

ПРОЦЕДУРА ПРОЕКТИРОВАНИЯ И НАСТРОЙКИ ЭФФЕКТА РЕВЕРБЕРАЦИИ В ПРОГРАММНОМ КОМПЛЕКСЕ AUDACITY® НА ОСНОВЕ ЯЗЫКА ПРОГРАММИРОВАНИЯ NYQUIST

PROCEDURE OF DESIGN AND ADJUSTING THE REVERBERATION EFFECT IN AUDACITY® SOFTWARE COMPLEX DRIVEN BY NYQUIST PROGRAMMING LANGUAGE

V. Taran

Summary. The paper covers engineering and technical aspects of audio material processing by computer methods. Audacity® editor and Nyquist programming language are basic auxiliary technical means for the material processing. The paper both from theoretic and practical point of views considers one of the most problematic issues in the area of computer engineering — reverberation of audio data. Relying on the impressive spectrum of technical manuals and carefully crafted problem oriented literature the author carries out the analysis of issues affecting the applicative features of audio signal formation by using simulation algorithms. Simulation of sound reverberation algorithms are the basis of ongoing research. Guided by his own technical experience as well as programming practices the author suggests personal concept of applying reverberation algorithms to processing audio material. Spring-controlled comb reverberation plug-in being developed by the author could be used as one of the samples when spatial emulating for the formation of 3D effects enriching of composition with original design. And in some (special) cases — when restoring the audio material. The argumentation of the author in favor of using the Audacity® software complex as a multifunctional tool for processing audio material allows expanding the multidimensional facets of scientific discussion in this area. Such a discussion will undoubtedly shed light on many areas concerning private engineering practices of audio processing.

Keywords: Audacity® editor, Nyquist programming language, NyquistIDE, NyquistPrompt, Nyquist-processing, LISP oriented programming languages, computer processing of audio material, audio data processing algorithms, simulation algorithms, space geometry, acoustics, reverberation, reverberation behaviors, spring-controlled comb reverberation.

Таран Василий Васильевич

Кандидат культурологии

Всероссийский институт научной и технической

информации РАН

allscience@lenta.ru

Аннотация. Статья освещает инженерно-технические аспекты обработки аудиоматериала компьютерными методами. Главными вспомогательными техническими средствами по обработке аудиоматериала выступают — редактор Audacity® и язык программирования Nyquist. В статье рассматривается одна из самых проблемных тематик в области компьютерной аудиоинженерии — реверберация аудиоданных как с теоретической, так и с научно-практической точек зрения. Автор, основываясь на внушительном спектре технических инструкций и тщательно проработанной проблемно-ориентированной литературе, проводит анализ вопросов, затрагивающих прикладные особенности формирования аудиосигнала с использованием имитационных алгоритмов. В основе проводимого исследования лежат алгоритмы имитационного моделирования реверберации звука. Руководствуясь собственным техническим опытом, а также практиками программирования, автор предлагает персональную концепцию применения алгоритмов реверберации к обрабатываемому аудиоматериалу. Модель плагина пружинно-гребенчатой реверберации, разработанная автором, может быть использована в качестве одного из образцов при виртуализации пространств с целью создания трёхмерных эффектов, насыщающих звуковые композиции оригинальным дизайном, а в некоторых (особых) случаях — при реставрации аудиоматериала. Аргументация автора данной статьи в пользу применения программного комплекса Audacity® в качестве мультифункционального инструмента обработки аудиоматериала, позволяет расширить многоаспектные грани научной дискуссии в данном направлении, которая, несомненно, прольёт свет на многие области, касающиеся частных инженерных практик обработки аудиоматериала.

Ключевые слова: редактор Audacity®, язык программирования Nyquist, NyquistIDE, NyquistPrompt, Nyquist-обработка, LISP-ориентированные языки программирования, компьютерная обработка аудиоматериала, алгоритмы обработки аудиоданных, имитационные алгоритмы, геометрия пространств, акустика, реверберация, свойства реверберации, пружинно-гребенчатая реверберация.

Вобрав в себя лучшие визуально-ориентированные (технические) инструменты по редактированию аудиоматериала и интегрировав в свою основу командно-ориентированное приглашение языка программирования Nyquist (далее по тексту — язык Nyquist) в сочетании с программой спут-

ником NyquistIDE — Audacity® вполне заслуживает характеристик полноценного профессионального программного комплекса по редактированию звука [1,2]. Вся история проекта Audacity® — это непрерывное совершенствование собственной (уникальной) интерфейсной оболочки и программного ядра, направлен-

ное на улучшение качества обработки аудиоматериала программным и интерфейсно-ориентированным путями.

Благодаря открытому распространению своего программного обеспечения, команда Audacity Team имеет возможность учитывать пожелания потенциальных пользователей редактора Audacity®, среди которых есть как профессионалы своего дела, так и любители, обладающие базовыми знаниями в рассматриваемой нами области. Первые в свою очередь могут являться разработчиками данного программного обеспечения, особенно если они имеют достаточный опыт в сфере проектирования модулей обработки звука в сочетании с навыками высокоуровневого программирования.

Такие люди могут быть чрезвычайно полезны для Audacity Team, поскольку они могут обладать уникальным опытом в области редактирования аудиоматериала аналоговым способом либо аналогово-цифровым, и могут являться незаменимыми советниками в процессах проектирования цифровых — интерфейсно-ориентированных и программных модулей Audacity®.

В свою очередь любители могут выступать в качестве тестировщиков программного обеспечения, сообщая в службу поддержки Audacity® ценные технические сведения, собранные в результате применения таких модулей к обработке аудиоматериала, среди которых могут быть: устойчивость отклика модуля при обработке больших массивов аудиоданных, удобство схемы интерфейса модуля, настройка модуля в разных версиях программы и прочие важные характеристики.

