$\boldsymbol{S := \text{if } (E(S_1)+E'(S_2))=0 \text{ then } a \text{ else } b}$$

$$\boldsymbol{S := 1+\text{if } E(S_1)=a_1 \text{ then } 0 \text{ else } 1}$$

(où $\boldsymbol{a}$ et $\boldsymbol{b}$ sont des états de $\boldsymbol{S}$, $\boldsymbol{a_1}$ un état de $\boldsymbol{S_1}$, et $\boldsymbol{E'(S_2)}$ est l'état précédent de $\boldsymbol{S_2}$)

Les primitives sont :

- +,*,-,/,%, &, | , &&, ||, ^, !, ~
- if ... then ... else ...
- rand (retourne un réel entre 0 et 1)
- sin, cos, tan, ...

On dit alors que l'observable $\boldsymbol{S}$ est en résolution arithmétique. Dans le cas contraire, l'observable $\boldsymbol{S}$ est en résolution quantique.

On montre qu'il y a inclusion de la résolution arithmétique dans la résolution quantique (c'est-à-dire qu'une relation arithmétique peut s'exprimer sous forme de relations gamma et tau), si bien que les considérations précédentes sur la résolution des espaces restent valides.

Pour garder la complexité de la résolution dans des limites raisonnables, on fixera les limitations suivantes :

- une observable numérique est toujours en résolution arithmétique, car sinon les matrices des relations quantiques auraient des tailles infinies
- on ne peut appliquer une contrainte sur une observable qui dépend d'une résolution arithmétique, car cela reviendrait à calculer l'inverse de toute fonction arithmétique :
    - l'observable n'est pas en résolution arithmétique
    - l'observable ne dépend pas, directement ou indirectement, d'une observable en résolution arithmétique

Cette limitation pourra être transgressée dans certains cas à la complexité réduite et qui seraient fastidieux à implémenter en résolution quantique. Par exemple : ***S=if E(S')!=E'(S') then a else b***

2.4 Equivalences

Dans ce chapitre, on démontre que les relations τ, les contraintes et les relations α peuvent s'écrire sous forme de relations γ.

2.4.1 Relations τ

On remarque qu'une relation τ est équivalente à une relation γ avec contrainte ; on remplace $\boldsymbol{S\ \tau\ S'}$ par :

- une observable $\boldsymbol{S_{prev}}$, congruente à $\boldsymbol{S}$ (c'est-à-dire avec les « mêmes » états)
- une relation $\boldsymbol{S_{prev}\ \gamma\ S'}$, de même matrice que la relation τ
- la contrainte $\boldsymbol{E(S_{prev})=E'(S)}$.

2.4.2 Contraintes

On remarque qu'une contrainte peut être vue de manière plus générale comme une relation γ entre une observable absolument contrainte et l'observable à contraindre. La matrice de cette relation est alors réduite à un vecteur dont tous les coefficients sauf un sont à zéro.

2.4.3 Probabilité infiniment faible ε

On introduira la probabilité ε considérée comme infiniment faible. Une manière de se présenter cette probabilité est de considérer que lors de la résolution d'une observable le dé de la Nature ne comporte jamais assez de faces pour en attribuer une à cette probabilité.

L'état qui présente une telle probabilité ne peut être choisi que s'il est le seul à avoir une probabilité non nulle.

2.4.4 Relations α et observables suspendues

On remarque qu'une relation α s'exprime sous forme de relations γ et d'un état supplémentaire que l'on appellera $\boldsymbol{\pi}$ (comme pause).

En considérant que la relation α est liée au dernier état de l'observable source S, et que l'état π est le dernier état de l'observable destination S', la relation γ s'écrit :

$$\gamma = \left[\begin{array}{c|c} 0 & 1 \\ \hline 1 & \epsilon \end{array}\right]$$

Les autres relations vers S' doivent être étendues pour définir une contribution de probabilité pour ce nouvel état π. Il faut que cette contribution soit ε.

$$\gamma' = \left[\begin{array}{c|c} \gamma & \epsilon \end{array}\right]$$

De cette manière on peut combiner les relations α et les relations γ ou τ et obtenir le résultat attendu.

2.4.5 Relations α et relations suspendues

On remarque qu'une relation α sur une autre relation s'exprime sous forme de relations γ et d'une observable supplémentaire.

