



УДК 004.75

## РАСПРЕДЕЛЕННАЯ ГЕНЕРАЦИЯ КОМПЬЮТЕРНОЙ МУЗЫКИ В ИНТЕРНЕТЕ ВЕЩЕЙ

Г.Г. Рогозинский<sup>a, b</sup>, Е.В. Черный<sup>c, d</sup>, Р. Уолш<sup>e</sup>, А.В. Щекочихин<sup>b</sup><sup>a</sup> СПбГУТ им. проф. М.А. Бонч-Бруевича, Санкт-Петербург, 193232, Российская Федерация<sup>b</sup> СПбГУКиТ, Санкт-Петербург, 191119, Российская Федерация<sup>c</sup> Университет ИТМО, Санкт-Петербург, 197101, Российская Федерация<sup>d</sup> Академия Або, Турку, 20500, Финляндия<sup>e</sup> Технологический институт Дандолк, Дандолк, Ирландия

Адрес для переписки: iamoscii@gmail.com

### Информация о статье

Поступила в редакцию 07.04.15, принята к печати 14.05.15

doi:10.17586/2226-1494-2015-15-4-654-660

Язык статьи – русский

**Ссылка для цитирования:** Рогозинский Г.Г., Черный Е.В., Уолш Р., Щекочихин А.В. Распределенная генерация компьютерной музыки в Интернете вещей // Научно-технический вестник информационных технологий, механики и оптики. 2015. Т. 15. № 4. С. 654–660.

### Аннотация

**Постановка проблемы.** Рассмотрена распределенная интеллектуальная система для генерации компьютерной музыки в сети автономных вычислительных агентов. В качестве решения одной из задач, связанных с реализацией такой системы, предложена математическая модель извлечения данных из окружающей среды для использования их в процессе генерации музыки. **Методы.** Для представления данных о тембральных характеристиках в описываемой системе используется Resource Description Framework. В качестве подсистемы синтеза и обработки звука используется специальный язык музыкального программирования Csound. Генерация звука происходит в соответствии с параметрами композиционной модели, извлекающей данные из окружающего мира. **Результаты.** Предложена архитектура распределенной системы генерации компьютерной музыки. Представлен пример ядра синтеза звука. Предложен метод отображения характеристик окружающего мира в пространство сигналов композиционной модели, имитирующей вдохновение у творческого индивидуума. Система генерации музыки была представлена в качестве экспоната в Центральном музее связи им. А.С. Попова в рамках акции «Ночь музеев». В ходе публичного эксперимента установлено, что система в целом проявляет тенденцию к быстрому установлению нейтрального состояния, при котором не генерируются музыкальные события, что говорит о необходимости разработки алгоритмов поддержания активного состояния сети агентов в целом. **Практическая значимость.** Реализация описываемой системы позволит создать технологическую платформу для целого класса новых приложений, связанных с дополненной акустической реальностью и алгоритмической композицией.

### Ключевые слова

компьютерная музыка, Интернет вещей, искусственный интеллект, распределенные системы, компьютерная креативность.

## DISTRIBUTED GENERATION OF COMPUTER MUSIC IN THE INTERNET OF THINGS

G.G. Rogozinsky<sup>a, b</sup>, E.V. Cherny<sup>c, d</sup>, R. Walsh<sup>e</sup>, A.V. Shchekochikhin<sup>b</sup><sup>a</sup> Bonch-Bruевич Saint Petersburg University of Telecommunication, Saint Petersburg, 193232, Russian Federation<sup>b</sup> Saint Petersburg State University of Film and Television, Saint Petersburg, 191119, Russian Federation<sup>c</sup> ITMO University, Saint Petersburg, 197101, Russian Federation<sup>d</sup> Åbo Academy, Turku, 20500, Finland<sup>e</sup> Dundalk Institute of Technology, Dundalk, Ireland

Corresponding author: iamoscii@gmail.com

### Article info

Received 07.04.15, accepted 14.05.15

doi:10.17586/2226-1494-2015-15-4-654-660

Article in Russian

**For citation:** Rogozinsky G.G., Cherny E.V., Walsh R., Shchekochikhin A.V. Distributed generation of computer music in the internet of things. *Scientific and Technical Journal of Information Technologies, Mechanics and Optics*, 2015, vol.15, no. 4, pp. 654–660.