Открытость проекта Audacity® даёт возможность любому человеку, имеющему соответствующую квалификацию, находящемуся в удалении от основного офиса разработки Audacity®, дорабатывать исходный код программы под свои нужды, согласно лицензии GNU. Все, перечисленные выше аргументы в пользу Audacity®, делают этот программный комплекс лидером среди открытого программного обеспечения в сегменте аудиопроизводства.

Audacity® присуща развитая система модулей обработки аудиоданных, среди которых есть и Nyquist-модули. Модули¹, написанные на языке Nyquist, отличаются

¹ Прим. автора. Модуль и «плагин» являются равнозначными понятиями с точки зрения подключаемого периферийного интерфейса обработки аудиоинформации. Модуль может являться плагином, в случае адаптации под него больших фрагментов кода одного языка программирования, но имеющего разный синтаксис. Модуль, как правило, прикрепляется к секции, в условиях редактирования аудиоданных секцией — может служить дорожка или набор дорожек. В первом случае программа ведёт

гибкостью настройки, возможностью переноса такой настройки в другие версии программы, а также позволяют вести доработку алгоритма модуля до уровня, соответствующего запросам инженера по звуку [3]. Такое положение дел выводит Nyquist-программирование в особый ранг инженерного ремесла. Программирование данного типа позволяет производить редактирование аудиоматериала, моделируя и применяя к нему различные психоакустические образы², с возможностью гибкой корректировки аудиосигнала и дальней-

обработку данных в неявном режиме, а во втором — сумма данных обрабатывается в явном режиме. При этом модуль использует оболочку штатного интерфейса программы и является зависимым от её архитектуры. Плагин, как правило, это автономное виртуальное устройство (с собственным интерфейсом), подключаемое к программе, осуществляющей поддержку технологий-посредников, к примеру, VST или AAX. И хотя между данными понятиями всё же имеется небольшая разница, для упрощения аппарата наших терминов мы уравниваем эти понятия, поскольку надлежащие технические инструкции, опубликованные в интернете командой Audacity® применительно к арсеналу Nyquist-программирования, оперируют термином «плагин». Это может происходить по причине того, что многие разработчики используют англоязычный термин «standalone» по отношению к плагинам, поддерживающим статус полностью автономной программы, позволяющей вводить и выводить данные напрямую с помощью интерфейса. В тоже время плагин может иметь и зависимый статус. Причинами могут выступать высокая ресурсоёмкость программного кода, которая задействует скрытые резервы программы, а также тяжёлый интерфейс, включающий в свою оболочку трёхмерные элементы дизайна конструкции.

² Прим.автора. (по определению автора — Таран В.В., 2022) Психоакустические образы — в компьютерном моделировании аудиоматериала — это акустические элементы, имитирующие определённые характеристики* звучания аудиоматериала.

*Обычно подобные характеристики задаются оператором обработки аудиоматериала (звукорежиссёром, саундпродюсером, саунддизайнером). Характеристики можно отнести к специализированным техническим эффектам, создающим, как правило, пространственную имитацию, позволяющую сформировать у потенциального приёмника аудиоинформации (слушателя) картину псевдореальности, погружая слушателя в искусственно созданную аудиальную атмосферу. В качестве примера «пространственной имитации» как раз может выступить реверберация. Реверберация, являясь физическим процессом, может быть успешно смоделирована компьютерными алгоритмами на основе которых с помощью аналогово-цифрового и обратного цифро-аналогового преобразования можно до бесконечности воспроизводить её образ, корректируя присущие ей акустические свойства. Рассматриваемую нами «пространственную имитацию», моделируемую компьютерными алгоритмами, можно разделить на два вида:

Пространственная имитация, цикл которой произведён методом разбивки аудиоматериала на микроотрезки времени, хронометраж которых может определяться наименьшими значениями числовых данных (нано-, зепто-, секунды)**.

Пространственная имитация, цикл которой смоделирован технологиями компьютерного аудиосинтеза***

** В первом случае, это наиболее приемлемая форма построения алгоритма реверберации с учётом числовых характеристик её цикла. Часто используется в качестве основного образца применения эффекта реверберации к обрабатываемому аудиоматериалу.

*** Во втором случае, это более сложная система алгоритмов, использующая технологии аддитивного и субтрактивного синтеза, применяемого к звуковой волне аудиоматериала, причём сам аудиоматериал может являться импульсным источником для повторения огибающей основного (обрабатываемого) звука, заранее предобработанного алгоритмом сигнала. Таким образом, происходит микширование на одной дорожке предобработанного аудиосигнала с основным звуковым материалом.

