REPRÉSENTATIONS VARIATIONNELLES INVERSIBLES : UNE NOUVELLE APPROCHE POUR LA SYNTHÈSE SONORE

Axel Chemla–Romeu-Santos
LIM – IRCAM
chemla@ircam.fr

Philippe Esling
IRCAM
esling@ircam.fr

Stavros Ntalampiras
LIM
stavros.ntalampiras@unimi.it

RÉSUMÉ

Dans cet article, nous proposons une nouvelle méthode de synthèse sonore basée sur des méthodes d'auto-encodage variationnel permettant simultanément d'*inférer* une représentation inversible d'un ensemble de données, que nous appelons ici *espace génératif*, et de générer à partir de ces propriétés structurelles extraites. Ces méthodes récentes, basées sur l'extraction d'espaces à faible dimensionnalité grâce à l'utilisation jointe de réseaux neuronaux et d'inférence bayésienne, permettent non seulement une grande flexibilité architecturale mais aussi l'extraction d'espaces génératifs haut-niveau, pouvant être explorés directement ou indirectement par l'interacteur. Néanmoins, le choix simultané de l'architecture et des données apprises conditionne de manière déterminante les propriétés émergentes des espaces génératifs extraits, dont l'organisation reste encore mal définie. Pour ce faire, nous proposons une approche expérimentale de ces systèmes par le développement d'une bibliothèque, vschaos, visant à développer une approche bijective entre l'évaluation de ces modèles, et leur exploitation dans des environnements créatifs.

1. INTRODUCTION

Les environnements de synthèse sonore modernes, que l'on peut représenter comme une boucle d'interaction entre un ensemble d'acteurs (humains ou non-humains) et de systèmes génératifs, basent notamment leur potentialité créative sur la richesse de l'interaction possible entre ces deux types d'acteurs, s'interfaçant généralement par un ensemble fini de *paramètres*. En effet, l'apparition de nouveaux dispositifs au cours du XX^{e} siècle, d'abord par l'utilisation de dispositifs analogiques puis numériques et informatiques, permirent d'abstraire l'aspect *contrôle* de ces procédés de leur aspect purement *génératif*. Ainsi, le découplage de ces deux aspects provoquèrent l'apparition de nouveaux discours musicaux, pouvant se baser soit sur la correspondance des phénomènes produits avec des phénomènes réels (avec donc une certaine intention *imitative*), mais aussi par l'extrapolation ou le détournement de ces principes générateurs induits (une intention que nous appellerons *créative*). Se crée ainsi de manière sous-jacente une complémentarité entre deux processus : d'une part, un processus d'*analyse* consistant à subdiviser un phénomène

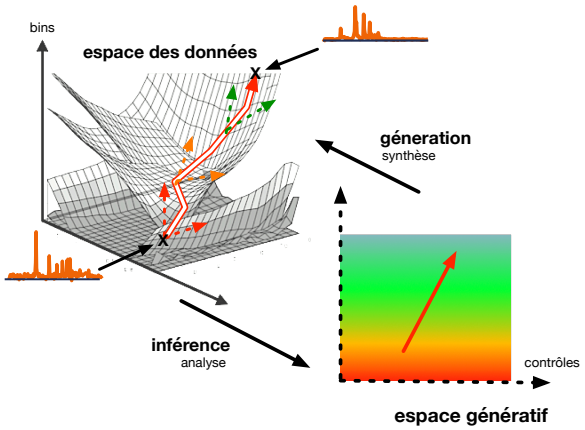

Figure 1. Dans cette méthode, nous proposons d'utiliser des méthodes d'extraction de sous-variétés non-linéaires inversibles dans l'espace de données, ici des informations spectrales, pour *inférer* un espace *génératif* pouvant être utilisé pour contrôler la génération sonore.

en parties interprétables et contrôlables, et d'autre part un processus de *synthèse* recomposant ces éléments pour non seulement pouvoir vérifier la validité de la décomposition effectuée par l'analyse, mais aussi recomposer ces éléments de manière inédite afin de permettre la création de nouveaux artefacts.

Concernant le traitement du signal sonore, ce procédé d'analyse est majoritairement représenté par l'extraction de paramètres musicaux (plus communément appelé *Music Information Retrieval* ou MIR [35]), consistant à inférer un ensemble d'informations perceptives (fréquence fondamentales, descripteurs acoustiques, etc.), dynamiques (détection d'attaques, estimation de tempo, etc.) ou encore sémantiques/syntaxiques (extraction de structure, reconnaissance de genre, etc.) à partir d'un signal audio donné (voir section 2.2). Le procédé de synthèse est quant à lui majoritairement basé sur le développement de nouveaux procédés génératifs dans les domaines physique (lutherie), analogique (synthétiseurs, modulaires) ou numérique (plug-ins, patches), livrés à un utilisateur afin d'être intégrés dans un environnement de création musicale (temps réel, composition).

Joignant ces deux processus, les méthodes d'*analyse-synthèse* se basent sur l'extraction d'une *représentation* alternative du signal sonore, que l'on peut inverser pour re-

couvrer le signal correspondant afin de faciliter l'extraction de certaines propriétés musicales. Les implications créatives de la réversibilité de cette représentation intermédiaire sont très intéressantes, permettant non seulement la génération de nouveaux contenus soit par la modification de sons existants (projetés dans la représentation par analyse), soit par interaction directe avec celle-ci. Ces méthodes permettent donc d'unifier ces deux paradigmes, permettant simultanément d'extraire des propriétés indirectes du signal observé, tout en étant capable de les recomposer pour recouvrer le signal final. Néanmoins, la réversibilité absolue de la plupart de ces méthodes implique en général un grand nombre de paramètres, pouvant rendre laborieuse leur utilisation directe pour la création de nouveaux matériaux, et ne permettent pas l'extraction de représentations spécifiques à un ensemble de sons choisis.

De manière parallèle, le développement d'approches statistiques, c'est-à-dire basées sur la description d'un ensemble de données par un ensemble de propriétés structurelles (répartition, organisation, corrélation, etc.), permirent une approche différente de ce procédé d'analyse. Ces approches, déjà couramment utilisée en analyse afin d'améliorer la robustesse de certains estimateurs de paramètres musicaux, peuvent aussi être utilisés en synthèse pour caractériser certains modèles génératifs, modélisant le signal sonore comme un processus stochastique [13]. L'utilisation récente de méthodes d'approximation haute-capacité d'inspiration connexioniste pour la modélisation de signaux permit le développement d'un ensemble d'approches capables à la fois d'affiner leur adaptabilité et leur expressivité, en plus de pouvoir incorporer une quantité de plus en plus importante de données (voir section 2.1). Ce gain quantitatif et qualitatif en termes de performances suscita un regain d'intérêt pour l'utilisation de ces systèmes comme méthodes à proprement parler *génératives*, permettant de générer de nouveaux exemples de manière non-supervisée. De plus, certaines de ces méthodes se basent sur l'extraction de représentations à basse dimensionnalité dans le cadre de l'*hypothèse de sous-variété*, considérant la structure d'un ensemble de données jugé consistant comme une sous-variété de l'espace des données possibles [5]. Cependant, malgré les résultats impressionnants obtenus par ces méthodes de génération, l'évaluation des propriétés réellement apprises est une procédure complexe sur plusieurs aspects. En effet, l'évaluation quantitative de ce type de méthodes s'effectue dans la discipline sur un ensemble de tâches précises, généralement incluses dans leur conception (critères *extrinsèques*), mettant ainsi de côté la caractérisation de certaines propriétés émergentes du système (critères *intrinsèques*) pourtant mises en jeu lors de leur utilisation dans des cadres créatifs [31]. De plus, la coïncidence pouvant apparaître entre critères d'évaluation et critères d'entraînement peut soulever des problèmes d'ordre épistémologiques [33], amplifiés par le coût de calcul généralement assez élevé de ce type de méthodes.

