

Gravity Pianist: 楽器の解体・再配置による音楽表現環境

矢田 裕基^{†1} 森本 浩尉^{†1} 宮下 芳明^{†1}

本稿では、物理法則が支配する空間にオブジェクト(ピアノの鍵盤)を配置し、衝突によって発音させる音楽表現環境を提案する。作曲を行ったり、リアルタイムで演奏することができるほか、意図したタイミングでの発音をはかるための先読み機能、出来上がった音楽を配信する動画出力機能、作り上げた系を新たな振奏楽器として持ち出す機能を実装した。

Gravity Pianist: An Environment for Musical Expression by Remapping Piano Keys

HIROKI YATA,^{†1} HIROYASU MORIMOTO^{†1}
and HOMEI MIYASHITA^{†1}

In this paper, we propose a musical expression environment where the user can remap piano keys in space so that they make sounds by object collision. In addition to basic functions for composition and performance, we made a look-ahead function to support arrangement in space, upload function of the movie, and the export function for iPod, that makes it maraca.

1. はじめに

近年、音楽と視覚的要素が密接に関係している楽器やコンテンツが人気を博している。例えば、TENORI-ON¹⁾による演奏は、聴こえる音以外にも、譜面や音に対する効果をLEDの発光で見せることで、演奏を耳と目で同時に楽しむことができる。他にも、ピアノロール

に絵を描くMIDIアニメ²⁾や、操作しなくてもクリアできるように改造しかつ効果音で演奏を行う全自動マリオ³⁾などの動画がある。これらの動画は、ニコニコ動画やYouTubeなどの動画共有サイトで人気のジャンルとなっている。これらのコンテンツが人気となる要因として、一つは作曲・演奏の過程と出力される音と操作の関係が視覚的に見えるため、鑑賞者にとっても楽しいコンテンツになっていることが挙げられる。また、動画を見れば、その作品の技術力の高さが一目でわかるということも要因の一つである。音楽に同期した作品を構成するのは労力を要するが、それゆえに意図通りに演奏されたときには人を感動させる。

一方、計算機の性能向上にともない、最近は物理演算を使ったインタラクティブなコンテンツが増えてきた。例として、ソニー・コンピュータエンタテインメントが発売した物理演算を利用したゲームであるリトルビッグプラネット⁴⁾、学習のための物理演算ソフトであるPhun⁵⁾やBox2DFlashAS3⁶⁾を利用したFlashコンテンツなどがある。これらのコンテンツは、直感的に物体を自由に配置したり動かしたりすることができ、慣れてくると複雑な動きをする物体も作成することができる。これらはNHKのテレビ番組「ピタゴラスイッチ」⁷⁾のような見ているだけで楽しい仕掛けを構築するにとどまらず、独自のゲームや8bitの計算機まで作ってしまう⁸⁾ことすら可能である。もちろん前述の作曲インタフェースの話と同様、意図通りの動作を行う系を構築するのは困難だが、その困難さゆえにうまく動作させられたものは喝采を得るし、それは視覚的に関係が見えやすいからこそであると考えられる。

そこで、筆者らは、音楽映像表現と物理演算システムの共通点をヒントに、物理演算を用いた作曲インタフェースであるGravity Piano⁹⁾を開発した。これは、ピアノという楽器をいったん解体し、それを物理法則が支配する空間に再配置することによって楽譜を構築するというシステムである。このシステムをデモ発表やインターネット上で公開し、多くの人に触れてもらった。システムを使った人の感想として、使っていて楽しい、新たな作品が作れそうなど概ね良好な感想を得ることができ、視覚的要素から音楽を作り出すことが製作者や鑑賞者にとって楽しいことであることを確認できた。また、タッチパネルや加速度センサを用いて操作を行いたい、既存の楽曲を演奏するのが難しい、作品を気軽に発表したいなどの多くの改善点も見つかった。