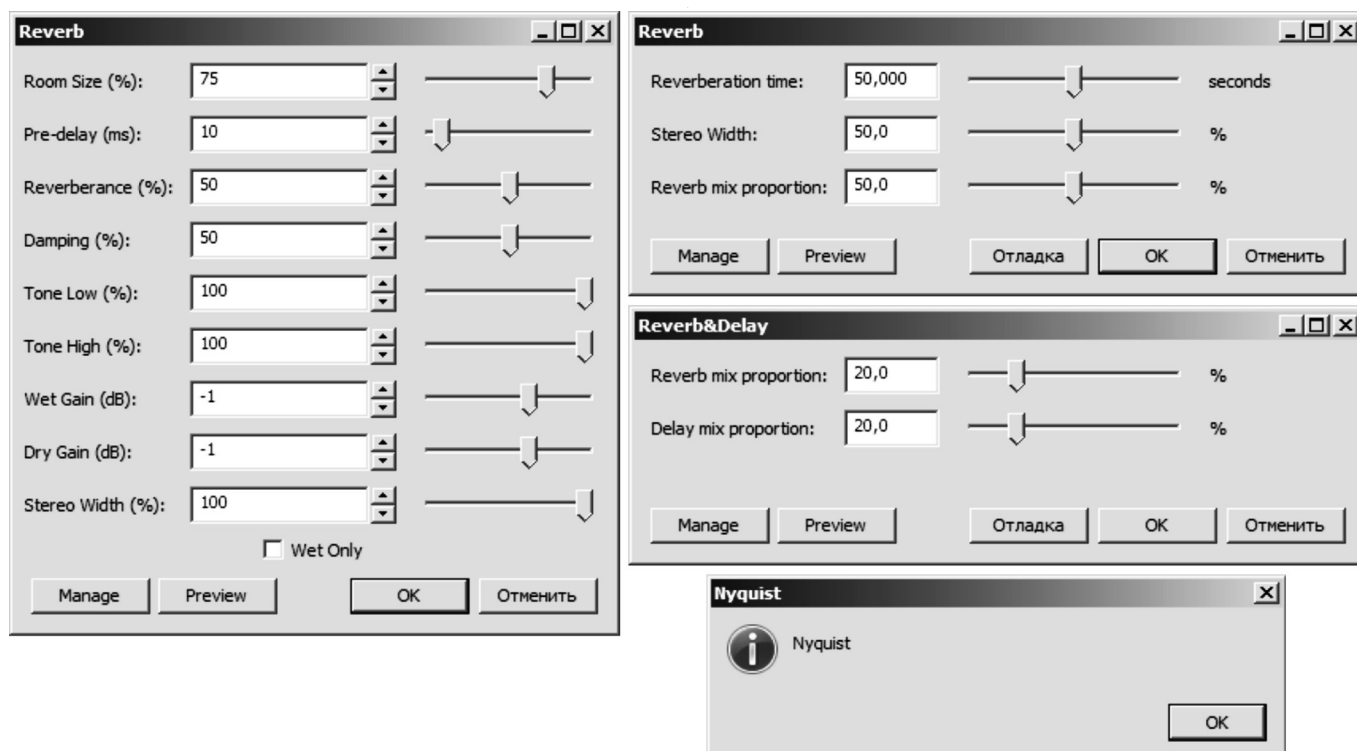


Рис. 1. Работа с модулями реверберации в программном комплексе Audacity®.

шего его вывода через цифровые библиотеки записи и чтения этого вида данных [4,5]. Одним из главных водоразделов изучения компьютерной аудиоинженерии и смыкающейся с ней аудиоинформатики, является психоакустика.

Психоакустика — это научное направление, сформировавшееся под влиянием нескольких дисциплин: акустики (как раздела общей физики), медицины, психологии и в недавнем прошлом — компьютерных наук (информатики). Психоакустика изучает аспекты влияния звука на человека. Данное направление исследований помогает разобраться, как и каким именно образом, человеческий мозг воспринимает звуковые сигналы [6].

В современном понимании психоакустика также затрагивает исследования, посвящённые вопросам моделирования психоакустических артефактов, помогающих создавать образное пространство в человеческой памяти и тем самым лучше запоминать и обрабатывать аудиоинформацию [7]. Одним из таких артефактов является реверберация [8,9].

Реверберация с точки зрения физики (акустики) — это способность волн производить кратковременные отклики, характеризующиеся отражением звука от границ замкнутого пространства.

С точки зрения медицины и психологии реверберация не имеет устоявшегося определения в силу недостаточного изучения мыслительных процессов, происходящих в человеческом мозге. Однако это не мешает нам не замечать реверберацию в условиях жизни, так или иначе мы сталкиваемся с ней повсюду, к примеру, когда слышим речь в замкнутом объёмном пространстве.

Некоторые характеристики реверберации взяты за основу профессиональными звукоинженерами для моделирования звуковых эффектов, задающих текстурные ландшафты, широко применяемые в звукозаписи, составлении алгоритмических композиций, компьютерных играх, кинематографии, трёхмерном моделировании, радиовещании и телевидении [10]. И это вполне объяснимо, потому что во всех перечисленных формах прикладного искусства, часто бывает просто необходимо вызвать у аудитории образ личного участия в моделируемых (постановочных) процессах — создать эффект присутствия (соучастия).

В аудиопроизводстве эффект присутствия может иметь разные формы, но все они так или иначе имеют связь с реверберацией. Например, эффект классической вибрации звука (vibe-эффект), создающий резонансные отклики, в основе которого используется наносекундная (иногда зептосекундная) задержка. Эф-

фект дилея (delay), также строится на похожем принципе, однако в этом случае существует разница в задержке сигнала по промежуточному хронометражу. И это только наиболее яркие (широко применяемые) эффекты, в основу которых заложены принципы реверберации.

Компьютерные технологии смогли дать огромную почву для проведения исследований, связанных со звуком и его акустическими артефактами. В понимании прикладных инженерно-технических процессов компьютерные технологии предоставили широчайшую базу для проведения экспериментов со звуком.

В Audacity®, в принципе, как и в любом другом редакторе по звуку, реверберация является одним из центральных звеньев периферийных модулей обработки звука (плагинов), физически моделирующих эффекты присутствия [11,12]. Реализация реверберации в Audacity® происходит как через штатный интерфейс, так и через периферийный — интерфейс Nyquist. Штатный интерфейс задействует dll-инструкцию «revdelay_1605», которая обеспечивает стандартный набор опций, необходимых для настройки эффекта реверберации (рисунок № 1 — левая сторона), периферийный интерфейс (рисунок № 1 — правая сторона) имеет возможность подключать Nyquist-модули настройки, которые могут быть запрограммированы оператором аудиомонтажа (программистом по звуку).

Примечание: *слева* — штатный интерфейс модуля реверберации с визуально-ориентированными опциями настройки реверберационного фона. *Справа*, периферийный интерфейс модулей реверберации и дилея (delay), спроектированный на языке Nyquist.

Штатный интерфейс модуля реверберации, проиллюстрированный на рисунке № 1, даёт возможность настраивать звуковые отклики в рамках стандартных установок программы. Это устоявшиеся среди инженеров по звуку, а также среди людей, профессией которых является обработка аудиоматериала, *наработки*, проверенные временем и реализованные в виде перечня опций с возможностью визуальной корректировки посредством ползунка. Интерфейсная реализация, в данном случае, позволяет получить приемлемые результаты в сжатые сроки, и очень полезна при стандартных процедурах обработки аудиоматериала.

