

物理ベースサウンドデザインのための例示ベースインタフェース

山本 和彦^{*†} 五十嵐 健夫^{*}

概要. 3DCG コンテンツではインタラクティブ性への要求が高まるにつれて、光のレンダリングのみならず、物体同士が衝突したときなどの音のレンダリングについても物理ベースで計算をおこなうことへの需要が高まっている。しかし、物理ベースレンダリングで音を生成する場合、音色を決定するのは材質という物理的なパラメータであるため、アーティストが音を自在にデザインすることができないという問題がある。そこで本稿では、この問題を解決するために例示ベースの物理ベースサウンドデザインインタフェースを提案する。本システムでは、ユーザは、音をデザインしたい 3D モデルに対して、好みの音波形ファイルをドラッグ&ドロップで割り当てていく。それらを入力として、システムは、モデル内部の材質分布を最適化し、割り当てられた音に近い音が出る 3D モデルを出力する。さらに、この最適化には、従来計算コストの非常に高い振動解析が必要だったが、機械学習によるメッシュ簡略化と、階層的メッシュ分解を利用した新しいアルゴリズムによって、インタラクティブな速度での計算を実現している。これにより、ユーザは従来の非物理ベースの音デザインと全く同じように音そのものの編集作業のみに注力することができる。

1 はじめに

3DCG コンテンツでは、VR 等の普及によりそのインタラクティブ性への要求が高まるにつれて、光のレンダリングのみならず、物体同士が衝突したときなどの音のレンダリングについても物理ベースで計算をおこなうことへの需要が高まっている [2]。本稿では、こうした物理ベースで 3DCG 内空間の音をシミュレーションして生成する手法のことを物理ベースサウンドレンダリングと呼ぶ。単純に音波形ファイルを再生する従来法に対して、物理ベースサウンドレンダリングには、ユーザの様々なインタラクションに応じて、デザイナーが予めその全てを設定しておかなくても自然な音色変化を伴った音の反応を返すことができるというメリットがある。この物理ベースサウンドレンダリングの音色をアーティストが自在にデザインできるようなインタフェースを提案することが本研究の目的である。

物理ベースサウンドレンダリングの手法にはいくつかの種類があるが、もっともよく使われているのはモーダル合成 [1] と呼ばれる手法である。本稿でもこのモーダル合成を対象とする。モーダル合成では、3D モデルの形状とその内部の材質分布に依存する固有振動パターンをあらかじめ前計算しておき、ランタイム時には剛体シミュレーションやユーザからのインタラクションで得られた衝突情報に応じた固有振動パターン（モード）を音として鳴らす。この周波数、音量、減衰パラメータ（モーダルパラメータ）の情報を含む固有振動パターンが音色となる。

実際のコンテンツ制作においては多くの場合、3D

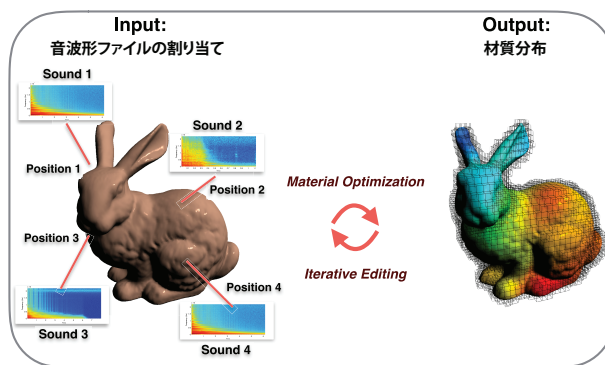


図 1. ユーザは 3D モデル表面のいくつかの場所にサウンドファイルを割り当てる。システムは割り当てられた音に近い音が物理ベースサウンドレンダリングで再現できるようにモデル内部の材質分布を最適化する。

モデルの形状が最初に与えられ、サウンドアーティストはそれに付随する音をデザインしていくことになる。ここで、3D モデルの形状が決まっているとき、モーダル合成の音色をデザインするための入力パラメータはモデル内部の材質分布という物理的なパラメータとなる。しかし、アーティストに限らず、どんな材質分布を入力してシミュレーションをおこなえばどんな音色が得られるのか、を予測することは我々には困難である。このため現状、物理ベースサウンドレンダリングの音色を実際のコンテンツ制作現場でデザインして利用することは現実的では無い。

そこで本稿では、この問題を解決するために、例示ベースの物理ベースサウンドデザインインタフェースを提案する (図 1)。本システムでは、ユーザは、音をデザインしたい 3D モデルの表面に対して、好み

Copyright is held by the author(s).