On considère par exemple une relation τ entre deux observables A et B, et une relation α entre une observable C et la relation τ.

On crée alors une observable A', congruente à l'observable A, avec un état supplémentaire π'. On définit la relation Id' qui est une relation γ entre A et A'.

$$Id' = \left[\begin{array}{ccc|c} 1 & & 0 & \\ & \ddots & & 1 \\ 0 & & 1 & \end{array}\right]$$

La relation α est remplacée par une relation γ entre C et A', dont la matrice est :

$$\gamma = \left[\begin{array}{c|c} 0 & 1 \\ \hline 1 & \epsilon \end{array}\right]$$

La relation τ est alors remplacée par une relation τ' entre A' et B :

$$\tau' = \left[\begin{array}{c} \tau \\ \hline 1 \end{array}\right]$$

2.5 Exemples de calculs en résolution quantique

Par convention, quand on ne précise pas les contributions de probabilité, elles sont considérées valoir 1.

2.5.1 Opérateur « Non »

Définitions :

SγS'
S={a,b}
S'={a',b'}

$$\gamma = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$$

Alors, en considérant que ***a≡a'≡true***, et ***b≡b'≡false*** :

E(S')=!E(S)

E(S)=a ⇔ E(S')=b'
E(S)=b ⇔ E(S')=a'

2.5.2 Opérateur « Nand »

Définitions :

$S_1 \gamma S'$
$S_2 \gamma S'$
$S' \gamma S$

$S_1=\{a_1,b_1\}$
$S_2=\{a_2,b_2\}$
$S'=\{a_1\ a_2,\ a_1\ b_2,\ b_1\ a_2,\ b_1\ b_2\}$
$S=\{a,b\}$
$p(a_1,\ b_1a_2)=p(a_1,\ b_1b_2)=0$
$p(b_1,\ a_1a_2)=p(b_1,\ a_1b_2)=0$
$p(a_2,a_1b_2)=p(a_2,b_1b_2)=0$
$p(b_2,a_1a_2)=p(b_2,b_1a_2)=0$
$p(a_1a_2,a)=0$
$p(a_1b_2,b)=0$
$p(b_1a_2,b)=0$

$p(b_1 b_2, b)=0$

Alors, en considérant que $a_1 \equiv a_2 \equiv a' \equiv true$, et $b_1 \equiv b_2 \equiv b' \equiv false$:

$E(S)= !E(S_1) \wedge E(S_2)$

2.5.3 Opérateur « Nand », bis !

On peut simplifier l'opérateur Nand en utilisant la probabilité ε.

Définitions :

$S_1 \gamma S$

$S_2 \gamma S$

$S_1=\{a_1,b_1\}$

$S_2=\{a_2,b_2\}$

$S=\{a,b\}$

$$\gamma = \begin{bmatrix} 0 & 1 \\ 1 & \epsilon \end{bmatrix}$$

Alors, en considérant que $a_1 \equiv a_2 \equiv true$, et $b_1 \equiv b_2 \equiv false$:

$E(S)= !E(S_1) \wedge E(S_2)$

La matrice AND :

$$\gamma = \begin{bmatrix} 1 & 0 \\ \epsilon & 1 \end{bmatrix}$$

La matrice OR :

$$\gamma = \begin{bmatrix} 1 & \epsilon \\ 0 & 1 \end{bmatrix}$$

2.5.4 Oscillateur

Définitions :

$S \tau S$

$S=\{a,b\}$

$p(a,a)=p(b,b)=0$

Alors, à chaque nouvelle résolution, l'observable S change d'état.

2.5.5 Rom

Définitions :

$S=\{a\}$

L'observable ***S*** est toujours dans l'état ***a***.

2.5.6 Disable

Définitions :

S={a}
S'={enable,disable}
SγS'
p(a,enable)=1
p(a,disable)=0

Alors, l'état ***enable*** est toujours actif, l'état ***disable*** n'est jamais actif.

2.5.7 Chaîne de Markov

Définitions :

SτS
S={a,b,c}
p(a,c)=0
p(b,a)=0
p(c,a)=0

Supposons que l'état initial de ***S*** est ***a***.
Alors l'observable reste un certain temps dans l'état ***a***, puis passe dans l'état ***b***, puis évolue sans fin entre l'état ***b*** et l'état ***c***, sans jamais revenir dans l'état ***a***.