Dans cet article, nous choisissons l'*auto-encodage variationnel* comme nouvel environnement pour la modélisation non-supervisée d'un espace génératif de synthèse sonore (voir figure 1). En effet, partant de l'hypothèse de sous-variété, l'extraction de la sous-variété sous-jacente à un ensemble de données sonores jugé consistent pourrait nous permettre d'en extraire un ensemble de facteurs génératifs spécifique, pouvant ainsi être utilisé comme espace de contrôle. Cette méthode est basée simultanément sur deux processus, pouvant être rapprochés des méthodes d'analyse-synthèse : d'une part l'*inférence* d'une représentation à basse dimensionnalité à partir d'un ensemble de données, et d'autre part la *génération* de nouvelles données à partir des facteurs génératifs extraits.

Le choix de cette méthode est motivé par plusieurs raisons. Premièrement, le choix de méthodes non-supervisées est à notre avis fondamental pour l'aspect créatif de leur utilisation, ainsi basée sur l'exploration de leur propriétés émergentes par leur représentation interne. Deuxièmement, ces modèles possèdent une grande flexibilité en terme d'architecture, la conjugaison entre approches bayésienne et connexioniste permettant de développer des systèmes à petite ou grande échelle de manière non-supervisée (requérant des données seules) ou (semi-)supervisée (incluant des méta-données d'ordre symbolique), tout en garantissant leur robustesse ainsi que leur faible complexité. Néanmoins, nous sommes convaincus que l'évaluation de cette famille de modèles dans des environnements créatifs requiert le développement d'une approche *expérimentale*, visant à approfondir la caractérisation de leur propriétés intrinsèques par leur mise en place pratique dans des contextes de manipulation et de création [31].

Dans cet article, nous présentons donc cette nouvelle méthode de synthèse sonore, présentant ses fondements mathématiques, le cadre choisi pour leur exploitation dans le domaine de la synthèse sonore, ainsi que les différentes applications créatives ouvertes par ce nouveau système. Ensuite, nous présentons notre bibliothèque open-source vschaos, visant à ouvrir l'utilisation à une communauté experte et non-experte, de manière à susciter des situations d'expérimentations et donc d'ouvrir la voie à une véritable approche de *recherche & création* pour ces nouveaux environnements.

2. ÉTAT DE L'ART

2.1. Génération sonore et méthodes génératives

L'intérêt croissant envers les méthodes d'apprentissage pour la génération de données, en partie causé par le développement de la modélisation de fonctions complexes par méthodes connexionistes, a permis le développement récent de nouvelles approches boîte noire pour la synthèse.

Dans le domaine de l'image, des systèmes tels les réseaux antagonistes génératifs ou les auto-encodeurs variationnels (voir section 3) permirent l'extension de ces approches au domaine de la génération. Néanmoins, leur application au domaine sonore est assez récente, majoritairement représentée par des systèmes auto-régressifs neuronaux [25, 23], de méthodes hybrides [10] ou encore sur l'utilisation des ces méthodes pour l'inversion de transformées audio non-inversibles [20].

Néanmoins, en dehors des auto-encodeurs variationnels, aucun de ces systèmes n'est basé sur l'extraction d'une représentation haut-niveau. En effet, ces systèmes sont basés sur une procédure non-supervisée en deux temps : l'extraction d'un espace *génératif* à partir des données par un processus d'*encodage*, et un processus de *décodage* garantissant la réversibilité de cette représentation.

De plus, ces méthodes d'apprentissage de représentation se sont montrées fortement flexibles, permettant le développement de techniques dites *semi-supervisées* pouvant inclure des méta-données symboliques (voir [18]), le développement d'approches trans-modales [15], ou encore le désenchevêtrement de facteurs génératifs [14]. Majoritairement utilisées dans le domaine de l'image, leur utilisation dans le domaine de l'audio a été par la suite proposée par [11], proposant une méthode de régularisation perceptive de ces espaces génératifs, ou encore pour la génération trans-modale signal/symbole [8].

2.2. Méthodes d'analyse-synthèse

Dans le domaine du traitement de signaux audio, les approches d'analyse et de synthèse sont généralement basés sur des processus séparés. Pour l'analyse, un nombre conséquent de méthodes basées sur l'extraction de propriétés perceptives par descripteurs acoustiques [24], ou de propriétés dynamiques par l'analyse de profils d'énergie ou de segmentation [27], permettent d'être par la suite utilisées pour l'inférence de propriétés musicales haut-niveau telle la reconnaissance de structure, de genre, d'artiste, et bien d'autres [32].

Dans le domaine de la synthèse numérique, la diversité des approches possibles pour la génération de contenus audio est conséquente [9, 29, 2]. Néanmoins, la plupart de ces approches sont contrôlées par une ensemble plus ou moins grand de paramètres inhérents à la structure de l'algorithme, et ne sont pas à priori directement issus d'une analyse préalable. Ainsi, un ensemble de méthodes d'analyse-synthèse se basent sur la projection d'un phénomène sonore dans une représentation alternative du signal correspondant, permettant l'altération de signaux existants ou même la composition directe de nouveaux sons dans ce nouvel espace. Néanmoins, l'inversion de la représentation utilisée est nécessaire pour permettre ce processus d'analyse-synthèse, à l'instar du vocodeur de phase [12], utilisant la transformée de Fourier court-terme, ou encore par ondelettes inversibles comme la transformée de Gabor non-stationnaire [3]. Néanmoins, l'inversibilité générale de ces représentations contraint leur dimensionnalité à être au moins équivalentes à celle des données analysées, nécessitant ainsi des systèmes de contrôles additionnels pour afin d'être facilement utilisées dans des contextes interactifs.

3. MODÈLES BAYÉSIENS ET APPRENTISSAGE VARIATIONNEL

La méthode de génération proposée par cet article se base sur la notion de modèles dits *latents*, une notion originellement associée au domaine de l'inférence bayésienne.

3.1. Méthodes bayésiennes et espaces latents

3.1.1. Inférence bayésienne

L'inférence bayésienne est une approche probabiliste consistant à considérer un ensemble de données $\mathbf{x}$ comme des échantillons d'une distribution sous-jacente $p(\mathbf{x})$, qu'il s'agit donc de modéliser afin d'en extraire leur structure. Cette distribution de probabilité peut ainsi dépendre d'un ensemble de paramètres $\boldsymbol{\theta}$, pouvant être de même considérés comme des variables aléatoires de distribution *a priori* $p(\boldsymbol{\theta})$: par exemple, une distribution normale $x \sim \mathcal{N}(\mu, \sigma^2)$ avec pour paramètres $\mu \sim p(\mu)$ et $\sigma^2 \sim p(\sigma^2)$. Dans ce cas, l'inférence bayésienne consiste à obtenir la distribution *à posteriori* de ces paramètres $p(\boldsymbol{\theta}|\mathbf{x})$ après l'observation des données $\mathbf{x}$ grâce à l'identité suivante, nommée *théorème de Bayes*

$$p(\boldsymbol{\theta}|\mathbf{x}) = \frac{p(\mathbf{x}|\boldsymbol{\theta})p(\boldsymbol{\theta})}{\int p(\mathbf{x}|\boldsymbol{\theta})d\boldsymbol{\theta}} \tag{1}$$