^{*} 東京大学

[†] ヤマハ株式会社

の音ファイルをドラッグ&ドロップで割り当てていく操作をおこなう。その入力に対して、システムは、モデル内部の材質分布を自動的に最適化して、割り当てられた音に近い音が出るような 3D モデルを出力する。さらに、この最適化には、従来計算コストの非常に高い振動解析が必要だったが、機械学習によるメッシュ簡略化と、階層的メッシュ分解を利用した新しいアルゴリズムによって、1000 倍以上の高速化を達成し、インタラクティブな速度での計算を実現している。本システムを利用することによって、ユーザは従来の非物理ベースの音デザインと全く同じように、物理的なパラメータやシミュレーションの待ち時間のことを一切考える必要がなく、音色そのものの編集作業のみに注力することができるようになる。

本稿は、国際学会で既に発表済みの内容 [6] を簡潔にまとめたものである

2 関連研究

モーダル合成によって得られる音色のデザインを例示ベースでおこなうことを目的とした既存手法には大きく分けて 2 種類ある。1 つは、実際に録音された複数のサウンドクリップを使う方法、もう 1 つは、サウンドファイル一つのみを使う方法である。

多数のサウンドクリップを使う手法として、Pai ら [9][7] と Corbett ら [13] は、音をデザインしたい 3D モデルと全く同じ形状の現実のものを用意し、それを回転台の上に置いて表面の様々な位置をロボットアクチュエータで叩いて、その音を大量に録音することで、モデルの固有振動パターン自体を実測し補間して、3DCG 空間内でそれを再現している。しかし、これらの手法では、現実の物体を用意しなくてはならないのに加え、大量の測定作業とサウンドファイルの編集作業が必要なため、あまり実用的とは言えない。さらに、彼らの手法では測定の労力を削減するために、少数のサウンドクリップを複数の表面位置で使い回すことも可能ではあるが、その場合、例えば 3DCG 空間内で物体の異なる場所を触っても全く同じ音出力されてしまうため、物体とのインタラクションが不自然になってしまうという問題がある。

大量の録音作業を避けるために、サウンドファイル 1 つのみを利用してモーダル合成の音色デザインを試みたものとして、Lloyed ら [5] はまず、サウンドファイルからモーダルパラメータ (振幅, 周波数, 減衰) を抽出し、それを物体が他のものに衝突するたびにランダムに駆動して鳴らしている。しかし、この手法では同じ物体の同じ場所が同じように衝突してもランダムに異なる音が鳴ってしまうなど、不自然な結果を生んでしまうという問題がある。

Ren ら [16] は、サウンドファイル 1 つのみを使ってそこからモーダルパラメータを推定するまでは

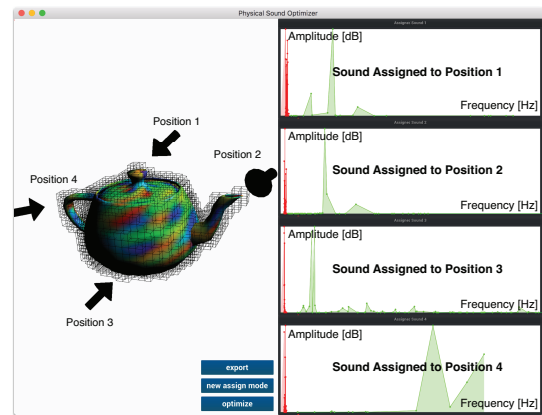


図 2. システム UI 画面. 左の画面でユーザは音の確認と割り当てを行う. 右の画面には、割り当てられた音とシミュレーション音のパワースペクトルがそれぞれ示されている。