2.5.8 Turing-complet

Armés d'un oscillateur et de portes logiques NAND, nous sommes en mesure de construire un microprocesseur, ce qui prouve la complétude du mix-calcul.

2.6 Algorithme de la résolution faible optimale

2.6.1 Version simplifiée

On commence par trier les contraintes et relations d'importance finie dans l'ordre d'importance décroissant. Soit **L** la liste correspondante.
On désactive tous les éléments de **L**.

On itère sur la liste L de la manière suivante :

- pour un élément C de L :
  - on active C
  - on tente une résolution faible
  - en cas de succès, on laisse C actif, et on poursuit l'itération
  - en cas d'échec, on désactive C, et on poursuit l'itération

A la fin de l'itération, on a trouvé la combinaison optimale des contraintes permettant une résolution. On refait une résolution faible avec cette combinaison et on trouve la résolution faible optimale.

L'algorithme pour une résolution faible est le suivant.

- préparation :
  - les observables sont initialisées dans l'état « non résolu »
  - on précalcule la table des dépendances : une observable X dépend d'une observable Y s'il existe une chaîne de relations (alpha, gamma ou tau) depuis Y vers X.
- puis récursivement, choisir une observable résoluble :
  - l'observable n'est pas encore résolue
  - les relations α de cette observable mènent à des états dont les observables sont résolues
  - si l'observable est en résolution quantique :
    - toutes les relations entrantes sont dans un état connu (suspendu ou non)
    - les relations entrantes gamma non suspendues ont une source résolue
  - si l'observable est en résolution arithmétique :
    - toutes les observables utilisées dans la relation arithmétique sont résolues
  - heuristique : on s'intéresse en priorité aux observables en résolution arithmétique, puis à l'observable en résolution quantique ayant le moins d'états possibles
- on détermine si l'observable est suspendue :
  - si oui, on regarde si une contrainte était associée à l'observable :
    - si oui : la résolution de cette observable échoue. Cette observable est temporairement considérée comme l'échec terminal.
    - si non : on résout l'observable en la mettant dans l'état « suspendu » et on poursuit récursivement
- sinon en résolution arithmétique :
  - on évalue l'expression arithmétique, qui donne le nouvel état
  - on résout récursivement
- sinon en résolution quantique :
  - on calcule les probabilités des états de l'observable
  - si aucun état n'est possible, ou si l'observable est contrainte sur un état impossible (ie de probabilité nulle), la résolution de cette observable échoue. Cette observable est temporairement considérée comme l'échec terminal.
  - on effectue un tirage en respectant les probabilités, puis on essaie les états en commençant par celui qui a obtenu le meilleur score
    - on choisit un état, et on résout récursivement
    - si la résolution récursive échoue, et que l'observable en échec terminal dépend de celle-ci, on essaye l'état suivant
    - si aucun état n'est possible, la résolution échoue

Lors de l'itération sur la liste L, afin de déterminer la liste des contraintes à appliquer, il est inutile de compliquer le calcul des probabilités, seul compte le fait qu'un état soit possible ou non. Ceci évite le tirage puis le tri des états lors d'une résolution quantique.

2.7 Première optimisation

Le principe de l'optimisation consiste à ne pas tout recalculer lors de l'itération sur la liste L.

On remarquera en effet que :

- ***une contrainte réduit l'ensemble des solutions, elle n'en permet pas de nouvelle***.

Avant d'itérer sur la liste L, on effectue une première résolution, qui réussit nécessairement si l'espace est librement calculable.