Ce théorème permet donc d'obtenir la distribution de probabilité des paramètres $\boldsymbol{\theta}$ suite à l'observation des données $\mathbf{x}$, pouvant ensuite être utilisée pour générer de nouveaux échantillons $\mathbf{x}$ grâce à la distribution générative $p(\mathbf{x}|\boldsymbol{\theta})$. Néanmoins, dans ce cas les paramètres $\boldsymbol{\theta}$ de la distribution générative $p(\mathbf{x}|\boldsymbol{\theta})$ sont globaux à tous les exemples de la base de données, pouvant donc *caractériser* les données analysées mais ne permettant pas d'en *contrôler* la génération. Pour ce faire, le modèle peut être étendu par l'ajout de nouvelles variables aléatoires $\mathbf{z} \sim p(\mathbf{z})$, appelées variables *latentes*, pouvant être considérées comme des paramètres libres ajoutés afin d'enrichir l'expressivité du modèle. Dans la mesure où ces variables latentes sont aussi aléatoires, le théorème de Bayes peut alors être écrit

$$p_{\boldsymbol{\theta}}(\mathbf{z}|\mathbf{x}) = \frac{p_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{z})p(\mathbf{z})}{\int p_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{z})d\mathbf{z}} \tag{2}$$

3.1.2. Méthodes variationnelles

Dans la mesure où la distribution postérieure $p(\mathbf{z}|\mathbf{x};\boldsymbol{\theta})$ n'a pas d'expression analytique en général (à cause de l'intégrale du terme $p(\mathbf{x}) = \int p_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{z})d\mathbf{z}$), l'identification systématique de la distribution de probabilité conjointe $p(\mathbf{x}, \mathbf{z})$ par calcul direct est impossible. Cette impossibilité nécessite donc l'usage de méthodes d'approximation à l'instar des méthodes dites *variationnelles*, remplaçant la distribution $p_{\boldsymbol{\theta}}(\mathbf{z}|\mathbf{x})$ exacte par une approximation $q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x})$, paramétrisée par ϕ, pouvant être choisie librement. De cette manière, la log-probabilité marginale $p(\mathbf{x})$ peut être exprimée (voir [6])

$$\log p(\mathbf{x}) = \mathcal{L}(q, \boldsymbol{\theta}, \boldsymbol{\phi}) + D_{KL}[q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x})\|p_{\boldsymbol{\theta}}(\mathbf{z}|\mathbf{x})] \tag{3}$$

où le premier terme $\mathcal{L}(q, \boldsymbol{\theta}, \boldsymbol{\phi})$ est défini par

$$\mathcal{L}(q, \boldsymbol{\theta}, \boldsymbol{\phi}) = \int q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x}) \log \frac{p_{\boldsymbol{\theta}}(\mathbf{x}, \mathbf{z})}{q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x})} d\mathbf{z} \tag{4}$$

et le second-terme est la divergence de Kullback-Leibler entre la distribution variationnelle $q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x})$ et la distribution

postérieure $p_{\boldsymbol{\theta}}(\mathbf{z}|\mathbf{x})$. Dans la mesure où cette divergence est positive $\forall(p, q)$, nous pouvons exprimer l'inégalité

$$\log p(\mathbf{x}) \geq \mathcal{L}(q, \boldsymbol{\theta}, \boldsymbol{\phi}) \tag{5}$$

de sorte que si la borne $\mathcal{L}(q, \boldsymbol{\theta}, \boldsymbol{\phi})$ est exprimable analytiquement, sa maximisation optimisera de la même manière la log-probabilité marginale $p(\mathbf{x})$.

3.2. Auto-encodage variationnel

Bien qu'initialement les méthodes variationnelles soient basées sur le choix de distributions simples pour le modèle $q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x})$, permettant ainsi l'expression analytique de la borne $\mathcal{L}(q, \boldsymbol{\theta}, \boldsymbol{\phi})$, ce choix peut être encore trop restrictif pour la modélisation de modèles stochastiques complexes.

Une méthode possible pour enrichir les distributions conditionnelles $q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x})$ et $p_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{z})$ est de recourir à des réseaux de neurones artificiels, qui peuvent être pensés comme des approximations de fonctions complexes, pour modéliser les paramètres des distributions $q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x})$ et $p_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{z})$. Cependant, l'utilisation de ces approches de type boîte noire nécessite des procédures fondées sur des techniques d'optimisation par *descente de gradient*, et donc la définition d'un critère d'entraînement exprimable analytiquement.

Dans la mesure où la maximisation de la borne $\mathcal{L}(q, \boldsymbol{\theta}, \boldsymbol{\phi})$ entraîne l'optimisation de la log-probabilité marginale $p(\mathbf{x})$, elle peut ainsi être choisie comme critère d'entraînement pour ce modèle étendu. Étant donné une distribution a priori $p(\mathbf{z})$, il est possible de reformuler l'équation (5) :

$$\begin{aligned}\mathcal{L}(q, \boldsymbol{\theta}, \boldsymbol{\phi}) = \mathbb{E}_{q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x})}\big[\log p_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{z})\big] \\ - D_{KL}\big[q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x}) \| p(\mathbf{z})\big]\end{aligned} \tag{6}$$

On peut constater que cette égalité sépare (5) en deux termes distincts : l'espérance de la log-probabilité $p_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{z})$ par rapport à $q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x})$, qui peut être assimilé comme un terme *reconstruction*, et une divergence entre le modèle variationnel $q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x})$ et la distribution à priori $p(\mathbf{z})$, agissant comme une *régularisation* de la distribution a posteriori. Dans la mesure où tous les termes impliqués dans cette décomposition peuvent être choisis librement (sous condition d'avoir une forme fermée pour le terme de régularisation), le gradient requis peut être obtenu [16] en posant :

$$\begin{aligned}\nabla_{\boldsymbol{\theta},\boldsymbol{\phi}}\mathcal{L}(q, \boldsymbol{\theta}, \boldsymbol{\phi}) = \nabla_{\boldsymbol{\theta},\boldsymbol{\phi}}\mathbb{E}_{q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x})}\big[\log p_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{z})\big] \\ - \nabla_{\boldsymbol{\phi}} D_{KL}\big[q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x}) \| p(\mathbf{z})\big]\end{aligned} \tag{7}$$

de manière à optimiser $-\mathcal{L}(q, \boldsymbol{\theta}, \boldsymbol{\phi})$ par descente de gradient. En paramétrant le système de la manière suivante

$$q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x}) = \mathcal{N}\big(\boldsymbol{\mu}_{\boldsymbol{\phi}}(\mathbf{x}), \boldsymbol{\sigma}^2_{\boldsymbol{\phi}}(\mathbf{x})\big) \tag{8}$$

$$p_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{z}) = \mathcal{N}\big(\boldsymbol{\mu}_{\boldsymbol{\theta}}(\mathbf{z}), \boldsymbol{\sigma}^2_{\boldsymbol{\theta}}(\mathbf{z})\big) \tag{9}$$

de telle manière que les fonctions $\boldsymbol{\mu}_{\boldsymbol{\phi}}, \boldsymbol{\sigma}^2_{\boldsymbol{\phi}}, \boldsymbol{\mu}_{\boldsymbol{\theta}}$ et $\boldsymbol{\sigma}^2_{\boldsymbol{\theta}}$ soient représentées par des réseaux de neurones artificiels (de paramètres respectifs $\boldsymbol{\phi}$ et $\boldsymbol{\theta}$), nous obtenons l'*auto-encodeur variationnel*, proposé par [19] (voir figure 2a). Ce système est donc basé sur une formulation variationnelle de la distribution de probabilité conjointe $p(\mathbf{x}, \mathbf{z})$ entre l'espace des données $\mathbf{x}$ et l'espace latent $\mathbf{z}$, paramétrant les distributions $q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x})$ et $p_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{z})$ avec une modélisation de fonctions haute capacité pouvant exprimer des relations complexes entre ces deux espaces.