Lloyed らと同じだが、ランダムにそれらを駆動するのではなく、そこから材質パラメータ (モーダルパラメータではなく材質固有の減衰係数) 自体を推定することで、物理的に自然な音のシミュレーションをおこなった。しかし、彼らの手法では、シミュレーションしたいモデルとほぼ同じ形状と材質の現実物体を用意して録音作業をおこなわなければならないという問題がある。

3 ユーザインタフェース

図 2 が本システムの GUI である。本システムでは、ユーザはまず、音をデザインしたい 3D モデルをシステムに与える。この 3D モデルはボリューメトリックメッシュである必要はなく、サーフェースメッシュでよい。システムは自動的に入力されたメッシュをボクセライズし、正 6 面体有限要素メッシュに変換する。モデル内部の材質分布はランダムに初期化される。ユーザは現在の材質分布で物理ベースサウンドレンダリングをした場合、どんな音が出るのかを、モデル表面をマウスクリックして叩くことによって確認できる。

次にユーザは、図 3 のように音を割り当てたいモデルの表面位置をマウスで選択し、さらに任意のサウンドファイルをドラッグ&ドロップして割り当てる。このサウンドファイルは、割り当てたモデル表面位置を叩いたときに鳴ってほしい音のものを用意する。この音は、自然物体を叩いたときの音のような減衰音である (オルガンや管楽器音のような持続音でない) 必要があり、インパルス応答のような音が理想的である。また、自然音である必要はなく、シンセサイザーなどで合成した音でもよい。割り当ては一箇所でも複数箇所でもよく、物体の異なる場所にそれぞれ異なる音色を与えることができる。

任意の箇所にサウンドファイルを割り当てたのち



図 3. サウンドファイルの割り当て.

Optimize ボタンを押すと、システムは、割り当てられたそれぞれの位置を叩いたときにそれぞれのサウンドファイルに近い音が鳴るように、モデル内部の材質分布（具体的には各有限要素のヤング率）を最適化する（図 4）。複数箇所にも音が割り当てられていた場合、それらの中間地点では物理的に自然に音がつながるように材質分布の設定がおこなわれる。

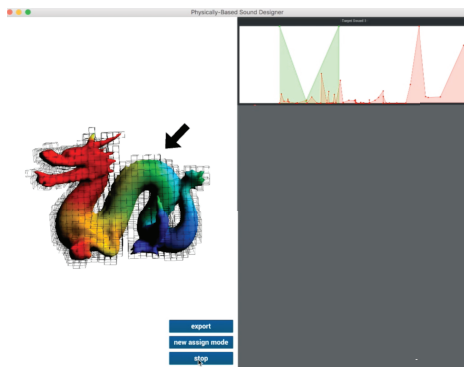


図 4. 材質分布の最適化. モデル表面の色はヤング率の相対値を可視化している.

ここで、密度とポアソン比は一様に設定される。最適化は反復的におこなわれ、モデルの大きさに依存して 1 ステップ 0.5 ~ 3.0 秒ほどかかる。システムは、1 ステップ最適化が進むごとに、表面付近のヤング率の変化を色で可視化する。ユーザは最適化の途中であっても現在の材質分布での音を確認することができ、任意のタイミングで最適化を中断する

ことができる。加えて、最適化を中断してさらに新しい音の割り当てをおこない再び最適化を開始することもでき、ユーザは満足いくまで物理ベースサウンドレンダリングの音を作り込んでいくことができる。最後に、出力ボタンを押すとシステムは固有振動パターンを含むメッシュデータを出力する。この出力データは、既存の物理ベースサウンドレンダリングのパイプラインでそのまま利用することができる。

4 アルゴリズム

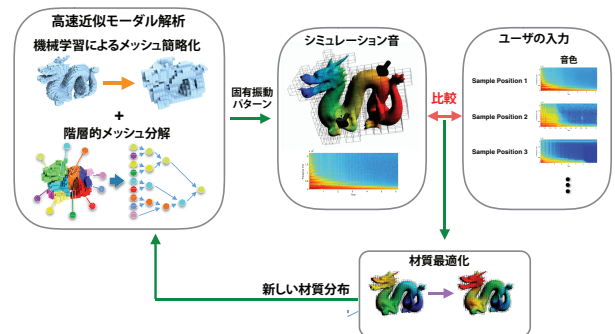


図 5. アルゴリズム全体像.