これによって得られた知見を基に、本稿では、システムをより発展させた物理演算の要素を取り入れた作曲・演奏環境システム「Gravity Pianist」を提案する。Gravity Pianistは、Gravity Pianoを基にいくつかの機能を加えた作曲・演奏環境である。主な追加機能としては、物理演算によるオブジェクトの動きを予測する機能がある。これにより、ユーザは、自身の意図に沿うタイミングで音を鳴らすことが容易になり、今までよりも楽に空間を設計で

^{†1} 明治大学理工学部情報科学科
Department of Computer Science, Meiji University

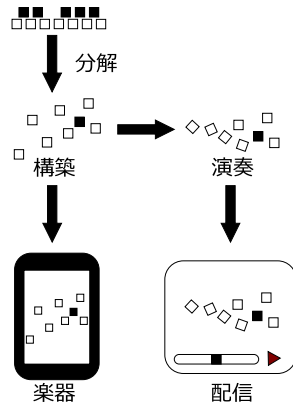


図 1 概要

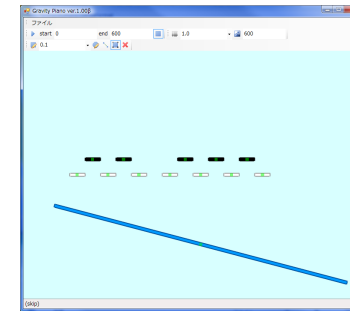


図 2 設計 View



図 3 編集ツールバー

きるようになった。他にも、複数の空間を設計・演奏することが出来るレイヤ機能、その設計した空間自体をひとつの振奏楽器として持ち出して特定少数規模での演奏パフォーマンスができる機能、さらに音楽と同期した動画として不特定多数に配信する機能をもたせた。

2. 関連研究

物理現象を作曲に取り入れた先行例としては、バネモデルで振動するオブジェクトを用いて演奏を行う SONASPHERE¹⁰⁾、一定周期で飛び出す魚が葉っぱにぶつくとそれに応じて音が鳴るエレクトロプランクトン¹¹⁾の「ハネンボウ」がある。また、江渡らが粒子の物理運動による音色合成の試み¹²⁾を行っている。一方、ゲームをモチーフとした演奏ツールとして藤岡らが Cubie¹³⁾ を提案している。他にも、空間を設計して音楽を奏でる試みとして、落書きやスタンプを使って音を奏でる事ができるスケッチピストン 4¹⁴⁾、弾性ボールの仮想バウンスを利用して音楽を奏でる試みとして、Hazumusic¹⁵⁾ がある。

本稿では構築プロセスがより直感的で、演奏プロセスがより明示的に見えるシステムを目標としている。

3. システム概要

本システムは、大きく分けて四つの役割がある (図 1)。鍵盤楽器を分解し再構築する「構築」、構築して出来た空間を物理演算によって動かし音楽を奏でる「演奏」、構築して出来

た空間を持ち運び特定少数に向けた演奏・パフォーマンスを行う「楽器」、演奏の映像を動画に変換し、インターネット上にアップロードして自分の作品を不特定多数に見てもらえる「配信」である。これらのうち「構築」「演奏」「配信」を Windows アプリケーションとして、「楽器」を iPhone アプリケーションとして実装を行った。

Windows アプリケーションは、設計 View と演奏 View という二つのモードに分かれている。設計 View は画面上にオブジェクトを配置するモードであり、ここでは、いわば演奏される「楽譜」を構築する作業を行う。これは、「構築」の役割に相当する。演奏 View は、設計 View で作ったオブジェクトを物理演算によって動かし音を鳴らすモードである。これは、「演奏」の役割に相当する。

3.1 設計 View

設計 View では、ユーザは四角形や円などのオブジェクトを画面上に自由に配置することができる。これらのうち、特定の音が割り当てられているのを「鍵盤オブジェクト」と呼ぶ。鍵盤オブジェクトは、後述の演奏 View で別のオブジェクトと衝突すると、割り当てられた音が鳴る。各オブジェクトについては、オブジェクトの質量・角度・反発係数やオブジェクトを空間に固定するなどの設定が行える (図 2)。