### Abstract

**Problem Statement.** The paper deals with distributed intelligent multi-agent system for computer music generation. A mathematical model for data extraction from the environment and their application in the music generation process is

proposed. **Methods.** We use Resource Description Framework for representation of timbre data. A special musical programming language Csound is used for subsystem of synthesis and sound processing. Sound generation occurs according to the parameters of compositional model, getting data from the outworld. **Results.** We propose architecture of a potential distributed system for computer music generation. An example of core sound synthesis is presented. We also propose a method for mapping real world parameters to the plane of compositional model, in an attempt to imitate elements and aspects of creative inspiration. Music generation system has been represented as an artifact in the Central Museum of Communication n.a. A.S. Popov in the framework of «Night of Museums» action. In the course of public experiment it was stated that, in the whole, the system tends to a quick settling of neutral state with no musical events generation. This proves the necessity of algorithms design for active condition support of agents' network, in the whole. **Practical Relevance.** Realization of the proposed system will give the possibility for creation of a technological platform for a whole new class of applications, including augmented acoustic reality and algorithmic composition.

#### **Keywords**

computer music, Internet of things, artificial intelligence, distributed systems, computer creativity.

### **Введение**

Итогом работ в области применения компьютерных технологий для задач музыкальной композиции в 50-х годах XX века стало рождение так называемой компьютерной музыки и алгоритмической композиции [1]. Композитор стал программистом, управляющим машиной, которая в буквальном смысле вычисляет партитуру музыкального произведения по заранее существующему набору правил и алгоритмов. Описанный подход едва ли применим к традиционным формам музыкального искусства, например, к написанию песен, однако он может быть эффективен при создании продолжительных звуковых ландшафтов, использующихся в звуковом дизайне или в различных формах экспериментального искусства. Становясь распределенной, система алгоритмической композиции приобретает новые свойства. Мир Интернета вещей предоставляет подходящую базу для экспериментов в этой области и новую идеологическую основу для форм коллаборационного компьютерного искусства. Подобные системы могли бы создавать оригинальные объекты искусства посредством сетевого взаимодействия между участниками творческого акта, в котором человек творит совместно с агентами искусственного интеллекта.

Интернет вещей характеризуется большим количеством разнообразных устройств, работающих вместе в одной сети [2, 3]. Разнообразие подключенных к сети устройств обуславливает необходимость обеспечения совместимости, основанной на единых методах представления и обмена данными. Одной из наиболее перспективных технологий по обеспечению этих функций является Semantic Web Stack [4, 5], включающая в себя универсальную модель данных (Resource Description Framework, RDF) с базовым набором терминов (RDF Schema, OWL) для описания понятий высокого уровня.

Задача автоматизации творческого процесса всегда была связана с исследованиями в области компьютерной музыки, где вычислительные методы применялись для формализации отдельных этапов процесса творения [6, 7]. Такого рода системы часто включают в себя так называемое композиционное ядро: набор алгоритмов и правил, описывающих контуры мелодии, гармонию, ритмические рисунки, фактуры и т.д. Таким образом, композиционная модель представляет семантику музыкального фрагмента как набор музыкальных правил и связей между ними. Подобные модели статичны по своей конструкции и не могут применяться для симуляции процесса творчества из-за его динамической природы. Например, музыкальные предпочтения человека меняются со временем, также меняются творческие занятия человека, изменяется парадигма процесса сочинения музыки. Таким образом, композиционная модель должна иметь возможность менять набор существующих фактов и связей между ними.

Исследование авторов настоящей работы посвящено разработке системы распределенной генерации музыки, основанной на потенциале Интернета вещей. Такая проблема требует решения различных задач, одной из которых является разработка метода установления взаимосвязи между окружающей внешней средой и результатами генерации звука.

### **Обзор существующих систем распределенного искусства**

**Хранитель экрана *Electric Sheep*.** Одним из наиболее значимых примеров распределенной системы коллективного творчества является хранитель экрана Electric Sheep [8], который может быть представлен как распределенный объект визуального искусства, непрерывно генерируемый тысячами компьютеров и мобильными устройствами по всей сети. Программа основана на работе генетического алгоритма, который управляет процессом эволюции некоторого начального набора фрактальных примитивов. В начале работы программа скачивает с сервера набор изображений, которые используются для создания новых изображений. Каждый пользователь может голосовать за те фрагменты, которые ему нравятся, повышая тем самым их рейтинг, управляющий ходом эволюции анимационных заставок.