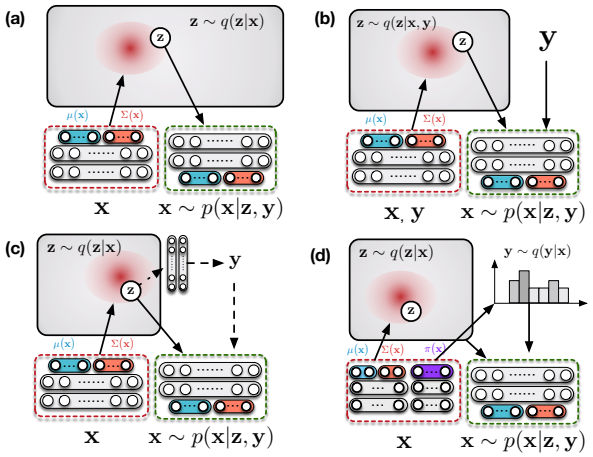

Figure 2. (a) Schéma fonctionnel d'un auto-encodeur variationnel. (b) Conditionnement du système par concaténation (*supervisé*), la variable de classe $\mathbf{y}$ étant ajoutée comme entrée supplémentaire et donc comme conditionnement de la distribution de sortie. (c) Utilisation d'un classificateur auxiliaire pour inférer $\mathbf{y}$, donnée ensuite au décodeur (*semi-supervisé*). (d) Utilisation des variables de classe $\mathbf{y}$ comme variables aléatoires latentes supplémentaires.

3.3. Améliorations et augmentations

Depuis la formulation originelle de ces systèmes [19], de nombreuses propositions d'amélioration et d'extension de ce modèle de base ont été développées par la communauté. Nous présenterons ici uniquement celles déjà implantées dans notre boite à outils et présentant un intérêt du point de vue interactif.

3.3.1. Apprentissage (semi-)supervisé

L'apprentissage des auto-encodeurs variationnels se fait de manière totalement *non-supervisée*, basée sur l'extraction automatique d'une représentation à partir des données même. Néanmoins, l'ajout d'informations *symboliques* issues de méta-données permettrait, en cas de disponibilité, d'influer la structure de l'espace obtenu à partir de ces données auxiliaires.

Parmi les approches possibles d'influence symbolique, nous en distinguons deux, toutes deux implantées dans notre système.

La première, appelée approche par *conditionnement*, consiste à fournir l'information de classe $\mathbf{y} \in [0...L]$ correspondant à une donnée $\mathbf{x}$ au décodeur et/ou à l'encodeur, modélisant ainsi les distributions $p_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{z}, \mathbf{y})$ et $q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x}, \mathbf{y})$. Ce faisant, nous obtenons une représentation $\mathbf{z}$ dépendante de l'information symbolique $\mathbf{y}$, ayant alors un espace latent

différent par classe (voir figure 2b). Cette méthode est très efficace pour contraindre la génération à une classe précise, et faciliter leur intégration dans des contextes d'utilisation (conditionnement par hauteur, typologies spécifiques, etc.).

Cependant, cette première méthode requiert la disponibilité du couple $(\mathbf{x}, \mathbf{y})$ pour tous les éléments de la base de données, ce qui peut être difficile dans le cas d'ensemble partiellement annotés.

La seconde méthode, proposée par Kingma et coll. [18] et appelée apprentissage *semi-supervisé*, consiste plutôt à permettre l'inférence des données symboliques $\mathbf{y}$ en cas de base de données incomplète. Une première manière consiste à entraîner un classificateur à partir de l'espace latent $\mathbf{z}$, entraîné de manière jointe avec l'auto-encodeur en cas d'information disponible, ou bien utilisé comme prédiction en cas d'information indisponible (voir figure 2c). Une seconde manière consiste à considérer les variables $\mathbf{y}$ comme variables latentes discrètes, pouvant ainsi être ajoutées à la borne variationnelle (5), pour donner l'expression d'une ELBO conditionnelle semi-supervisée $\mathcal{L}_s(q, \boldsymbol{\theta}, \boldsymbol{\phi})$ (voir figure 2d).

$$\begin{aligned}\mathcal{L}_s(q, \boldsymbol{\theta}, \boldsymbol{\phi}) = \mathbb{E}_{q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x},\mathbf{y})}\big[\log p_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{z},\mathbf{y}) + \log p(\mathbf{y})\big] \\ - D_{KL}\big[q_{\boldsymbol{\phi}}(\mathbf{z},\mathbf{y}|\mathbf{x})\|p(\mathbf{y})\big] \qquad (10)\end{aligned}$$

où la log-probabilité $\log p(\mathbf{y})$ est généralement une distribution multinomiale

$$\begin{aligned}\mathbf{y} \sim \text{Multinomial}\big[\boldsymbol{\pi}(\mathbf{z})\big] \qquad (11) \\ \sum_{i=1}^{L} \pi_i = 1\end{aligned}$$

où les probabilités π_i pour un $\mathbf{z}$ d'appartenir à la classe i sont générées par un réseau discriminateur. Si l'information de classe $\mathbf{y}$ est manquante, on peut marginaliser sur l'ensemble des classes afin d'obtenir l'ELBO conditionnelle non-supervisée $\mathcal{L}_u(q, \boldsymbol{\theta}, \boldsymbol{\phi})$

$$\mathcal{L}_u(q, \boldsymbol{\theta}, \boldsymbol{\phi}) = \sum_{\mathbf{y}} \mathcal{L}_s(q, \boldsymbol{\theta}, \boldsymbol{\phi}) + \mathbb{H}\big[q(\mathbf{y}|\mathbf{x})\big] \qquad (12)$$

où $\mathbb{H}$ dénote l'entropie de Shannon de la distribution symbolique inferrée. Dans le premier cas, l'ajout d'un classificateur basse capacité à partir de l'espace latent $\mathbf{z}$ incitera l'encodeur à segmenter l'espace latent $\mathbf{z}$ afin de lui permettre de séparer correctement les classes, permettant ainsi d'agir sur la structure de l'espace génératif. A l'inverse, la seconde méthode incite le désenchevêtrement de l'espace latent continu $\mathbf{z}$ et de l'espace latent symbolique $\mathbf{y}$, permettant ainsi de modéliser des espaces génératifs indépendants d'une information de classe donnée.

3.3.2. Régularisations alternatives

Le terme de régularisation, c'est à dire la divergence de Kullback-Leibler $D_{KL}[q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x})\|p(\mathbf{z})]$ émergeant naturellement dans la formulation de (5), est fondamental pour la consistance de la représentation obtenue, poussant le terme $q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x})$ vers la distribution à priori $p(\mathbf{z}) = \mathcal{N}(\mathbf{0}, \mathbb{I})$ et forçant donc les différents exemples de la base de données à partager de l'information et ainsi garantir la continuité de l'espace [17]. Néanmoins, ce processus de régularisation peut également mener à des représentations sous-optimales, dûes premièrement aux propriétés inhérentes à la divergence de Kullback-Leibler, et deuxièmement à la régularisation de $q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x})$ sur $p(\mathbf{z})$: en effet, si $q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x}) = p(\mathbf{z})$, alors la représentation $\mathbf{z}$ devient indépendante des données $\mathbf{x}$ et devient donc *inactive* (le processus d'analyse ayant ainsi disparu du modèle).