図 5 が本システムのアルゴリズムの全体像である。ユーザがモデルの T 箇所にそれぞれサウンドクリップの割り当てを行った場合、システムはまずそれらのサウンドクリップからモーダルパラメータ（各モードの周波数、音量、減衰係数）を Ren らの手法 [16] で抽出する。抽出されたモードの数が合計 N 個であったならば、システムはモデルの剛体モード 6 つを差し引いた低次のモード N 個の周波数と音量バランスが、割り当てられた T 箇所それぞれにおいて、割り当てられた音にマッチするように、モデル内部の材質（各有限要素のヤング率）最適化をおこなう。これは非線形非凸最適化問題となるので、最適化には CMA-ES[10] と準ニュートン法のハイブリッド法 [4] を使った。ここで本システムでは、モードの減衰パラメータに関しては最適化をおこなわず、サウンドクリップから抽出されたパラメータをそのまま使う。

4.1 高速近似モーダル解析

本システムでは、最適化の 1 ステップごとに必ず一回のモーダル解析（固有振動解析）をおこなう必要がある。モーダル解析では、3D モデルの形状と材質分布を入力として、固有振動パターンを計算する。この計算には、有限要素法で巨大な行列の一般化固有値問題を解く必要がある。この計算は非常に計算コストが高く、数千要素程度のモデルでも通常数時間かかる処理であるため、本システムのように何度も反復的に計算しなくてはならない問題にそのまま

使うことは現実的ではない。そこで本稿ではこの問題を解決するために、機械学習によるメッシュ簡略化と、階層的メッシュ分解と効率的なエラー訂正手法、という2つの異なる手法を組み合わせ、インタラクティブな速度でメッシュの一般化固有値問題を計算することができる手法を提案する。提案手法では図5左上のようにまず、入力された有限要素メッシュを機械学習で精度を落とさないように粗くしたのち、階層的に細かく分解して自由度を削減しつつマージし直していき、最後にエラー訂正をおこなうことで一般化固有値問題を近似的に高速に解く。本提案手法に関して、本稿では概要の説明のみにとどめる。手法の詳細に関しては山本ら [6] を参照されたい。提案手法による音のシミュレーション結果を図6に示す。この結果では、従来手法と提案手法とで同じ剛体シミュレーション情報を使って音を生成したときのスペクトログラムを比較している。2つのスペクトログラムは非常に似通っており、1000倍以上の高速化を実現しつつ、正解に非常に近い結果が得られていることがわかる。

4.1.1 機械学習によるメッシュ簡略化

まずシステムは、入力された有限要素メッシュを機械学習を使うことによって、精度を保ったまま粗くする。我々の手法はChenらのData-Driven FEM [3] を拡張している。Data-Driven FEMでは、隣り合う8つの正六面体要素（それぞれ別々のヤング率をもつ）の変形を精度良く近似できるような1つの正六面体要素のヤング率（正確には1要素内の8つの積分点それぞれでのヤング率）を求める（2次元の場合には隣り合う4要素を近似できる1要素のヤング率）。これを実現するため、様々なヤング率をもつ8要素からなる立方体を用意して、それらに様々な方向から力を加えてシミュレーションをおこない、その変形を1つの正六面体要素で近似できるように機械学習する。

しかし、ChenらのData-Driven FEMでは、単純に8つの材質を入力としてデータベースから該当する結果を一对一で検索しているだけなため、汎化性能が全くないという問題がある。そこで我々は、広い範囲の連続値をとるヤング率の入力を高速に扱うことができるように（データベースに含まれない入力にも対応できるように）、Regression Forests [12] を拡張して使うことで、この問題を解決した。