Другое дело, когда инженеру необходимо использовать подобные инструменты для достижения нестандартных эффектов при редактировании звуковых массивов повышенной сложности. Для подобных экспериментов и предусмотрено стандартное приглашение языка Nyquist, встроенное в общую интерфейсную

архитектуру программы Audacity®, которое позволяет проектировать модули обработки аудиоданных для конкретных целей с последующим их использованием через штатный интерфейс Audacity®. Помимо прочего, поддержка языка Nyquist внутри Audacity®, позволяет вести прямую отладку программного кода (с переключением на текущую версию синтаксиса Nyquist¹) проектируемого модуля, не отрываясь от интерпретации звукового массива.

Это обстоятельство открывает неограниченные возможности для экспериментов с обработкой аудиоданных и применения к ним специализированных эффектов. Чтобы сравнить возможности штатного модуля реверберации в Audacity® и его аналога в приглашении Nyquist, предлагаем провести *опциональный* анализ возможностей этих двух модулей — спутников.

Штатный модуль Reverb — «Реверберация» как уже говорилось выше, предназначен для стандартных операций с реверберацией, он добавляет быстрые и, местами изменённые повторы², в структуру аудиоматериала, тем самым, вызывая у человека (слушателя) эффект плавного изменения воспроизводимого звукового ландшафта. Как следует из технической документации программы Audacity®, эффект основан на оригинальном алгоритме «freeverb», который успешно реализовывал функции подмешивания гармоничных откликов, добавляя резонансную реверберационную свёртку в полезную структуру аудиоматериала. Эффект реверберации в зависимости от типа аудиоматериала³ бывает очень полезен в следующих случаях:

¹ Прим.автора. Начиная с третьей версии, Nyquist поддерживает правила SAL — обеспечивающие альтернативу синтаксису LISP.

² Прим.автора. Эффект любой реверберации можно охарактеризовать кратковременными «повторами», повторы — это своего рода импульсы, выталкивающие аудиосигнал из интервалов, предварительно установленных оператором (заданных алгоритмом). Интервалы определяются временем, минимальный интервал, это и есть время задержки аудиосигнала, задающее темп и колорит реверберации.

³ Прим.автора. Тип аудиоматериала может быть разным, как правило, он зависит от содержимого, формирующего общий аудиосигнал. Тип содержимого также может иметь различные, между собой, акустические характеристики. К примеру, это может быть сигнал, содержащий запись клавишных, струнных либо духовых инструментов, с развитием электронной музыки сюда также можно добавить оригинальные звуки, получаемые в результате компьютерного синтеза. Весь этот материал может быть перемешан и, что очень важно, степень насыщенности каждого из присутствующих в звукозаписи аудиосистем, может варьироваться в отличающихся друг от друга диапазонах громкости, не говоря уже о спектроакустических параметрах. Поэтому, чтобы упростить понимание рассматриваемых нами процессов, мы условно разделим аудиоматериал на два типа:

Аудиоматериал с моносодеждимым.

Аудиоматериал со смешанным содержимым*.

Такое искусственное разделение аудиоматериала на типы даст нам возможность упрощённо представить его основные акустические свойства.

*См. статью автора: «Корректировка аудиосигнала при монтаже аудиозаписей в программной среде Audacity», используя мультифункцио-

Таблица 1. Штатные опции модуля классической реверберации в Audacity®, осуществляющие управление над процессами обработки аудиоматериала.

Опция	Ед.измерения	Инструкция	Устан. Ед.измерения	
1	Room Size	%	Размер помещения	%
2	Pre-delay	ms*	Предварительная задержка	%
3	Reverberance	%	Реверберация	%
4	Damping	%	Демпфирование колебаний	%
5	Tone Low	%	Низкий тон	%
6	Tone High	%	Высокий тон	%
7	Wet Gain	дБ	Влажный коэффициент усиления	%
8	Dry Gain	дБ	Сухой коэффициент усиления	%
9	Stereo Width	%	Стереодиапазон	%

* Прим.автора. Предварительная задержка (в мс — миллисекундах (millisecond) — 10^{-3} секунд); микросекунда (microsecond); сокращенно — μ s; 1 микросекунда = 10^{-6} секунды.

- ◆ При необходимости создания, плавно удаляющегося от первоисточника, протяжения звука.
- ◆ При необходимости обогащения аудиоматериала эффектом пространственной среды.
- ◆ При необходимости регенерации гармоник, пропадающих или опускающихся ниже установленного порога пиковой громкости, в результате некорректного использования аналогово-цифрового преобразования.
- ◆ При необходимости поддержания эффекта псевдо стереофазы на моно каналах¹.
- ◆ При изменении геометрии пространства (виртуализации) архитектурных сооружений, внутри которых транслируется звук.
- ◆ При недостатке естественной реверберации в силу её поглощения условиями окружающей среды.

- ◆ При проведении процедур, направленных на формирование аудиоматериала (включая синтез звуковых комбинаций — таких как мастеринг и компьютерный дизайн звука).
- ◆ При проведении процедур реставрации аудиоматериала и его ремастеринге (премастеринге).

Для всего пула перечисленных выше процедур, в ограниченном порядке² может быть применён штатный модуль реверберации. Для этого в данном модуле предусмотрены специализированные заводские опции, позволяющие максимально гибко настроить эффект реверберации. Для удобства понимания всех опций расшифруем их в табличном виде (таблица 1).

Размер помещения — задаёт размер моделируемой комнаты (кабинета). 0% — это как шкаф, 100% — как

нальные возможности языка программирования Nyquist», в журнале Современная наука актуальные проблемы теории и практики: Серия естественные и технические науки//Информатика, вычислительная техника и управление.— 2021.— № 3.— С. 187–188 [4].

¹ Прим.автора. Применение небольшого объёма стерео реверберации к необработанному монофоническому сигналу, дублированному в двухканальную стереодорожку, обычно делает его звучание более естественным.