Pour contrer ces effets négatifs, plusieurs auteurs ont donc proposé de recourir à des régularisations alternatives pour remplacer $D_{KL}[q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x})\|p(\mathbf{z})]$. D'une part, remplacer la divergence de Kullback-Leibler par d'autres divergences plus robustes et paramétrables, modelant ainsi la distribution des exemples dans l'espace latent. Deux divergences alternatives sont pour le moment implantées dans la boîte à outils : la *divergence de Rényi*, une généralisation de la D_{KL} permettant de contrôler le comportement englobant / focalisant grâce à un paramètre α [21], et la divergence de Jensen-Shannon généralisée, une version symétrique et lissée de la D_{KL}.

Néanmoins, ces divergences alternatives de permettent pas de pallier la dégénérescence induite par la comparaison de $q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x})$ et $p(\mathbf{z})$. Ainsi, certaines méthodes proposent donc de plutôt régulariser la divergence entre la distribution a posteriori *agrégée* $q(\mathbf{z}) = \int q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x})d\mathbf{z}$ avec la distribution a priori $p(\mathbf{z})$. Néanmoins, dans la mesure ou l'expression de $q(\mathbf{z})$ n'est pas exprimable analytiquement, des tests statistiques permettant la comparaison de distributions *implicites* doivent être utilisées. Une première méthode, l'*auto-encodeur antagoniste*, propose d'estimer cette divergence par l'entraînement d'un réseau discriminateur auxiliaire visant à séparer les échantillons issus de $p(\mathbf{z})$ et ceux de $q(\mathbf{z})$ (voir [22]). Une méthode alternative, proposée par [34], propose plutôt de recourir à la *divergence maximale de moyenne*, un test statistique visant à mesurer l'écart entre deux distributions seulement à partir d'échantillons. Dans les deux cas, l'objectif est bien de rapprocher $q(\mathbf{z})$ de $p(\mathbf{z})$, permettant de prévenir la dégénérescence des représentations latentes et d'améliorer les reconstructions, mais qui à l'inverse ne force pas le partage d'information entre les différents exemples de la base de données.

4. DE L'AUTO-ENCODAGE VARIATIONNEL COMME MÉTHODE DE SYNTHÈSE SONORE

Bien qu'en première apparence l'intérêt créatif d'utiliser des systèmes basés sur la *reconstruction* d'exemples audio puisse sembler équivoque, il est néanmoins multiple. En premier lieu, les hautes capacités démontrées de ces méthodes nous permettent d'extraire des représentations à faible dimensionnalité malgré une erreur de reconstruction très convaincante (pouvant diviser la dimensionalité jusqu'à 128 sans perte notable), démontrant la capacité du système à extraire automatiquement des facteurs générateurs continus $\mathbf{z}$ à partir de l'ensemble de données analysées $\mathbf{x}$ (pour une analyse quantitative, se référer à [7]). Bien que

l'adéquation entre les dimensions latentes $\mathbf{z}$ extraites de manière non-supervisée et les réels facteurs génératifs des données analysées ne soit pas garantie sans conditionnement spécifique, nous pensons que ces systèmes peuvent proposer une formule originale pour contrôler divers algorithmes de synthèse par réseaux de neurones, permettant ainsi une interaction riche avec ce type de modèles, absente de ses équivalents (comme les réseaux antagonistes). L'exploration de cette représentation permet de synthétiser un quantité infinie de contenus sonores n'existant pas au préalable dans la base d'apprentissage, auto-organisée par l'algorithme et permettant ainsi une interaction riche avec l'utilisateur, pouvant être très proches ou au contraire très éloignés de la distribution des exemples de base, permettant ainsi une synthèse réaliste ou au contraire la production d'artefacts.

De plus, la flexibilité de ces méthodes, en plus d'autoriser plusieurs applications créatives intéressantes (voir section 4), permettent la définition haut-niveau de propriétés structurelles du modèle (définition des distributions impliquées, conditionnement, stratégies de régularisation, ajout de critères spécifiques), la construction du modèle pouvant elle-même être intégrée dans le processus créatif de l'utilisateur. Néanmoins, ces méthodes ayant été très peu utilisées dans des cadres applicatifs et créatifs, nous trouvons fondamental de permettre une approche *expérimentale* de ce genre de méthode par l'ouverture de ces méthodes, à la fois pour valider les propriétés émergentes de ces algorithmes et pour valider leur intérêt d'un point de vue sonore et compositionnel. Dans cette section, nous développons d'abord le côté technique du procédé de synthèse analysé, ainsi que l'ensemble des applications créatives permises par le système ; dans la section suivante, nous décrirons brièvement les fonctionnalités de la bibliothèque open-source que nous mettons à disposition.

4.1. Données et transformées audio

4.1.1. Représentations d'entrées

Dans la mesure où les auto-encodeurs variationnels sont en mesure d'apprendre sur une base de données $\{\mathbf{X}\}$ de N variables multi-variées $\mathbf{X} = \{\mathbf{x}_i\}_{i=1...N}$ sans condition supplémentaire, l'apprentissage de données audio implique de sélectionner une représentation d'entrée adéquate pour garantir la convergence de l'apprentissage et la restitution correcte des contenus sonores correspondants.

La représentation la plus directe d'un signal est la *forme d'onde*, correspondant à la description temporelle de l'onde de pression acoustique portant le signal sonore. Dans la mesure ou la dimensionnalité d'entrée d'un auto-encodeur variationnel est fixe, apprendre la forme d'onde requerrait de découper le signal sonore en petites unités, que l'on peut appeler *grains* en référence à la synthèse granulaire, desquels seraient inféré une représentation latente pouvant donc être pensée comme un espace auto-organisé de grains sonores. Néanmoins, cette représentation d'entrée est particulièrement sensible à l'effet de lissage réalisé par l'auto-encodeur variationnel (à cause de la généralisation apportée par l'apprentissage bayésien), et la variabilité temporelle d'un signal dont le contenu est pourtant perceptiblement stationnaire peut altérer la qualité des générations obtenues.

Une solution pour remédier à ce problème est l'utilisation de transformées spectrales à court terme, permettant d'obtenir des représentations *fréquentielles* locales du signal. Cependant, ces représentations (*représentation de Fourier court-terme* ou la *transformée de Gabor non-stationnaire*) sont des représentations complexes $\mathbf{x} \in \mathbb{C}^D$, et donc incompatibles avec les réseaux neuronaux classiques (la notion de dérivabilité étant singulière dans le domaine complexe). Heureusement, ces transformées permettent de détacher l'aspect stationnaire d'un contenu fréquentiel de son aspect temporel par la décomposition d'un contenu spectral entre son *amplitude* et sa *phase*. Ainsi, nous pouvons apprendre avec des auto-encodeurs variationnels le contenu fréquentiel d'un son (divisant ainsi la dimensionnalité du signal d'entrée par 2), et reconstruire l'aspect dynamique du signal soit en reconstruisant sa phase avec l'algorithme Griffin-Lim [28], soit en récupérant la phase du signal d'entrée (s'il est disponible). L'usage de représentations spectrales non-complexes, comme la transformée cosinus discrète, a été essayée dans un précédent article, et s'est révelée impropre pour un apprentissage correct de l'auto-encodeur [11].