4.1.2 階層的メッシュ分解とエラー訂正

次に粗くなったメッシュを階層的メッシュ分解をおこなうことによって、さらに高速に一般化固有値問題を解く。我々の手法はComponent Mode Synthesis (CMS) [15] の拡張である。まず、メッシュをWangら [11] の手法を使って細かく分解し、分解されたそれぞれのパーツだけで一般化固有値問題を解く。分

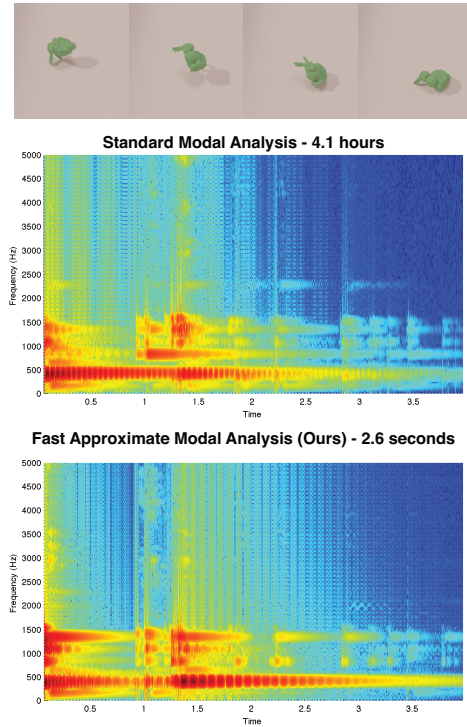


図6. 従来手法（上）と高速近似モーダル解析（下）との比較。従来手法で4.1時間かかった計算を近似的に2.6秒で計算している。

解されたパーツの自由度は非常に小さいため、この計算は高速におこなうことができる。次に、自由度を削減しつつ、隣り合うパーツを階層的にマージしてさらに小さな一般化固有値問題を解く、ということを全てのパーツが繋がるまで繰り返す。これらの処理は並列化して効率的におこなうことができる。最後に全てマージされたモデルの一般化固有値問題の計算結果のエラー（分解とマージを繰り返すことによって生じたエラー）をSubspace Iteration Method [8] で訂正する。我々はSubspace Iteration MethodにさらにOverrelaxation Weightsとreduced mass Modified Gram-Schmidt process [14]を導入することで高速なエラー訂正を実現した。

5 結果

本システムを用いて、実際に物理ベースサウンドレンダリングで得られる音色のデザインをおこなった。本章のすべての実験において、モデル内部のポアソン比は一律に0.25、密度は 1.0 kg/m^3 に設定した。使用したPCのスペックは、CPU: Intel Core i7 2.6 GHz, RAM: 16GB, GPU: NVIDIA GeForce GT 750Mである。

まず、想定される基本的なサウンドデザインケースとして、図7のようなフライパンの3Dモデルに対して音の割り当てをおこなった。フライパンは一般的に取っ手と皿の部分で材質が異なるため、取っ

手の部分には木材を叩いた音、皿の部分には鉄の皿を叩いた音をそれぞれ図7最上段の Position 1 と 2 の場所に割り当てて最適化をおこなった。割り当てられた2つの音から抽出されたモードの数は合計で30個となり、モデルの低次30モードの周波数と音量バランスが最適化対象となった。Position 1, 2それぞれの場所を叩いたときの音の最適化前、最適化後ののスペクトログラムをそれぞれ図7中段に示す。明らかにそれぞれの最適化後のスペクトログラムは割り当てられた音に近づいているのがわかる。また、ユーザによって何も音が割り当てられていない中間点 (Position 3) を叩いたときの音は、割り当てられた2つの音色の中間の音色で、かつ物理的に自然に補間されていることがわかる (図7最下段)。これは提案手法の大きなメリットの一つであり、ユーザに大量の音の割り当てを要求することないためデザインの負担が少なく済む。実際、FoleyAutomatic [7] でもこの例に似たフライパンのモデルへの音の割り当てをおこなっているが、彼らの手法では皿の部分だけでも5つの音を割り当てを必要としているのに対し、提案手法では一箇所のみの割り当てで済んでいる。