² Прим.автора. Под ограниченным порядком имеется в виду, что штатный модуль, имитирующий реверберацию, не может учесть всех созидательных идей, выдвигаемых оператором технологического процесса. Работа со звуком, его дизайном — это, прежде всего, сложная техническая работа, требующая креативного мышления. Чем выше квалификация программиста-звукоинженера, тем выше его потребности в опциональных возможностях соответствующего отрасли программного обеспечения. И, конечно, в таком случае интерфейсных опций становится уже недостаточно.



Рис. 2. Опции переключения между влажным и сухим режимами реверберации.

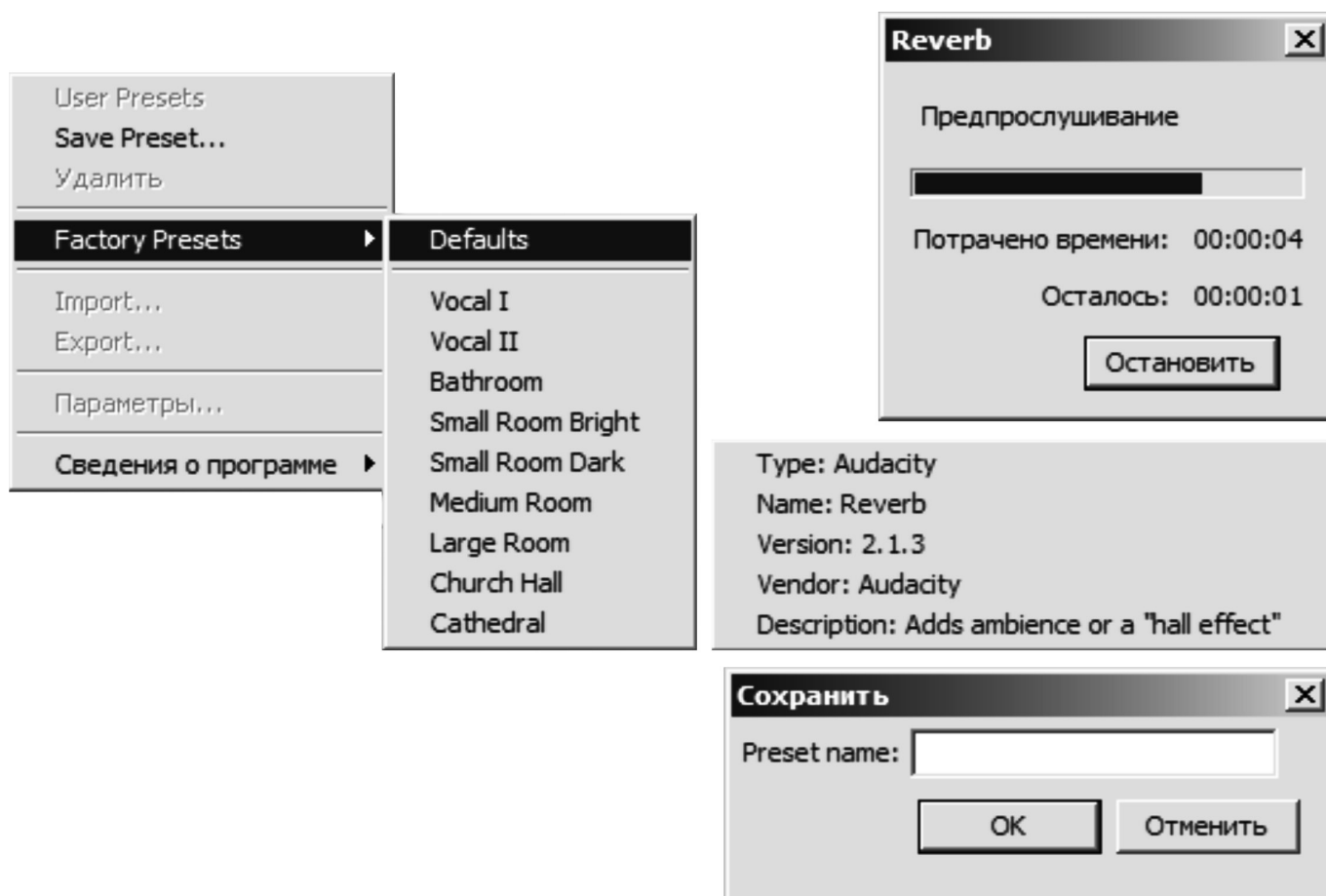


Рис. 3. Фрагменты интерфейса, демонстрирующие штатные предустановки модуля «Реверберация» в редакторе Audacity®.

огромный собор или большая аудитория. Высокое значение будет имитировать эффект реверберации большой комнаты, а низкое значение будет имитировать эффект маленькой комнаты.

Предварительная задержка — задерживает начало реверберации на установленное время после начала исходного ввода. Также задерживает начало хвоста реверберации. Максимальная предварительная задержка составляет 200 миллисекунд. Тщательная настройка этого параметра может улучшить чёткость результата.

Реверберация — задаёт длину хвоста реверберации. Определяет, как долго продолжается реверберация по-

сле окончания реверберации исходного звука и, таким образом, имитирует «живость» акустики помещения. Для любого заданного значения реверберации её хвост будет больше для вместительных размеров помещения.

Демпфирование — увеличение демпфирования приводит к более «приглушённому» эффекту. Реверберация нарастает не так сильно, и высокие частоты затухают быстрее, чем низкие частоты. Имитирует поглощение высоких частот при реверберации.

Низкий тон — установка этого режима ниже 100% уменьшает низкочастотные компоненты реверберации, создавая менее «гулкий» эффект.

Таблица 2. Русскоязычные эквиваленты названий штатных предустановок, входящих в модуль «Реверберация» в редакторе Audacity®, имитирующие пространства различных геометрических форм.