4.2. Base de données

L'intérêt de choisir un processus bayésien et la robustesse des représentations obtenues, permettant d'entraîner à la fois sur des petites bases de données sans risque de sur-apprentissage de la part des réseaux neuronaux, mais pouvant pareillement être entraînés sur des basés de données de grande envergure. De plus, ces méthodes ont un taux de convergence très rapide, permettant un entraînement satisfaisant sur CPU pour des bases de données de taille modeste et ne nécessitant donc pas l'accès à des solutions de calcul parallèle (accélérant néanmoins considérablement l'apprentissage).

Afin de procéder à l'évaluation expérimentale de la capacité auto-organisatrice de ces modèles, nous proposons aussi une approche par *base de données de jeu*, permettant de décrire leur performance sur des données simples. Ces bases de données, pourtant communes dans le domaine de la génération d'images, est encore inexistante dans la génération audio, permettant pourtant d'évaluer qualitativement et quantitativement le modèle autour de facteurs de variations simples et précis. Pour ce faire, nous proposons dans notre boîte à outils un générateur de base de données de jeu automatisé autour de petites unités de synthèse, échantillonnant en grille ses paramètres. Ce procédé simple peut permettre ainsi de générer une grande quantité de données organisées autour de principes générateurs précis, permettant ensuite de rapidement expérimenter le comportement de ces systèmes sur des facteurs de variation simples. Les deux premières que nous avons utilisées, une basée sur la synthèse additive (`toy_additive_mini`) et

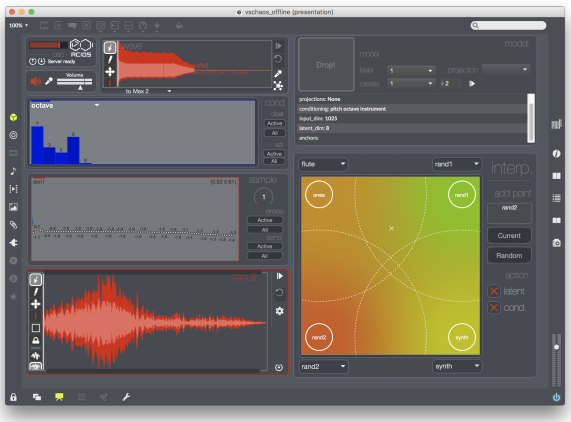

Figure 3. Interface pour l'exploitation des modèles en contexte hors-temps.

l'autre sur de la synthèse par modulation de fréquence (`toy_fm`), sont disponibles ici.
Enfin, pour tester les capacités de reconstruction sur des bases de données provenant d'enregistrements sonores, nos modèles sont entraînés avec des extraits audio provenant de la banque *Studio Online* [4], sur laquelle nous avons prélevé deux sous-bases : une base, `acidsInstruments-ordinario`, avec les sons *ordinario* de 11 instruments prélevés de l'intégralité de leur tessiture ainsi que 3 modes dynamiques différents (*pp*, *mf*, *ff*), et un ensemble `acidsInstruments-violin` de sons bruités de violon , afin d'évaluer la performance des algorithmes sur des bases de taille restreinte.

4.3. Applications créatives

Un des intérêts de l'usage de l'auto-encodage variationnel est la multiplicité des utilisations possibles pour la synthèse, à la fois dans des cadres hors-temps (dans des contextes de composition ou de production) et temps-réel (dans des contextes de performance). Dans cette section, nous présentons de manière succincte les différents processus de synthèse permis par ce système, ainsi que des exemples audio disponibles sur notre page support [1].

4.3.1. Applications hors-temps

Les contextes hors-temps, dans la mesure où la temporalité de la génération est décorrelée de la temporalité de la création, permettent d'une part un usage récursif des modèles, c'est-à-dire une démarche de génération/correction pouvant être intégrée dans des situations de composition, ainsi que la transformation de sons pré-existants. Ainsi, l'interaction hors temps avec ces modèles consiste en la génération de *trajectoires* latentes, soit par manipulation directe, à l'aide d'un générateur de trajectoire, où encore grâce à des trajectoires latentes issues de l'encodage d'un fichier audio pris à l'intérieur où a l'extérieur de la base de données d'entraînement.

[1] `domkirke.github.io/vschaos_package`, accédé le 21 octobre 2020

Génération par trajectoires. La génération de nouveaux contenus audio peut être effectuée par le biais d'un générateur de trajectoires (lignes, ellipsoïdes, marches aléatoires, etc.), pouvant être paramétrées par des contrôles globaux (amplitude, vitesse, origine, etc.) et l'écoute récursive des sons obtenus.

Génération par interpolation. Dans la mesure ou un son entier est ici encodé comme une séquence de positions latentes, les trajectoires obtenues par l'encodage de deux fichiers audio de la base de données peuvent donc être interpolées directement dans l'espace génératif, et ensuite être décodées pour obtenir les sons correspondants. Cette méthode peut donc être pensées comme une « interpolation haut-niveau », réalisant l'interpolation sur la sous-variété extraite par le modèle.

Génération par transfert. La génération dite *par transfert* consiste à entraîner un modèle sur un ensemble de données considéré comme appartenant à un domaine spécifique, pour ensuite encoder puis décoder des données sonores sensiblement ou considérablement différentes du domaine appris. Cette méthode de génération permet donc à la fois d'évaluer intuitivement les capacités de généralisation du modèle, c'est à dire à quelle mesure les caractéristiques apprises sont spécifique au domaine, mais aussi d'obtenir une traduction généralement très différente du fichier d'origine.

Génération par traduction. Si le modèle utilisé est conditionné par une information symbolique, l'espace latent obtenu devient dépendant de son conditionnement, de sorte qu'un fichier audio peut être encodé avec un conditionnement donné, puis régénéré avec une information différente. Ce processus, que nous pouvons appeler *traduction*, permet ainsi de projeter un son sur la variété extraite d'un signal de conditionnement différent, et donc à la fois d'évaluer les capacités d'extrapolation du système, et de transformer un signal existant avec un processus de décodage différent.

4.3.2. Interactions temps-réel

Dans la mesure ou, dans des contextes temps-réel, le temps de la génération est simultané au temps de la création, elle est donc possible soit par l'interaction directe avec la représentation latente de l'algorithme, soit par l'encodage/décodage en temps réel d'un signal sonore entrant.

Exploration temps-réel. De par la légèreté du processus d'auto-encodage variationnel, il est possible d'interagir directement avec les dimensions latentes et les différentes variables de conditionnement afin de générer en temps réel de nouveaux artefacts sonores. Dans le cas d'un trop grand nombre de dimensions, l'utilisation de techniques de réduction dimensionnelles comme l'analyse en composantes principales (PCA) ou l'analyse en composantes indépendantes (ICA) inversibles avec perte, peut permettre de réduire le nombre de paramètres de manière à faciliter l'interaction.