**Проект *EMI*.** Похожий подход был рассмотрен Д. Коупом в работе «Эксперименты над музыкальным интеллектом» – Experiments in Musial Intelligence (EMI) [9]. Новая музыка генерировалась в соответствии с результатами анализа существующих музыкальных произведений. Композиционный процесс включал в себя три этапа: разбор (анализ существующих композиций и разбиение их на структур-

ные части), определение стиля музыкальной композиции и комбинирование выделенных фрагментов в новое произведение. Такая модель описывает не только организацию последовательностей нот и аккордов, но и процесс построения формы всего музыкального фрагмента. При использовании такого подхода результат будет всегда зависеть от набора анализируемых системой произведений.

Основным недостатком описанной системы является ее зависимость от заранее существующей базы музыкальных партитур, которая, в свою очередь, полностью определяет конечный результат работы модели. Это исключает возможность интеллектуального отбора материала для композиции, что мешает использованию алгоритмов для автоматического объединения музыкальных фрагментов по их особенностям.

**The Listening Machine.** В проекте «The Listening Machine» (<http://thelisteningmachine.org>) производился анализ активности 500 пользователей британского сегмента социальной сети Twitter [10]. Используя средства машинного обучения и обработки естественного языка, система генерировала музыкальные фрагменты в реальном времени, основываясь на сообщениях в социальной сети. Проект был активен с мая 2012 г. по январь 2013 г., и в течение всего этого периода времени система непрерывно генерировала и транслировала через Интернет музыкальные фрагменты.

#### **#tweetscapes**

Данный проект [11] представляет собой сервис, преобразующий сообщения пользователей немецкого сегмента сети Twitter в аудиовизуальную информацию путем анализа содержания сообщений. В качестве управляющей информации для ядра синтеза звука служат результаты геолокации сообщения, хеш-тег, информация об авторе и др. В результате короткое сообщение порождает определенный звук с соответствующей визуализацией.

### **Распределенная система генерации алгоритмической композиции**

В данном разделе рассматривается модель автоматической генерации музыки, основанная на распределенной сети автономных агентов. В этой модели каждый агент способен хранить некоторое количество знаний или состояние композиционной модели, использующиеся для генерации музыки. Каждый агент может устанавливать некоторое количество соединений с другими агентами.

Для описания системы введем понятие контекста и определим его как совокупность всех соединений агента. Тем или иным образом контекст влияет на состояние агента, изменяя композиционную модель. Таким образом, процесс генерации музыки становится результатом анализа и обработки информации, содержащейся в соседних объектах. Результаты анализа контекста используются для управления звуковыми синтезаторами. Каждый агент может быть представлен как синтезатор в глобальном оркестре машин, сочиняющих музыку посредством сетевого обмена идеями и музыкальными фрагментами.

Модель использует OWL (Web Ontology Language) для представления знаний и язык музыкального программирования Csound для синтеза звука.

На текущем этапе исследования система существует в упрощенном виде. Для сбора информации из окружающего мира используется набор агентов, способных взаимодействовать с пользователем. В настоящее время реализованы два вида взаимодействия с пользователем: на основе графического пользовательского интерфейса и системы видеослежения.

Каждый агент в системе состоит из следующих функциональных модулей:

- пользовательский интерфейс (UI) собирает информацию с датчиков и обрабатывает ее для предоставления высокоуровневых характеристик взаимодействия со средой;
- сетевой интерфейс (Network Core) позволяет находить других агентов в сети и передавать SPARQL-запросы для получения информации о состоянии агентов;
- база знаний (Knowledge Base) содержит всю информацию об агенте и его окружении. Она используется сетевым ядром для получения информации об окружении и алгоритмическим ядром для установки управляющих параметров композиционной модели;
- управляющее ядро (Decision Core) отвечает за логику работы приложений и обеспечивает взаимодействие модулей друг с другом;
- алгоритмическое ядро (Algorithmic Core) является реализацией композиционной модели. Оно использует знания из композиционной модели для управления композиционной моделью и генерирует набор команд для ядра Csound;
- ядро Csound отвечает за синтез и обработку звука. Оно получает команды от алгоритмического ядра и инициирует различные звуковые события.