A chaque itération, on regarde si la contrainte est compatible avec la dernière solution trouvée:

- pour une contrainte sur un état, on vérifie si l'état est déjà choisi. Si une contrainte est déjà en place sur un état différent on supprime la nouvelle contrainte qui est moins importante (étant donné l'ordre d'itération)
- pour une relation, on vérifie qu'elle n'est pas incompatible avec l'état de l'observable sur laquelle elle s'applique

En cas d'incompatibilité, on effectue le traitement suivant :

- sauvegarde de la solution courante
- activation de la contrainte
- recherche d'une nouvelle solution, « from scratch »
- si échec, restauration de la solution précédente, et désactivation de la contrainte

La recherche d'une nouvelle solution peut se faire également de manière incrémentale. Pour cela il faut :

- conserver la liste des états encore possibles d'une observable (la liste issue du tirage, celle des états ayant une probabilité non nulle et sur laquelle on est « en train » d'itérer)
- déterminer statiquement un ordre de résolution des observables
- cette liste des états possibles doit être mise à jour lors de l'ajout d'une contrainte pourtant compatible avec la solution courante :
  - une contrainte sur un état limite la liste des états possibles à cet état contraint
  - une relation filtre la liste des états possibles
- lorsqu'une contrainte nécessite de recalculer une observable S, on reprend l'algorithme en tenant pour acquis le début de la recherche des observables précédentes, et en recalculant complètement l'observable S et ses suivantes.

Question : peut-on ne pas recalculer complètement toutes les observables suivantes de S ?
On a vu le cas de trois observables A, B, C, dans cet ordre :

- A et B ont une relation vers C.
- on prend le premier état de A
- on doit aller jusqu'au dernier état de B pour trouver un état de C
- puis on fixe une contrainte sur le deuxième état de A (qui est en fait compatible via C avec le premier état de B)
- bien que B ne dépende pas de A, il faut le recalculer complètement

Pour toutes les observables X dépendant de A, on réinitialise X ainsi que toutes les observables W dont X dépend et qui suivent A, et récursivement sur celles qui dépendent de W.
Cette matrice peut être précalculée. Son usage est :

- chaque fois qu'on essaie un nouvel état pour une observable A (par ajout d'une contrainte, ou dans le cadre normal de l'algorithme de résolution faible), la matrice indique les observables suivantes qui doivent être complètement recalculées
- dans l'algorithme de résolution faible, on a alors deux cas :
  - soit l'observable solvable doit être recalculée, et dans ce cas c'est l'algorithme « normal », et on utilise la matrice pour « vider » les observables suivantes concernées
  - soit l'observable solvable n'a pas à être recalculée, et dans ce cas, on tente la résolution sur la solution en cours (sans « vider » les observables suivantes concernées), et on itère si ça ne marche pas sur la suite des solutions (fin de l'algorithme normal)

Autre approche :
Pour chaque observable, on conserve deux listes : la liste des états restant à essayer, et ceux déjà essayés mais éliminés.
Lorsqu'on modifie une observable A, on regarde les observables X suivantes :
- si X dépend de A, on la vide ; il faudra la recalculer complètement
- si X ne dépend pas de A, on remet dans la liste des états à essayer ceux déjà éliminés

2.8 Complexité du mix-calcul

2.8.1 Cas général

La résolution du mix-calcul est un problème NP-complet.

résolution 3Sat dans mix-calcul ?
=> on construit la formule avec des portes OR, AND et NOT (construction polynomiale).
=> les observables racines sont les littéraux
=> on contraint la sortie à 1 (vrai)
le mix-calcul trouve la solution...

On peut « câbler » la contrainte de sortie, par exemple avec une matrice « dégénérée » [0 1], ce qui nous conduit à un espace de calcul sans contrainte externe. Une matrice dégénérée étant une matrice dont une des colonnes ne contient que des zéros (ce qui signifie qu'un état n'est pas atteignable si la matrice est active).

Sans matrice dégénérée, on peut appliquer sur la sortie une relation supplémentaire (matrice 2x2 identité) issue d'une observable qui vaut toujours 1, par exemple NOT (x XOR x).

D'où la question :
- quel est le sous-ensemble des graphes mix-calcul qui sont simplement NP ou P ?

On considère les valeurs suivantes :
- S : nombre d'observables
- E : nombre d'états moyen par observable
- R : nombre de relations moyen entrante par observable

Calcul des probabilités d'un système : R.E
Calcul, tirage et tri : R.E+E.lnE

2.8.2 Cas le plus simple

On considère un graphe mix-calcul dans lequel aucune contribution de probabilité n'est nulle et dans lequel aucune contrainte n'est appliquée.
Dans ce cas, la résolution réussit toujours du premier coup !
Le temps de calcul est : S.R.E