Transfert/traduction temps-réel. Dans un contexte performatif, il est aussi possible de faire passer un flux audio venant d'une source enregistrée dans l'auto-encodeur, puis

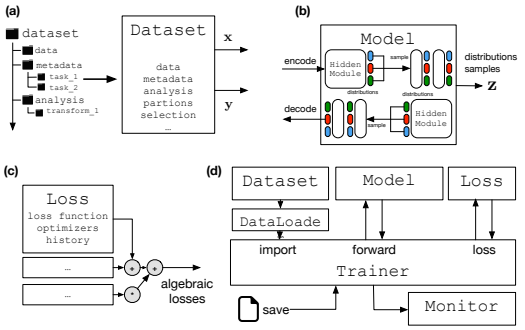

Figure 4. Les 3 objets principaux de notre bibliothèque : (a) `Dataset`, (b) `Model`, (c) `Loss`, et (d) entraînement global.

de décoder les positions latentes obtenues pour générer par transfert en temps réel. Dans le cas d'un modèle conditionné, il est aussi possible de donner une information symbolique différente, afin de générer par traduction.

5. DESIGN ET IMPLANTATION

Afin de permettre cette approche expérimentale de la synthèse neurale par auto-encodage variationnel, nous proposons une bibliothèque open-source, vschaos, basée sur la bibliothèque pytorch [26], destinée à la fois à l'exploitation de modèle pré-entraînés, implantant tous les modes de génération présentés dans la partie précédente pour des utilisateurs non-expérimentés, et pour l'apprentissage haut-niveau de nouveaux modèles, facilitant l'apprentissage de ces systèmes sur des données audio.

5.1. Description de la bibliothèque

5.1.1. Base de données audio et transformées

Dans la mesure où les processus d'apprentissage bayésiens peuvent fournir des performances satisfaisantes même dans le cas de données peu nombreuses, il est tout à fait possible à un utilisateur d'entraîner un modèle sur une base de donnée personnalisée. De même, la spécification de méta-données pouvant être utilisées dans le conditionnement devraient pouvoir être facilement intégrées. vschaos définit donc un formalisme de base de données, utilisé par l'objet `Dataset`, visant à facilement pouvoir importer des données quelconques facilitant l'import de méta-données. Une classe additionnelle, `DatasetAudio`, permet aussi de gérer automatiquement les transformées audio les plus communément utilisées, pouvant être chargé de façon synchrone ou asynchrone.

5.1.2. Définition de modèles

L'objectif de la bibliothèque vschaos est aussi de faciliter la création de modèles, fournissant des méthodes de construction haut niveau permettant de définir la structure du système avec un simple système de signature spécifiant les propriétés des modules d'encodage/décodage, le conditionnement, ainsi que le type des distributions variationnelle et générative choisis. La procédure d'entraînement est

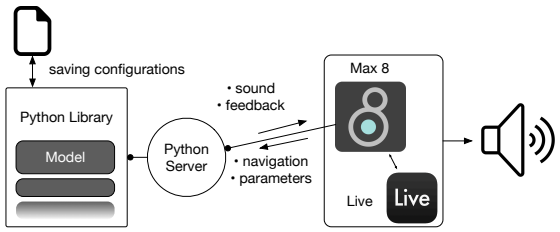

Figure 5. Schéma de la configuration de génération temps-réel avec vschaos.

entièrement prise en charge par une classe d'entraînement, appelée `Trainer`, implantant la plupart des étapes standard de l'apprentissage machine, pour faciliter cette étape aux utilisateurs non-experts. Néanmoins, chaque étape de l'apprentissage peut être spécifiée, afin de permettre le développement de procédures spécialisées.

5.1.3. Critères d'entraînement

Tous les critères d'entraînement disponibles dans la bibliothèque à la fois pour les critères de reconstruction et de régularisation dérivent d'une classe abstraite, appelée *Loss*, implantant la plupart des opérations algébriques pour permettre la définition de critères personnalisés, et un système d'historique, permettant l'enregistrement automatique des profils d'erreur au cours de l'entraînement. Les critères d'entraînement implantés permettent de recouvrir de nombres systèmes existants tels l'auto-encodeur, l'auto-encodeur variationnel [19], le β-VAE [15], l'auto-encodeur Wasserstein [34], l'auto-encodeur antagoniste [22], l'auto-encodeur Rényi [21], et toute composition arbitraire à explorer par l'utilisateur.

5.2. Interfaces utilisateur

Alors que le rassemblement, le traitement des données, ainsi que l'entraînement du modèle se font par l'intermédiaire de Python, les exploitations hors-temps et temps-réel du modèle exigent une interface appropriée pouvant être directement appréhendée par l'utilisateur. Les deux interfaces proposées, conçues à l'aide du logiciel *Max*[2], et utilisant les bibliothèques *MuBu* [30] et *Bach* [1], permettent l'exploitation de modèles pré-entraînés dans pour les utilisations créatives présentées en section 4.3. Ces interfaces communiquent avec un serveur Python externe par l'intermédiaire du protocole réseau OSC, permettant par exemple l'utilisation à distance de GPUs externes (voir figure 5).

5.2.1. Interface hors-temps

Cette interface interagit en temps réel avec le serveur Python pour l'importation et de la transformation de fichiers audio externes, la spécification de conditionnements, la manipulation de trajectoires latentes et l'interpolation entre des séquences latentes mémorisées (voir figure 3). La

[2] `cycling74.com`, accédé le 21 octobre 2020

Figure 6. Interfaces pour l'exploration d'espaces latents en temps réel.

synthèse à partir de trajectoires est effectuée dans l'environnement Python, afin de permettre l'utilisation d'algorithmes de reconstruction de phase qui seraient coûteuses dans des cadres temps réel.

5.2.2. Interfaces temps-réel

Pour le cadre temps-réel, la trajectoire donnée par l'utilisateur et la synthèse doivent être générée simultanément. Ainsi, les interactions possibles sont d'une part la manipulation directe (permettant le contrôle MIDI avec des contrôleurs externes) des dimensions latentes, où bien la resynthèse d'un signal audio rentrant. Pour le moment, cette synthèse est générée en temps réel par l'environnement *Max* (voir figure 6), soit par transformée de Fourier inverse, soit par banque de sinus.

6. CONCLUSIONS ET PERSPECTIVES

Dans cet article, nous proposons l'usage de l'auto-encodage variationnel comme technique inédite de synthèse sonore, basée sur l'*inférence* d'une représentation inversible à base dimensionnalité à partir d'une base de donnée restreinte, et la *génération* de contenus sonores à partir des facteurs génératifs extraits. Afin de permettre une approche *expérimentale* de ces nouvelles techniques de génération, encore employées de manière marginale dans le domaine de la génération audio, nous fournissons une bibliothèque open-source, vschaos, permettant à la fois l'exploitation hors-temps ou temps-réel de modèles pré-entraînés (à travers des interfaces *Max* & *Max4Live*), et des fonctionnalités haut-niveau pour la définition et l'entraînement de nouveaux modèles sur des bases de données arbitraires, afin d'inclure aussi la conception de modèle dans le flux créatif de l'utilisateur. Ainsi, nous espérons que cette bibliothèque permette l'investigation de ces nouvelles techniques de génération selon un axe réflexif de recherche & création, permettant ainsi de mieux qualifier les propriétés émergentes de ces modèles par leur intégration dans des processus artistiques pour ainsi contribuer à leur développement, et inversement de créer de nouvelles pratiques musicales par l'utilisation d'une technique de synthèse novatrice. Cette bibliothèque devrait servir de base à de multiples travaux futurs, notamment l'amélioration de l'interface, l'implantation de nouveaux type de modèles, de modèles pré-entraînés, et de nouveaux types d'interaction.