Архитектура агента представлена на рис. 1.

Результаты взаимодействия агентов с пользователем можно использовать для создания разнообразных тембральных пространств, применяя строгие алгоритмы композиционной модели к непредсказуемым результатам действия пользователя. Система будет пытаться стабилизироваться в отсутствии взаимодействия с пользователем.

Для синтеза и обработки звуковых сигналов системой используется Csound – высокоуровневый компьютерный язык музыкального программирования [12, 13]. Csound является кросс-платформенной средой, поддерживающей Windows, OS X, GNU/Linux, Android и iOS. Кроме того, он может работать с платформами Raspberry Pi, BeagleBone и Arduino. Csound был разработан Барри Верко в 1985 г. в MIT. Начиная с 90-х годов, язык активно развивается благодаря группе разработчиков и большому сообществу волонтеров. Также любопытно отметить, что Csound является основой языков структурного аудио SASL и SAOL ISO/IEC MPEG4 [14, 15].

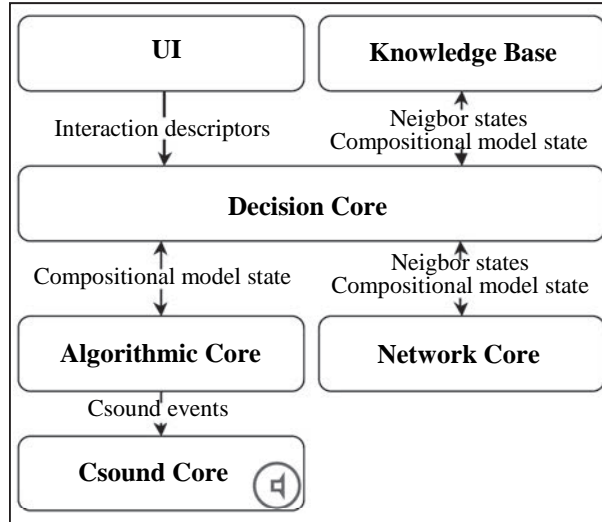


Рис. 1. Архитектура агента системы

Принцип работы Csound основан на модульной структуре. Это означает, что процесс синтеза и обработки звука происходит в результате коммутации базовых модулей: генераторов сигналов, фильтров, устройств временной обработки и т.д. Типичная программа в Csound состоит из двух частей: оркестра (orchestra) – набора описаний структуры инструментов, и партитуры (score) – набора управляющих событий для инструментов, описанных в orchestra.

### Структура синтезатора

Основным музыкальным инструментом в текущей реализации является синтезатор звуковых ландшафтов. Он используется для создания звуков с динамическими тембрами на основе аддитивно-субтрактивного синтеза.

Инструмент состоит из набора голосов, обладающих одинаковой структурой, но отличающихся значениями параметров. Структура одного голоса представлена на рис. 2. Он состоит из генератора белого шума (WG), генератора огибающей (EG), генератора низкой частоты (LFO) и полосового фильтра второго порядка (BP). Сумма всех голосов подвергается временной обработке (блоки DEL и REV). Пунктирной линией на схеме показаны управляющие сигналы, а непрерывными линиями – звуковые.

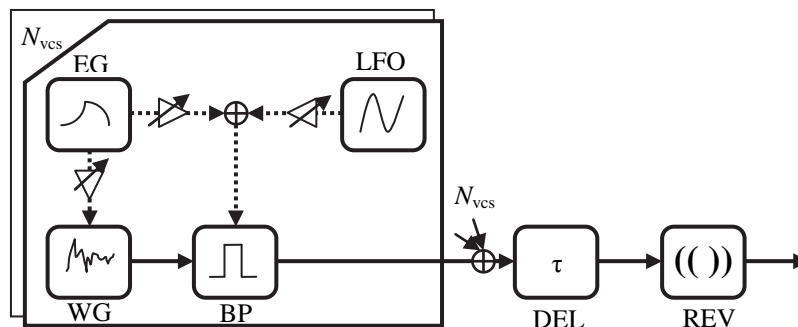


Рис. 2. Структура синтезатора ( $N_{vcs}$  – номер голоса синтезатора,  $\tau$  – время в линии задержки)

Генераторы огибающих каждого из голосов срабатывают в случайном порядке с заданными средней частотой срабатывания и дисперсией, создавая ощущение падающих капель дождя, когда параметры атаки и затухания малы, или создавая пласты звука при относительно больших значениях этих параметров. Также огибающая управляет центральной частотой полосового фильтра.