Заводские настройки		Эквивалент на русском языке
1	Vocal I	Вокал I
2	Vocal II	Вокал II
3	Bathroom	Ванная
4	Small Room Bright	Небольшое, хорошо освещённое помещение
5	Small Room Dark	Небольшое тёмное помещение
6	Medium Room	Помещение среднего размера
7	Large Room	Помещение большого размера
8	Church Hall	Церковный зал
9	Cathedral	Кафедральный собор



Рис. 4. Центральная панель управления модулями обработки сигнала в Audacity®.

Высокий тон — установка этого режима ниже 100% уменьшает высокочастотные компоненты реверберации, создавая менее «яркий» эффект.

Влажный коэффициент усиления — прилагает регулировку громкости к компоненту «влажной» реверберации в миксе. Увеличение этого значения относительно «Сухого коэффициента усиления» (см. ниже) увеличивает силу реверберации.

Сухой коэффициент усиления — прилагает регулировку громкости к исходному («сухому») звуку в миксе. Увеличение этого значения относительно «Влажного коэффициента усиления» (см. выше) уменьшает силу реверберации.

Если значения «Влажного коэффициента усиления» и «Сухого коэффициента усиления» одинаковы, то микс «влажного» эффекта и «сухого» звука, выводимый на до-

рожку, будет сделан громче, либо тише, именно на это значение (при условии, что режим «Только влажный»¹, показанный ниже, не проверяется, рисунок 2).

Сtereo диапазон — задаёт видимую «ширину» эффекта реверберации только для стереодорожек. Увеличение этого значения приводит к большему изменению между левым и правым каналами, создавая более «пространственный» эффект. При установке на ноль, эффект применяется независимо к левому и правому каналам. Только влажный — когда этот режим управления уста-

¹ Прим.автора. «Только влажный» — если данный режим активен, то аудиосигнал будет насыщен реверберацией максимально, если же снят (в зависимости от настроек) — реверберация будет практически не слышна. Режим «Только влажный» хорошо использовать для предпрослушивания записи, на основу которой налагается эффект реверберации. Включение данного режима позволит на слух соотнести баланс между оригинальным аудиоматериалом и его преобразовываемой редактором Audacity® копией.

новлен, в результирующем выходе будет только «влажный» сигнал (добавленная реверберация), а исходный звук будет удалён.

Это может быть полезно при предварительном просмотре эффекта, но в большинстве случаев вы должны снять этот режим при применении эффекта. Однако режим «Только влажный» можно использовать для создания трека «Только с реверберацией», который затем можно микшировать в большей или меньшей пропорции с исходным треком. Полные действия приведены в примерах ниже (рисунок № 3, таблица 2).

Модуль предполагает следующие заводские настройки (таблица 2).

Центральная панель управления одинакова для всех штатных плагинов в Audacity® и имеет следующий вид (рисунок 4).

- ◆ Manage — Управление.
- ◆ Preview — Предварительный просмотр (прослушивание).
- ◆ ОК — подтверждение ввода данных.
- ◆ Отменить — отмена ввода данных.

Управление даёт возможность использовать раскрывающееся меню, позволяющее сохранять или удалять пользовательские настройки для инструмента, выбирать заводские настройки, включая сброс настроек по умолчанию («Defaults») и просматривать информацию об инструменте.

Предварительный просмотр обеспечивает краткую предварительную визуализацию того, как будет восприниматься звук, если эффект будет применен с текущими настройками, без внесения фактических изменений в звук. Продолжительность предварительного просмотра определяется вашими настройками в меню Правка → Настройки → Воспроизведение, для которых предварительный просмотр по умолчанию составляет 6 секунд.

Кнопка «ОК» применяет эффект к выбранному звуку с текущими настройками эффекта.

Кнопка «Отменить» прерывает эффект и оставляет звук без изменений.

Рассмотрим пример применения штатной реверберации к аудиодорожке. Для наглядного примера воспользуемся предварительной установкой Вокал I и Вокал II¹. Этот метод использует встроенный, предва-

рительно установленный Вокал II. В качестве альтернативы, Вокал I производит более мягкую, менее заметную реверберацию. Для продвинутых пользователей любая предварительная установка может быть использована в качестве отправной точки для создания новых предварительных установок.

1. Убедитесь, что в конце режима аудио есть несколько секунд тишины, чтобы последняя нота могла исчезнуть естественным образом. При необходимости добавьте несколько секунд тишины в конец трека.
2. Выберите аудиотрек, содержащий запись вокала.
3. Нажмите «Эффект» → «Реверберация», чтобы запустить эффект реверберации.
4. На панели «Предварительные настройки» нажмите кнопку загрузки, выберите Вокал II из списка и нажмите кнопку «ОК». Теперь элементы управления будут настроены в соответствии с выбранной предварительной установкой.
5. Для применения эффекта к сигналу нажмите кнопку «ОК».

Если вокальная дорожка была в монорежиме, можно создать более «пространственный» эффект реверберации, превратив дорожку в «стереодорожку» перед применением эффекта реверберации, как описано в действиях 1) — 4) ниже.

Чтобы сохранить эффект реверберации на отдельной дорожке дублируем дорожку, к которой хотим применить реверберацию, затем применяем эффект реверберации к дублируемой дорожке, устанавливая режим «Только влажный» в диалоговом окне «Реверберации». После этого у вас появляется возможность управлять объёмом реверберации в конечном миксе, регулируя усиления дорожки на реверберируемой и исходной дорожках. Такой подход даёт вам возможность гибко настраивать пропорцию реверберации в любое время, после применения эффекта. Теперь схематично распишем действия в виде подробной инструкции. Для более длинного резонанса хвоста реверберации добавляем несколько (и более) секунд тишины в конец трека. Нажимаем на опцию создания «тишины». Выбираем продолжительность тишины (20 секунд — это оптимальное значение для реализации пышного хвоста реверберации, в стандартных случаях будет достаточно нескольких секунд 3–5 секунд), далее действия будут следующими.