次に、ユタティーポットのモデル (図2) を使用して逐次的な物理ベースサウンドのデザイン作業を試みた。ユーザはまず、ティーポットの本体に金属音を割り当て最適化し、ある程度最適化が進んだところで、最適化を中断し、音のチェックをおこなった。さらにユーザはモデルの衝突箇所による音色変化の幅を増やすために蓋や、注ぎ口の部分に別の金属音を割り当て、出音を確認しつつ何度か最適化の再開/中断を繰り返してこのモデルの音をデザインしていった。このように、インタラクティブに出音を確認しながら物理ベースサウンドの音を逐次デザインしていくワークフローは従来に無く、提案手法によって実現されたものである。

従来手法に対する提案手法のメリットの一つとして、現実には存在しないような音の割り当てをおこなうことが可能である。付属の動画に収録した例では、Standord Bunny のモデルに対し、4つの異なる場所に4種類の音高のピアノの音 (C4, E4, G4, F3) を割り当てた。このような音を発する物体は現実には存在しないものの、システムは物理的に自然に割り当てられた音と場所を補間するように材質分布を最適化している。

最後に、多くの種類のオブジェクトが登場する大規模シーンのサウンドデザインをおこなった (図8)。このシーンでは、登場する全ての音を本システムでデザインし、物理シミュレーションエンジンの衝突イベントのみをトリガーとして発音している。作業時間は、物理シミュレーションエンジン (アニメーション) の設定に一週間、画像のレンダリングに3日間を要したが、音そのものをデザインするのにか

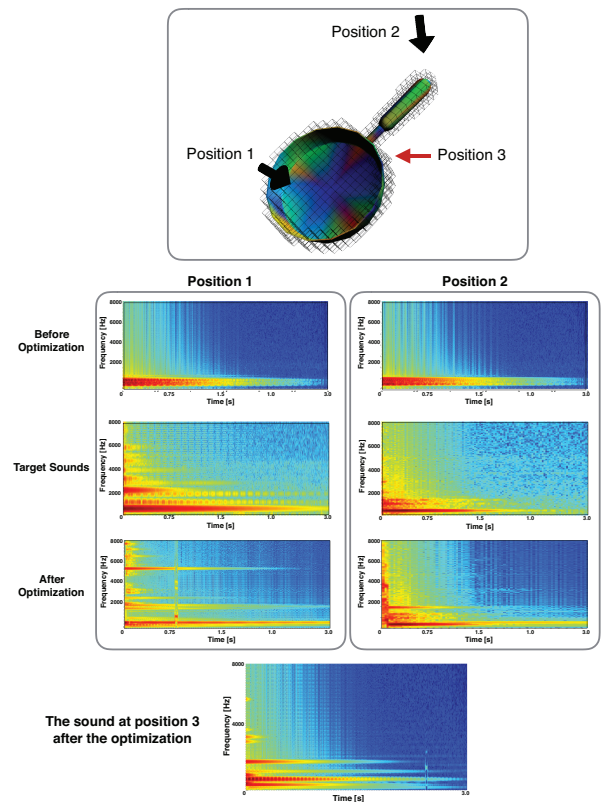


図7. フライパンモデルへの音の割り当て。Position 1, 2の位置にそれぞれ音を割り当てた。

かった時間はトータルで30分ほどであった。この作業速度で物理ベースサウンドのデザインをおこなうことは従来手法では困難であったものであり、提案手法の大きなメリットの一つである。

6 リミテーション

提案手法では3DCG空間内でのシミュレーションを対象にしている都合上、モデル内部の材質分布を連続最適化しており、その結果現実には存在しない物性の組み合わせが出力されてしまうことがある。このためファブリーケーションのように実際のものの音をデザインする用途にはそのまま使うことができないという問題がある。現実的な材質の組み合わせ最適化を採用することによってこれを実現することも考えられるが、様々な材質を細かく組み合わせ形状を出力することは現在の加工機では困難であり、今後の課題である。

また、本稿では、材質パラメータとして各有限要素のヤング率のみを最適化したが、他にも密度とポアソン比を考慮することは今後の課題である。密度とポアソン比を考慮することで、より最適化によって再現できる音色のクオリティが向上する。さらに、提案手法では物体の振動のみを最適化しており、物体表面からの音放射の影響を考慮していないという