7. RÉFÉRENCES

[1] Agostini, A., Ghisi, D. « Bach: An environment for computer-aided composition in max », *Proceedings of the International Computer Music Conference*, Ljubljana, Slovénie, 2012.

[2] Avanzini, F., Rocchesso, D. « Controlling material properties in physical models of sounding objects. », *Proceedings of the International Computer Music Conference*, La Havane, Cuba, 2001.

[3] Balazs, P., M. D. Orfler, N. Holighaus, F. Jaillet, G. A. Velasco. « Theory, implementation and applications of nonstationary gabor frames » *Journal of Computational and Applied Mathematics* 236/6 (2011), p. 1481-1496.

[4] Ballet, G., Borghesi, R., Hoffmann, P., Lévy, F. « Studio online 3.0: An internet" killer application" for remote access to ircam sounds and processing tools », *Actes des Journées d?Informatique Musicale*, Paris,1999.

[5] Bengio, Y., Yao, L., Alain, G., Vincent, P. « Generalized denoising auto-encoders as generative models », *Advances in neural information processing systems 26*, Burges, C. J. C., Bottou, L., Welling, M., Ghahramani, Z., Weinberger, K. Q. (dir.) Curran Associates, Red Hook, 2013, p. 899-907.

[6] Bishop, C. M. *Pattern Recognition and Machine Learning*, 2006, Springer-Verlag, New York.

[7] Chemla-Romeu-Santos, A. *Manifold representations of musical signals and generative spaces*, thèse de doctorat, sous la dir. de Assayag, G., Esling, P., Università degli Studi di Milano – Sorbonne Université, Milan, Italie, 2020.

[8] Chemla-Romeu-Santos, A., Ntalampiras, S., Esling, P., Haus, G., Assayag, G. « Cross-modal variational inference for bijective signal-symbol translation », *Proceedings of the International Conference on Digital Audio Effects*, 2019, Birmingham, United Kingdom.

[9] Chowning, J. M. « The synthesis of complex audio spectra by means of frequency modulation », *Journal of the Audio Engineering Society* 21/7 (1973), p. 526-534.

[10] Engel, J., Hantrakul, L. H., Gu, C., Roberts, A. « Ddsp: Differentiable digital signal processing », *Proceedings of the International Conference on Learning Representations*, Virtual, 2020.

[11] Esling, P., Chemla-Romeu-Santos, A., Bitton, A. « Generative timbre spaces with variational audio synthesis », *Proceedings of the International Conference on Digital Audio Effects*, 2018, Aveiro, Portugal.

[12] Flanagan, J. L. « Phase vocoder speech synthesis system », 1976, US Patent 3,982,070.

[13] Godsill, S. J. « Bayesian enhancement of speech and audio signals which can be modelled as arma processes », *International Statistical Review* 65/1 (1997), p. 1-21.

[14] Higgins, I., Matthey, L., Pal, A., Burgess, C., Glorot, X., Botvinick, M., Mohamed, S., Lerchner, A. « beta-vae: Learning basic visual concepts with a constrained variational framework », *Proceedings of the International Conference on Learning Representations*, Toulon, 2017.

[15] Higgins, I., Sonnerat, N., Matthey, L., Pal, A., Burgess, C., Botvinick, M., Hassabis, D., Lerchner, A. « Scan: Learning abstract hierarchical compositional visual concepts », *CoRR* abs/1707.03389 (2017). `http://arxiv.org/abs/1707.03389`, accédé le 21 octobre 2020.

[16] Hoffman, M. D., Blei, D. M., Wang, C., Paisley, J. « Stochastic variational inference », *The Journal of Machine Learning Research* 14/1 (2013), p. 1303-1347.

[17] Hoffman, M. D., Johnson, M. J. « Elbo surgery: yet another way to carve up the variational evidence lower bound », *Workshop in Advances in Approximate Bayesian Inference, NIPS*, 2016.

[18] Kingma, D. P., Mohamed, S., Rezende, D. J., Welling, M. « Semi-supervised learning with deep generative models », *Advances in Neural Information Processing Systems 27*, édité parZ. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, K. Q. Weinberger, Curran Associates, Inc., p. 3581-3589, 2014.

[19] Kingma, D. P., Welling, M. « Auto-encoding variational bayes », *Proceedings of the International Conference on Learning Representations (ICLR)*, 2014.

[20] Kumar, K., Kumar, R., de Boissiere, T., Gestin, L., Teoh, W. Z., Sotelo, J., de Brébisson, A., Bengio, Y. , Courville, A. C. « Melgan: Generative adversarial networks for conditional waveform synthesis », *Advances in Neural Information Processing Systems 32*, édité parH. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, R. Garnett, Curran Associates, p. 14 910-14 921, 2019.

[21] Li, Y., Turner, R. E. « Rényi divergence variational inference », *Advances in Neural Information Processing Systems*, p. 1073-1081, 2016.

[22] Makhzani, A., Shlens, J., Jaitly, N., Goodfellow, I., Frey, B. « Adversarial autoencoders », *arXiv preprint arXiv :1511.05644*, 2015.

[23] Mehri, S., Kumar, K., Gulrajani, I., Kumar, R., Jain, S., Sotelo, J., Courville, A., Bengio, Y. « Samplernn: An unconditional end-to-end neural audio generation model », *arXiv preprint arXiv :1612.07837*.

[24] Muller, M., Ellis, D. P., Klapuri, A., Richard, G. « Signal processing for music analysis », *IEEE Journal of Selected Topics in Signal Processing* 5/6 (2011), p. 1088-1110.

[25] Oord, A. v. d., Dieleman, S., Zen, H., Simonyan, K., Vinyals, O., Graves, A. , Kalchbrenner, N., Senior, A., Kavukcuoglu, K. « Wavenet: A generative model for raw audio », *arXiv preprint* arXiv :1609.03499 (2016).

[26] Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, et coll.. « Pytorch: An imperative style, high-performance deep learning library », *Advances in Neural Information Processing Systems* 2019, p. 8024-8035.

[27] Peeters, G. « Time variable tempo detection and beat marking. », *ICMC*, 2005, Citeseer, p. 539-542.

[28] Perraudin, N., Balazs, P. , Søndergaard, P. L. « A fast Griffin-Lim algorithm », *2013 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, IEEE, 2013, p. 1-4.

[29] Roads, C.« Introduction to granular synthesis », *Computer Music Journal* 12/2 (1988), p. 11-13.

[30] Schnell, N., Röbel, A., Schwarz, D., Peeters, G., Borghesi, R. et coll.. « Mubu and friends-assembling tools for content based real-time interactive audio processing in max/msp », *ICMC* 2009,.

[31] Scurto, H., Chemla, A. et coll.. « Machine learning for computer music multidisciplinary research: A practical case study », *14th International Symposium on Computer Music Multidisciplinary Research (CMMR'19)* 2019.

[32] Sturm, B. L. « A survey of evaluation in music genre recognition », *International Workshop on Adaptive Multimedia Retrieval*, 2012, Springer, p. 29-66.

[33] Theis, L., v. d. Oord, A., Bethge, M. « A note on the evaluation of generative models », *arXiv preprint arXiv :1511.01844* 2015.

[34] Tolstikhin, I., Bousquet, O., Gelly, S., Schoelkopf, B. « Wasserstein Auto-Encoders », 2017.

[35] Typke, R., Wiering, F. , Veltkamp, R. C. « A survey of music information retrieval systems », *Proc. 6th International Conference on Music Information Retrieval*, Queen Mary, University of London, p. 153-160, 2005.

Texte édité par Corentin Guichaoua.