Добротность фильтров также влияет на характер звучания: при высокой добротности генерируемый звук близок к синусоидальному. С помощью модуляции частоты фильтра можно достичь эффекта

«вибратор». Также возможно использование генератора низкой частоты (LFO) для управления параметрами синтезатора.

### Математическая модель извлечения данных из окружающей среды

В системах, связанных с компьютерной креативностью, используются различные способы управления генерируемым материалом: текстово-звуковое кодирование, интерпретация визуальных образов, взаимодействие с пользователем по абстрактному пользовательскому интерфейсу [6]. Однако каждый раз построение такого отображения обуславливается, в первую очередь, творческой идеей автора. Нами была поставлена задача предложить метод интеллектуального, самостоятельного построения системой способа интерпретации внешнего мира для управления процессом создания музыкальной композиции.

Внешний мир способен меняться, как и сама структура системы, которая может приобретать/терять новые сенсоры, поэтому система должна уметь адаптироваться к подобным изменениям. Кроме того, может меняться и композиционная модель, что, в свою очередь, потребует использования других источников из внешнего мира для интерпретации. Другими словами, подобно человеку, который получает вдохновение от объектов окружающего мира – аудиовизуальных образов, литературы, природных явлений, разрабатываемая система извлекает различные данные с помощью сенсорных полей.

Синтезируемый звук может быть описан в терминах траектории в тембральном пространстве. Тембральное пространство [16] – пространство музыкальных признаков, таких как громкость, длительность атаки, частота модуляции и т.д.

Пусть  $x(t)$  – траектория в тембральном пространстве  $\Xi$ . Это  $n$ -мерная функция времени, где  $n$  – размерность тембрального пространства, определяемая сложностью композиционной модели.

Ядро синтеза звука управляется набором из  $m$  параметров. Обозначим функции управляющих параметров как  $y_j(t)$ . Это могут быть интерпретации данных различной природы из окружающего мира: от показаний сенсоров роботов-агентов до данных, полученных из сети (фондовые рейтинги, прогнозы погоды) или анализа характера дисперсии скорости перемещения стада диких слонов в Африке. Обозначим множество управляющих функций как  $\Sigma$ . Задачей становится построение набора отображений  $\{\zeta_{ij}\}: \Sigma \rightarrow \Xi$ . Функция  $\zeta_{ij}$  задает влияние  $j$ -ой функции управляющих параметров на значение  $i$ -ой координаты траектории в тембральном пространстве.

Семейство функций  $y_i$  описывает поведение уже существующих процессов, внешних по отношению к системе. Они могут быть изучены заранее и подвергнуты анализу. Необходимо определиться с тем, какими характеристиками могут обладать функции координат траектории в тембральном пространстве.

В музыкальном процессе можно выделить различные временные масштабы:

- *HNF* – сверхбыстрые изменения – от 10 до 100 мс, граница области временной разрешающей способности человека, гранулярные процессы,  $\chi_{HNF}(f)$ ;
- *HF* – быстрые процессы, от 0,1 до 1 с, ритмические изменения, отдельные ноты/аккорды,  $\chi_{HF}(f)$ ;
- *MF* – среднечастотные процессы, от 1 до 10 с, отдельные музыкальные фразы,  $\chi_{MF}(f)$ ;
- *LF* – медленные процессы, от 10 до 1000 с, элементы музыкальной формы,  $\chi_{LF}(f)$ ;
- *LLF* – сверхмедленные процессы, свыше 1000 с, масштабы перезагрузки композиционной модели в целом,  $\chi_{LLF}(f)$ .