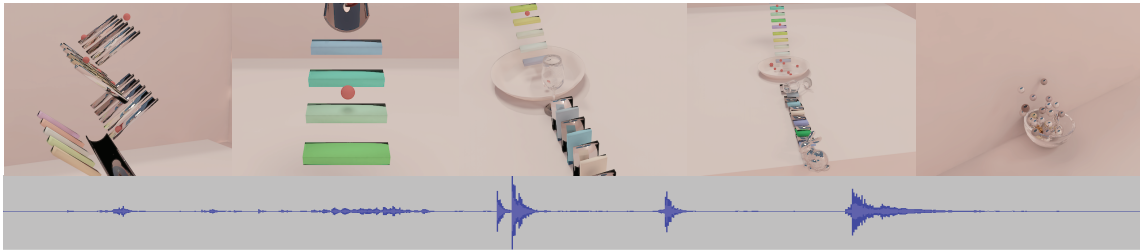


図 8. 多数のオブジェクトが登場するシーン.

制限がある. この音放射は音色に少なからず影響を与えるため, 音放射も含めた最適化の手法を提案することは今後の課題である.

7 まとめ

本稿では, 物理ベースサウンドレンダリングによって得られる音色を, アーティストでも自在にデザインできるようにするための, 例示ベースのインタフェースを提案した. さらに, 機械学習によるメッシュ簡略化と, 階層的メッシュ分解によって, インタラクティブな速度での振動解析の計算を実現し, ユーザへのシミュレーション待ち時間の負担を大幅に軽減した. これによって, アーティストは物理的なパラメータ設定やシミュレーションのことを一切考える必要がなく, 音そのもののデザイン作業に注力できるようになった.

参考文献

- [1] Aean-Marie Adrien, The missing link: Modal synthesis. In *Representations of Musical Signals*, pp. 269–298, MIT Press Cambridge, 1991.
- [2] Changxi Zheng and Doug L. James, Toward high-quality modal contact sound, p.38, *ACM Trans. on Graphics (SIGGRAPH)*, 2011.
- [3] Chen, Desai, Levin, David I. W., Sueda, Shinjiro and Matusik Wojciec, Data-driven Finite Elements for Geometry and Material Design, Vol.34-4, pp.74:1–74:10, *ACM Trans. on Graphics (SIGGRAPH)*, 2015.
- [4] Chen, Xuefeng, Liu, Xiabi and Jia, Yunde, Combining Evolution Strategy and Gradient Descent Method for Discriminative Learning of Bayesian Classifiers, pp. 507–514, *Proceedings of the 11th Annual Conference on Genetic and Evolutionary Computation*, 2009.
- [5] D. Brandon Lloyd, Nikunj Raghuvanshi and Naga K. Govindaraju, Sound synthesis for impact sounds in video games, *ACM In Symposium on Interactive 3D Graphics and Gam*, 2011.
- [6] Kazuhiko Yamamoto and Takeo Igarashi, Interactive Physically-Based Sound Design of 3D Model using Material Optimization, *ACM SIGGRAPH / Eurographics Symposium on Computer Animation (SCA)*, 2016
- [7] Kees van den Doel, Paul G. Kry and Dinesh K. Pai, FoleyAutomatic: physically-based sound effects for interactive simulation and animation, pp. 537–544, In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques (SIGGRAPH)*, 2001.
- [8] Klaus-Jurgen Bathe, The subspace iteration method - Revisited, pp. 977–983, *Computers and Structures*, 2013.
- [9] Dinesh K. Pai, Kees van den Doel, Doug L. James, Jochen Lang, John E. Lloyd and Joshua L. Richmond and Som H. Yau, Scanning physical interaction behavior of 3D objects, pp. 87–96, *Proceedings of the 28th annual conference on Computer graphics and interactive techniques (SIGGRAPH)*, 2001.
- [10] Hansen, N, Muller, SD and Koumoutsakos, P, Reducing the time complexity of the derandomized evolution strategy with covariance matrix adaptation (CMA-ES), pp. 1–18, *Evolutionary Computation*, 2003.
- [11] Hao Wang, Tong Lu, Oscar Kin-Chung Au and Chiew-Lan Tai, Spectral 3D mesh segmentation with a novel single segmentation field, pp. 440–456, *Graphical Models*, 2014.
- [12] Ladický, L'ubor, Jeong, SoHyeon, Solenthaler, Barbara, Pollefeys, Marc and Gross, Markus, Data-driven Fluid Simulations Using Regression Forests, pp.199:1–199:9, *ACM Trans. on Graphics (SIGGRAPH Asia)*, 2015.
- [13] Richard Corbett, Kees van den Doel, John E. Lloyd and Wolfgang Heidrich, Timbrefields: 3D Interactive Sound Models for Real-time Audio, pp. 643–654, *Presence: Teleoperators and Virtual Environments - Special section: Advances in interactive multimodal telepresent systems*, 2007.
- [14] Yang, Yin, Li, Dingzeyu, Xu, Weiwei, Tian, Yuan and Zheng, Changxi, Expediting Pre-computation for Reduced Deformable Simulation, pp.243:1–243:13, *ACM Trans. on Graphics (SIGGRAPH Asia)*, 2015.
- [15] W. C. Hurty, Dynamic analysis of structural systems using component modes, pp.678–685, *AIAA Journal*, 1965.
- [16] Zhimin Ren, Hengchin Yeh and Ming C. Lin, Example-guided physically based modal sound synthesis, pp. 987–995, *ACM Transactions on Graphics (TOG)*, 2013.