1. Нажимаем кнопку «ОК», чтобы добавить тишину в конец трека.
2. Выбираем всю дорожку, нажав на пустое место на Панели управления дорожкой, или используем команду (SHIFT + J).

¹ Прим.автора. Всегда добавляйте несколько секунд тишины к любой дорожке, к которой вы собираетесь применить эффект реверберации. Хвост реверберации будет продолжаться в течение нескольких секунд

после окончания звука. Если вы не добавите тишину в конец трека, реверберация отключится неестественным образом.

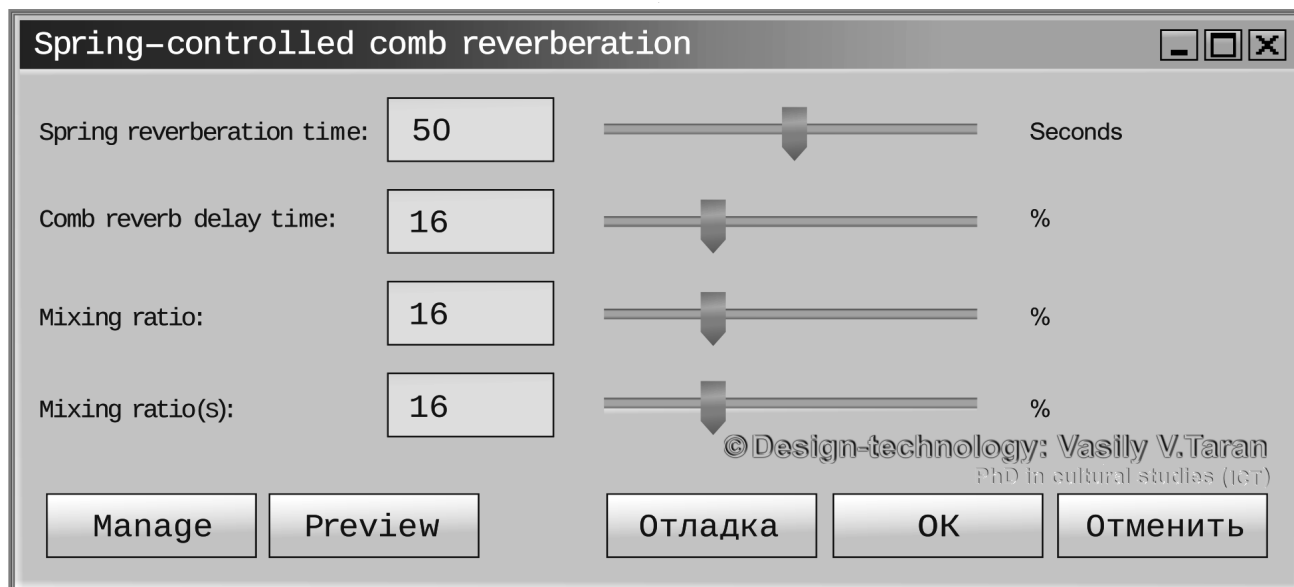


Рис. 5. Прототип модуля пружинно-гребенчатой реверберации.

3. Нажимаем опции: «Редактировать» → «Границы клипа» → «Присоединить», либо используем сочетание клавиш (CTRL + J).
4. Продублируем трек, выбрав трек, и вызовем опции: «Редактировать» → «Дублировать».
5. Выбираем дублирующийся трек на сонограмме.
6. Применяем эффект реверберации к дублирующемуся треку, убедившись, что режим «Только влажный» активен.
7. Воспроизводим треки, затем используем Регулятор усиления на треке с *реверберацией* (влажной) и на *исходном* треке, чтобы смикшировать желаемый объем результирующей реверберации. Убеждаемся, что комбинированное усиление достаточно низкое, чтобы не загорались красные индикаторы отсечения в конце полос воспроизведения измерителя громкости проигрывания аудиоматериала.

Чтобы применить стереореверберацию к монофонической дорожке¹, выполняем следующие действия.

8. Добавляем несколько секунд тишины в конец трека.
9. Продублируем трек, выбрав (трек) и нажав «Редактировать» → «Дублировать».
10. В раскрывающемся Меню звуковой дорожки на верхней дорожке выбираем «Создать стереотрек».

¹ Прим.автора. Часто необходимо добавить реверберацию к монотреку (например, к вокалу), но при этом добавить к реверберации стереофонический «разброс» для окраски сигнала. Как правило, звучание будет лучше, когда конечный микс представлен стереопанорамой.

11. Выбираем новый стереотрек.
12. Применяем эффект «Реверберация» к треку, выбрав *подходящее значение* для «Сtereo диапазона» и убедившись, что режим «Только влажный» не активен.

Данный метод можно изложить и в альтернативной редакции:

Добавляем несколько секунд тишины в конец трека. Дублируем трек (выбираем трек и нажимаем «Редактировать» → «Дублировать»).

13. Выбираем новый стереотрек.
14. Дублируем трек, выбрав трек и нажав «Редактировать» → «Дублировать».
15. Выбираем дубликат трека на сонограмме.
16. Применяем эффект реверберации к дублирующемуся треку, убедившись, что режим «Только влажный» находится в активном состоянии.
17. В раскрывающемся Меню аудиотрека *исходного* трека выбираем команду «Извлечь Моно из Сtereo».
18. Удаляем один из двух результирующих монотреков, нажав Кнопку закрытия трека или выбрав этот трек, затем выбрав «Треки» → «Удалить» треки (этот последний шаг не является строго необходимым, но уменьшает загромождение дорожки и даёт понять, что исходная дорожка монотрековая).

19. Воспроизводим треки. Затем, как и в первом примере, регулируем усиления реверберационных (влажных) и оригинальных треков, чтобы смикшировать желаемый объем реверберации, гарантируя, что результат не обрывается.