3.1.1 オブジェクト配置・編集

主な操作は、上部にあるツールバーとマウスを使って行う。まず、オブジェクトを配置するには、編集ツールバー (図 3) の左端にある長方形書き込みボタンをクリックすることで行



図 4 編集支援ツールバー

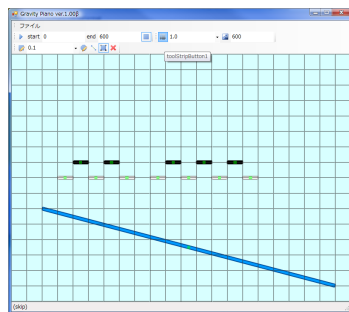


図 5 グリッド機能

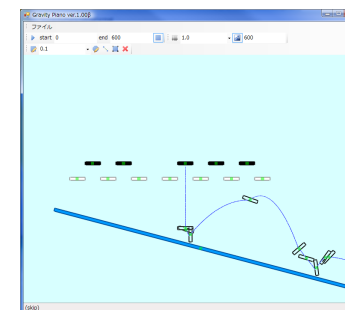


図 6 先読み機能

える。その後、中央部にある設計フィールドをドラッグアンドドロップすれば、棒状のオブジェクトが生成される。円形書き込みボタンをクリックすると、同じ手順で円形のオブジェクトが生成され、回転体書き込みボタンで、その場で回転するオブジェクトが生成される。

生成された各オブジェクトには予め質量・角度・反発係数などの物理演算に必要な値と音高や音色といった音の出力に必要な値が設定されている。これらの値は、編集ボタンをクリックすることで行える。これでオブジェクトの中央に表示されている点を囲むように選択すればオブジェクトの値を編集するダイアログが表示される。複数のオブジェクトに同じ値を同時に設定することも可能である。また、オブジェクトの削除は右端の削除ボタンで行え、前述の値の編集と同じように点を囲むと選択されたオブジェクトは削除される。視点の変更はマウス右ボタンで行う。マウス右ボタンで設計フィールドをドラッグアンドドロップすると、視点がマウスの方向に移動する。

3.1.2 オブジェクト編集支援機能

操作を支援する機能として、グリッド機能とレイヤ機能と先読み機能がある。グリッド機能とは、オブジェクトを格子の交点に吸い付くように補正する機能で、編集支援ツールバー(図4)の左端にあるグリッドボタンを選択することで利用することができる。グリッド機能を選択すると画面上に格子が表示され、オブジェクトの配置や移動の調整は、格子と重なるように補正される(図5)。格子の幅の調整は、グリッドボタン横のテキストボックスに数字を入力することで出来る。

レイヤ機能は、複数の空間を設計・演奏する機能である。キーボードのカーソルキーを押すと、次の空間へ移動したり、空間の新規作成・複製を行うことができる。各空間は独立しており、背景や重力などを個別に設定できる。これにより、より多くの空間を設計できる。また、好きなタイミングで次の空間へ切り替えることが可能となり、演奏の幅が広がる。

先読み機能は、指定のオブジェクトがどのような軌跡を辿りどこで衝突するのかを先読みする機能で、先読みボタンをクリックすることで利用することができる。先読み機能選択中にオブジェクトの中心部の点をドラッグアンドドロップで囲むと、オブジェクトの指定フレームまでの軌道と、指定フレームと衝突時のオブジェクトの輪郭を表示する(図6)。これにより、指定のフレームで音を鳴らすにはどの位置にオブジェクトを置けばいいかがわかりやすくなる。

3.1.3 その他支援機能

他の支援機能として、BPM 計算機能・ログ機能がある。これら機能は主に後述する計画的な設計のための機能である。

BPM 計算機能は、何フレーム間隔で音を出すと音符になるかの計算を予め計算して、オブジェクトの配置を支援してくれる機能である。BPM を入力すると、全音符・2分音符・4分音符・8分音符・符点8分音符・16分音符が何フレームかを表示する(図7)。これにより、ユーザは BPM に合わせての空間設計が行いやすくなる。