Функции  $\chi(f)$  фактически являются частотными характеристиками соответствующих процессов. Для построения отображений  $\{\zeta_{ij}\}: \Sigma \rightarrow \Xi$  предлагается разделять функции координат траектории в тембральном пространстве на классы по принадлежности описываемых ими процессов к одному из временных масштабов. Тогда, для построения связывающих отображений должно быть верно уравнение:

$$\text{supp}(\chi(f)) \supset \text{supp}(F\{\zeta_{ij}(y_i(t))\}(f)),$$

т.е. носитель спектра результата отображения управляющей функции в тембральное пространство должен быть сосредоточен внутри носителя одной из частотных характеристик музыкальных временных масштабов. Строгое выполнение данного условия, безусловно, не является определяющим и в реальной системе может выполняться с наперед заданной точностью, т.е. мы допускаем, что управляющие функции не полностью попадают под соответствующее частотное разбиение.

Кроме того, можно производить статистический анализ распределения значений координат в тембральном пространстве. Аналогично можно производить и статистический анализ управляющих функций. Связывающие отображения  $\{\zeta_{ij}\}$  необходимо строить так, чтобы мера разности функций распределения отображения управляющего параметра  $\rho_{\zeta(y)}$  и соответствующей координаты тембрального пространства  $\rho_{x_i}$  не превышала заданной величины  $\varepsilon$ :

$$\mu(\rho_{\zeta(y)} - \rho_{x_i}) \leq \varepsilon.$$

Таким образом, связывающие отображения, удовлетворяющие данным условиям, будут обеспечивать «умное» управление музыкальными параметрами, т.е. соответствующие музыкальные признаки будут изменяться по правилам, присущим данной композиционной модели.

Кроме того, к задаче можно подходить и с обратной стороны. Следует изначально ограничить класс возможных преобразований  $\{\zeta_{xy}\}$ , в простейшем случае – линейными функциями. Тогда они будут просто выполнять роль согласования масштабов между координатами тембрального пространства и управляющими функциями. В то же время выписанные условия можно использовать для определения возможности использования некоторой управляющей функции для конкретной координаты тембрального пространства.

### Заключение

Авторами была предложена концепция распределенной компьютерной системы для создания музыки, особенностью которой является использование семантического описания состояния узлов сети для моделирования агентов и языка структурного аудио для описания алгоритмов генерации звука.

Предложенный в работе способ извлечения информации из окружающей среды позволит системе имитировать творческий процесс человека, в основе которого лежит метод интерпретации показаний сенсоров в соответствии с характером их изменения. Исследуя и подвергая анализу различные нестационарные процессы, система позволяет связать их особенности с параметрами композиционной модели, устанавливая тем самым взаимосвязь между окружающей внешней средой и результатами генерации звука.

Результаты работы были представлены в Центральном музее связи им. А.С. Попова в рамках акции «Ночь музеев»<sup>1</sup>.

Система состояла из пяти ноутбуков, объединенных в локальную сеть, каждый из которых был подключен к отдельному звуковому монитору. В качестве источника входных данных для модели использовались веб-камеры, анализирующие активность зрителей. Компьютеры располагались в ряд перед аудиторией. В ходе эксперимента было установлено, что система в целом проявляет тенденцию к быстрому установлению нейтрального состояния, при котором не генерируются музыкальные события. Это объясняется тем, что в описанную модель не было заложено функции, определяющей поведение системы при отсутствии поступающих данных (низкая активность зрительского зала обусловлена форматом представления).

Проведенный эксперимент дает представление о путях дальнейшего развития системы и метода. Главной проблемой на текущий момент является отсутствие алгоритмов поддержки активного состояния сети агентов, из-за чего останавливается процесс генерации музыки. Также имеет смысл направить дальнейшие исследования на усложнение алгоритмов взаимодействия между агентами системы, совершенствования ядра генерации звука и тестирование интеллектуального извлечения знаний системой.