採録区分：ロング採録

判断理由：

提案されている 3D モデル内部の材質分布を複数の音色を割り当てることで最適化する手法は、3DCG コンテンツ作成において作業負荷を軽減する手法として大いに期待できます。ビデオでは提案理論によるプロトタイプの実現可能性が示されており、提案手法の有効性も確認できます。また、提案手法のリミテーションや将来展望についてもしっかりと言及されており、ロング採録にふさわしいと判断されました。

レビューサマリ：

全体の構成：

本論文は国際学会で既に発表済みの内容をまとめたものです。好みの音ファイルをドラッグ&ドロップで 3D モデルの表面に割り当て、インタラクティブに出音を確認しながら逐次デザインしていくという手法は、既存の音色デザイン手法には無く、特に複数の音ファイルを割り当て可能な点など、新規性および有用性が高いと考えられます。

改良に向けたコメント等：

- ・衝突時の音は何で叩くか、どこにぶつかるかなど、相手の形状や材質にも依存すると思われますが、それらの表現分けは可能でしょうか？ もしくは必要でしょうか？ Lloyed らの手法はそのあたりをランダムに変化させることでうまくごまかしていますが、何で叩いてもどこに落ちてても同じ音だと不自然に感じることはないかという点が気になります。自然な音とは何か、ということの議論と評価があってもよいのではと思います。今後の課題としてあげられていた、密度とポアソン比を考慮する必要があるのか、効果がどれほどあるのかの検討材料にもなります。

- ・Ren らの手法に関する説明で、「シミュレーションしたいモデルとほぼ同じ形状と材質の現実物体を用意して録音作業を行わなければならない」とありますが、少なくとも形状に関しては全然異なるものに適応しているので正確な比較ではないと思われます。

- ・LIMITATION にも記載がありますが、材質が滑らかに変化するような計算結果がでたとして、それを物理的に実現することは難しいです。実物を作る方法がないとしてしまうと、この技術の有用性に疑問が生じます。例えば、少数（複数の）パーツ（ただし、パーツごとには均一の素材）のようなものならば、物理的な実現がより容易になると思われます。

- ・軽微な編集誤りがあります：

- p.5 要求することないため → 要求することがないため

本論文に対する各査読者の詳しいコメントは以下のページを参照のこと：

<http://www.wiss.org/WISS2016Proceedings/oral/10.html>

***本ページは論文本体ではありません**

【物理ベースサウンドデザインのための例示ベースインタフェース】