ログ機能は、前述の先読み機能と並行して使う機能で、先読みした結果を元に、どのフレームで衝突が発生し、前の衝突と何フレームの差があったのかを別ウィンドウに表示する(図8)。これにより、ユーザは、どのオブジェクトがどのタイミングで発音するかを正確に把握することが出来る。



図 7 BPM 計算機能

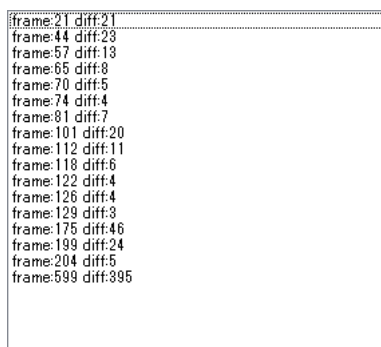


図 8 ログ機能

3.2 演奏 View

演奏 View に切り替えると、設計 View で配置したオブジェクトはそれぞれ物理演算によって動き出す。特にオブジェクトと鍵盤オブジェクトが接触した場合、割り当てられた音出力される。内部的には、接触した時にどの音を鳴らすかをスタックなどに保存しておき、物理演算による処理が終了したあとに音を鳴らしている (図 9)。

3.2.1 操作方法

演奏・設計ツールバー (図 10) の左端にある演奏ボタンを押すと、演奏 View に移行する。演奏 View に移行すると、オブジェクトはそれぞれ物理演算によって 1/60 秒間隔で動き出す。演奏ボタンの横にあるテキストボックスに開始フレームと終了フレームを設定できる。

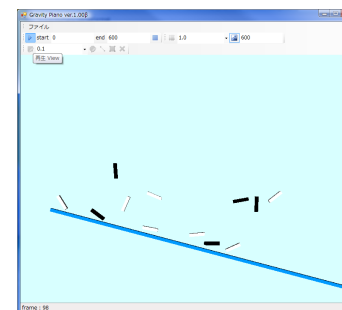


図 9 演奏 View



図 10 演奏・設計ツールバー

開始フレームとは、演奏開始時の秒数である。1 フレームは 1/60 秒であり、通常は 0 に設定されている。終了フレームとは、演奏終了時の秒数である。終了フレームまでいくと、次のレイヤを設定していれば、次のレイヤに切り替わる。次のレイヤを設定していなければ、同じレイヤを繰り返す。

演奏中には、動いているオブジェクトをマウスでドラッグすることでオブジェクトを手動で動かしたりと、空間に干渉することも可能である。これにより、いわば楽譜通りに演奏するのではなくユーザー側がアレンジを加えてアドリブで演奏することも可能となっている。

3.3 動画への出力

本システムでは、演奏 View での演奏過程を録画し、動画ファイルとして出力する機能を設けている。これにより、不特定多数が気軽に本システムで作られた作品を見ることが可能となる。そして、技術の競い合いが生まれ、さらに高い技術を用いた動画が生まれていくのではないかと期待している。

3.4 楽 器

設計 View で構築して出来た空間は、iPhone や iPod touch などのデバイスでも演奏することができる (図 11)。これにより、タッチパネルと加速度センサによる操作を行えるようになる。例えば、機材を傾け重力の操作を行ったり、オブジェクトを実際に指で触って移動したりすることにより、特定の少数に向けた演奏パフォーマンスが行える。これは、分解し再配置した鍵盤楽器を振奏楽器として使うというメタファである。