### Литература

1. Edwards M. Algorithmic composition: computational thinking in music // *Communications of the ACM*. 2011. V. 54. N 7. P. 58–67. doi: 10.1145/1965724.1965742
2. Kortuem G., Kawsar F., Sundramoorthy V., Fitton D. Smart objects as building blocks for the internet of things // *IEEE Internet Computing*. 2010. V. 14. N 1. P. 44–51. doi: 10.1109/MIC.2009.143
3. Гольдштейн Б.С., Кучерявый А.Е. Сети связи пост-NGN. СПб.: БХВ-Петербург, 2014. 160 с.
4. Berners-Lee T., Hendler J., Lassila O. The semantic web // *Scientific American*. 2001. V. 284. N 5. P. 34–43. doi: 10.1038/scientificamerican0501-34
5. Dadzie A.-S., Rowe M., Petrelli D. Hide the stack: toward usable linked data // *Lecture Notes in Computer Science*. 2011. V. 6643. P. 93–107. doi: 10.1007/978-3-642-21034-1\_7
6. Colton S., Wiggins G.A. Computational creativity: the final frontier? // *Frontiers in Artificial Intelligence and Applications*. 2012. V. 242. P. 21–26. doi: 10.3233/978-1-61499-098-7-21
7. McDermott J. Functional representations of music // *Proc. 3<sup>rd</sup> Int. Conf. on Computational Creativity*. Dublin, Ireland, 2012. P. 224.
8. Draves S. The electric sheep screen-saver: a case study in aesthetic evolution // *Lecture Notes in Computer Science*. 2005. V. 3449. P. 458–467.
9. Cope D. *Computer Models of Musical Creativity*. Cambridge: MIT Press, 2005. 456 p.
10. Jones D., Gregson P. The listening machine: generating complex musical structure from social network communications // *Proc. 50<sup>th</sup> Anniversary Convention of the AISB*. London, 2014. 6 p.
11. Hermann T., Nehls A.V., Eitel F., Barri T., Gammel M. Tweetscaples – real-time sonification of twitter data streams for radio broadcasting // *Proc. 18<sup>th</sup> Int. Conf. on Auditory Display*. Atlanta, USA, 2012. P. 113–120.
12. Vercoe B. *Csound: A Manual for the Audio-Processing System*. MIT Media Lab, 1995.
13. Boulanger R. *The Csound Book: Perspectives in Software Synthesis, Sound Design, Signal Processing, and Programming*. Cambridge: MIT Press, 2000. 782 p.

<sup>1</sup> Ссылка на анонс события в социальной сети Facebook <https://www.facebook.com/events/528781420513115/>

14. ISO/IEC 14496-3:2009. Information Technology – Coding of Audio-visual Objects – Part 3: Audio – Subpart 5: Structured Audio (SA). Geneva, International Organization for Standardization. 1381 p.
15. Scheirer E.D. The MPEG-4 structured audio standard // Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP-1998). Seattle, USA, 1998. V. 6. P. 3801–3804. doi: 10.1109/ICASSP.1998.679712
16. Wessel D.L. Timbre space as a musical control structure // Computer Music Journal. 1979. V. 3. N 2. P. 45–52.

- Рогозинский Глеб Гендрихович* – кандидат технических наук, руководитель направления, СПбГУТ им. проф. М.А. Бонч-Бруевича, Санкт-Петербург, 193232, Российская Федерация; доцент, СПбГИКиТ, Санкт-Петербург, 191119, Российская Федерация, iamoscii@gmail.com
- Черный Евгений Викторович* – аспирант, Университет ИТМО, Санкт-Петербург, 197101, Российская Федерация; аспирант, Академия Або, Турку, 20500, Финляндия, iamoscii@gmail.com
- Уолш Рори* – лектор (Ирландия), лектор, Технологический институт Дандолк, Дандолк, Ирландия, rorywalsh@ear.ie
- Щекочихин Алексей Викторович* – аспирант, СПбГИКиТ, Санкт-Петербург, 191119, Российская Федерация, ashkochikhin@gmail.com
- Gleb G. Rogozinsky* – PhD, Deputy discipline head, Bonch-Bruevich Saint Petersburg University of Telecommunication, Saint Petersburg, 193232, Russian Federation; Associate professor, Saint Petersburg State University of Film and Television, Saint Petersburg, 191119, Russian Federation, iamoscii@gmail.com
- Eugene V. Cherny* – postgraduate, ИТМО University, Saint Petersburg, 197101, Russian Federation; postgraduate, Åbo Academy, Turku, 20500, Finland, iamoscii@gmail.com
- Rory Walsh* – Master, lecturer, Dundalk Institute of Technology, Dundalk, Ireland, rorywalsh@ear.ie
- Alexei V. Shekochikhin* – postgraduate, Saint Petersburg State University of Film and Television, Saint Petersburg, 191119, Russian Federation, ashkochikhin@gmail.com