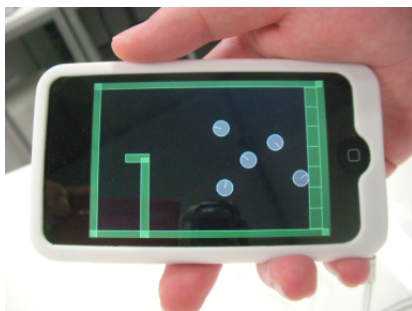


図 11 iPhone アプリケーション版

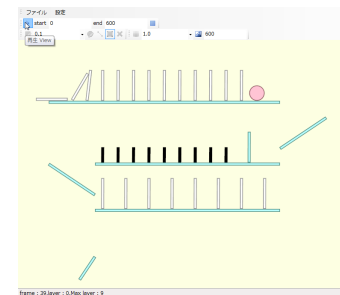


図 12 作例 : Domino Piano

4. 作品制作の流れ

本システムで作品を制作する手法は、大きく分けて二つに分かれている。一つは、空間から音楽を作り出す偶発的な設計、もう一つは音楽から空間を作り上げる計画的な設計である。ここでは、これらの手法の一例を紹介する。ここで紹介するのは、あくまで一例であり、他の手法で設計しても全く問題はない。

4.1 偶発的な設計

偶発的な設計とは、先に空間を設計していき、それに伴う音を調整することで音楽を作り上げていくことである。

最初に適当な空間を設計し、空間が出来たら、音を鳴らしたいオブジェクトに音の設定をして、鍵盤オブジェクト化する。このときに不協和音が発生しないように、隣り合うオブジェクト同士の音の設定を和音になるようにするなどの調整を行う。

この手法には、比較的簡単に作品が出来る、自由な発想で作品が作れるなどのメリットがある。しかし、リズム感のある作品を作るのは難しく、「ただ、音を鳴らしているだけ」と見られがちである。

4.2 計画的な設計

計画的な設計とは、予め演奏したい曲を決めておき、その曲を演奏するように空間を設計していくことである。この手法は、前述の偶発的な設計よりも難易度が高いため、本システムでは、多くの支援機能を使って設計する必要がある。

始めに、BPM 計算機能を用いて、フレーム数を把握する。例えば BPM が 60 の場合、全音符が 240、2 分音符が 120、4 分音符が 60、8 分音符が 30、16 分音符が 15 となる。次

に、適当なところに鍵盤オブジェクトを配置する。その後、先読み機能を用いて、そのオブジェクトが指定のタイミングでどの位置にいるのかを把握する。この時、場合によっては、オブジェクトに初速度を与え、運動エネルギーを稼ぐ必要がある。位置を把握したら、そのすぐ近くに壁となるオブジェクトを配置する。再度、先読み機能とログ機能で指定のタイミングで衝突しているかを確認してから、次の指定のフレームではどの位置にいるのかを把握する。オブジェクトの傾き・反発係数などを調整することにより、意図した動きで意図したタイミングで発音するように調整していく。これらの手順を繰り返し、空間を設計していく。一つの空間で長い時間演奏するのは、この手法では難しいため、長い時間演奏するには、レイヤ機能を用いて、複数の空間を設計する必要がある。

この手法は、前述の偶発的な設計よりも、多くの労力と時間が必要となるが、その作品の技術力は、演奏動画を見れば一目瞭然となる。

4.3 作 例

4.3.1 Domino Piano

偶発的に設計した作例 (図 12)。

鍵盤を分解して作ったドミノで演奏する作品。

4.3.2 Remapped Piano Keys play march

計画的に設計した作例 (図 13)。

先読み機能とレイヤ機能を用いて、分解された鍵盤が「おもちゃの兵隊のマーチ」の一部を演奏する作品。

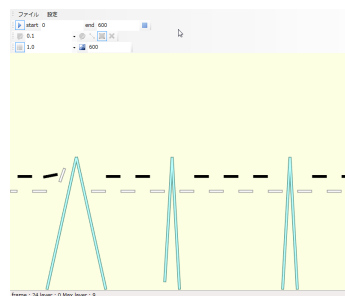


図 13 例：Remapped Piano Keys play march

5. 展 望

今後は、作品制作に要する時間がさらに軽減されるための支援機能の実装と共に、時間の流れの速さを制御できるインターフェースによってテンポコントロールが行えるようにしたい。また、本システムの音声出力は、MIDI 出力のみで、映像と音との遅延が発生するなどの問題がある。そこで、Steinberg 社の Virtual Studio Technology 規格のプラグインで音声を出力することを検討している。これにより、より多くの音色で発音できるようになり、より豊かな楽曲が作れる。また、本システムそのものをプラグイン化し、他のデジタルオーディオワークステーションとの提携を行うことも検討している。これにより、音声にエフェクトをかけたり、録音が容易に行える。楽曲制作そのものが効率良く行えることを期待している。他にも、音の強弱・ピッチなどの表情付けをオブジェクト衝突時の位置やベクトルなどで行うことも検討している。ただし、その表情付けが自然なものになるかは、現時点では不明なので、様々な方法を試していきたい。

6. おわりに

本研究では、物理法則が支配する空間にオブジェクトを配置し衝突によって発音させる音楽表現環境を提案し、その実装を行った。また、実装したシステムで実際に作品を制作した。

今後は、本システムと制作した作品をインターネット上で配布するなどして、本システムを多くの人に利用してもらい、知見や意見を貰いたい。また、利用した人が作品を気軽にインターネット上で配布・評価が出来る環境を作り、本システムによるコミュニティを成形していく。これは、ユーザー同士が作品を評価し合うことにより、作品の技術力を競い合い、よ

り高い技術力を持った作品が作られていくことを期待している。

謝辞 本研究は、情報処理推進機構 (IPA) の 2009 年度上半期末踏 IT 人材発掘・育成事業の支援を受けている。

参 考 文 献

- 1) TENORI-ON,
<http://www.yamaha.co.jp/tenori-on/>
- 2) 【MIDI アニメ】 マリオワールド 1-3 【ステージ再現】 ,
<http://www.nicovideo.jp/watch/sm2290583>
- 3) 組曲『ニコニコ動画』 全自動マリオ Ver,
<http://www.nicovideo.jp/watch/sm3462394>
- 4) リトルビッグプラネット,
<http://www.jp.playstation.com/scej/title/lbp/>
- 5) Phun - 2D physics sandbox - ,
<http://www.phunland.com/wiki/Home/ja>
- 6) AS3 Flash Physics Engine Box2DFlashAS3,
<http://box2dfash.sourceforge.net/>
- 7) ピタゴラススイッチ,
<http://www.nhk.or.jp/kids/program/pitagora.html>
- 8) LittleBigPlanet : Little Big Computer,
<http://www.youtube.com/watch?v=ZiRgYBHoAoU>
- 9) 矢田裕基, 宮下芳明.Gravity Piano: 物理演算による作曲インターフェース, エンターテインメントコンピューティング 2009 pp.181-182 (2009).
- 10) 徳井直生.SONASPHERE : 動的な三次元インターフェースに基づくインタラクティブな音楽環境, 芸術科学会論文誌 (2004).
- 11) エレクトロプランクトン ELECTROPLANKTON,
<http://electroplankton.com/>
- 12) 江渡浩一郎, 渡辺訓章. 粒子の物理運動による音色合成の試み, IPSJ SIG Notes 2007(127) pp.23-24 20071214 (2007).
- 13) 藤岡定, 中村滋延, 栗原詩子.Cubie — パズル・ゲームをモチーフとした「思考型」演奏ソフトウェア, IPSJ SIG Notes 2007(102) pp.73-77 20071011 (2007).
- 14) Wacom — Bamboo Fun キャンペーンサイト (スケッチピストン 4) ,
<http://tablet.wacom.co.jp/bambooplayroom/index.html>
- 15) 野原勇基, 橋口久, 松浦昭洋.Hazumusic: 弾性ボールの仮想バウンスを利用した音楽エンタテインメントシステム, エンタテインメントコンピューティング 2009 pp.87-90 (2009).