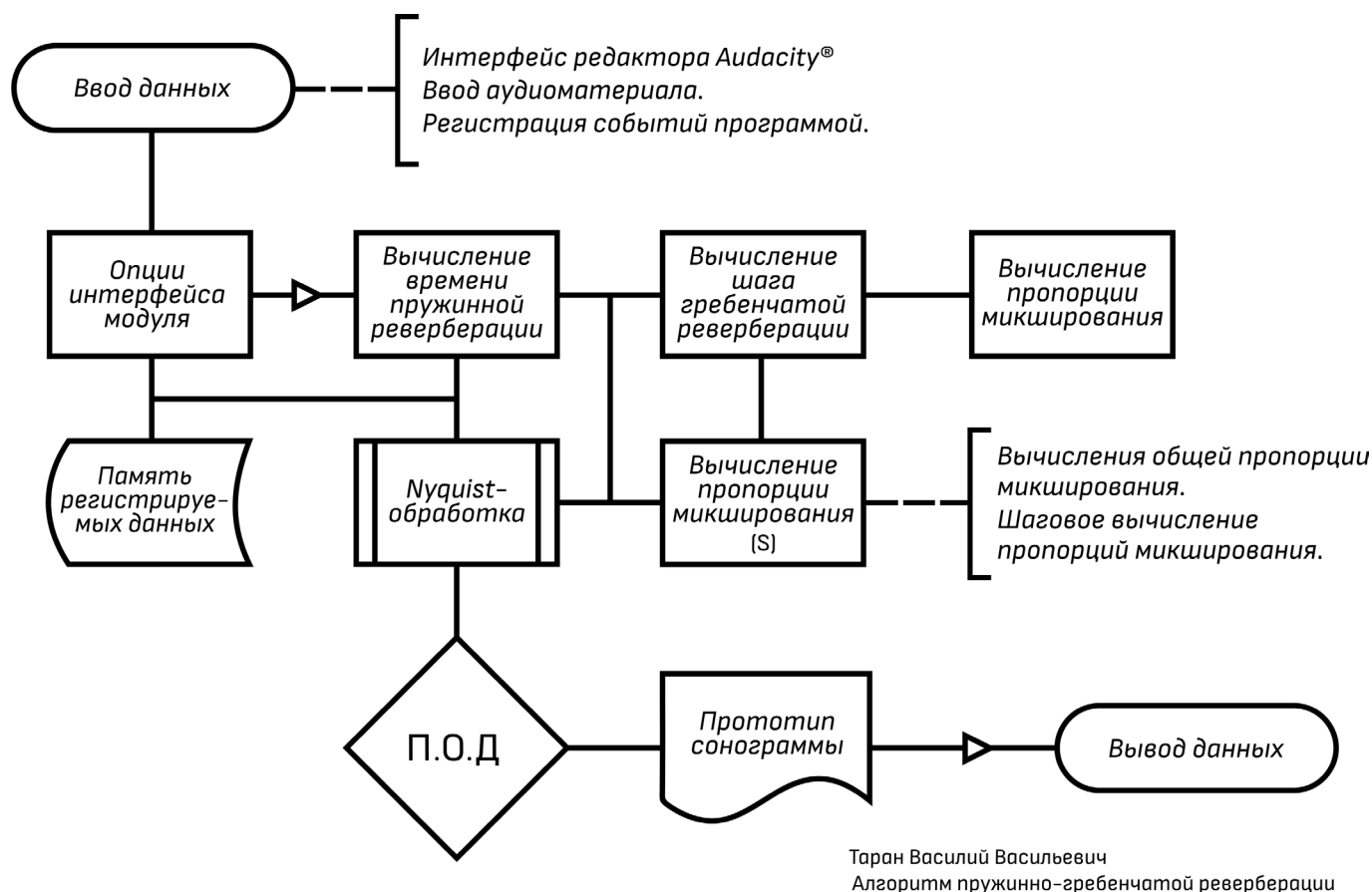


Рис. 6. Общая схема работы модуля пружинно-гребенчатой реверберации. Схема кратко описывает основные процедуры обработки аудиоматериала.

Как мы видим, перечень штатных функций модуля «Реверберация» довольно внушителен, однако, о чём уже говорилось ранее, он может быть недостаточен при обработке данных большой сложности там, где требуются *нестандартные решения*, поставленных перед оператором звука задач. И здесь стоит обратить внимание на язык Nyquist.

Nyquist можно отнести к узкоспециализированным языкам программирования, функционал которых призван сопровождать процессы обработки аудиоматериала и обеспечивать надёжные технические решения при нестандартных манипуляциях со звуком. Иначе говоря, это инженерный язык, дополняющий визуально-ориентированную опциональную базу штатных средств редактора Audacity®, направленных на профессиональное редактирование аудиоданных.

Говоря о расширении границ штатных средств Audacity®, будет уместным привести пример программирования на языке Nyquist модуля, который бы имитировал реверберацию с возможностью различных специализированных настроек. Допустим, в наши задачи

входит создание эффекта пружинно-гребенчатой стерео реверберации с ротацией сигнала из канала в канал¹.

Пружинно-гребенчатая реверберация — это достаточно редкий, но многофункциональный эффект, который может применяться к различным физическим источникам звука, которые записаны и распределены по дорожкам аудиоредактора.

Пружинно-гребенчатая реверберация хорошо применима к вокалу, струнным и духовым инструментам,

¹ Прим.автора. Пружинно-гребенчатая реверберация — это акустический эффект, который направлен на формирование у слушателя специальной психоакустической картины, в которой описывается непрерывное постоянное удаление объекта от источника звука, с одновременным подмешиванием в звуковую волну гармонических колебаний, выражающихся в интервальных плавных нарастаниях сигнала. Эффект может использоваться в качестве дизайн шаблона при оформлении аудиокomпозиций, в кибермедицине — для проверки и тестирования различных аппаратов, направленных на улучшение человеческого слуха (включая аппараты искусственного преобразования речи в синаптические сигналы), а также в научных целях — для экспериментов в области акустической физики (исследования звука в замкнутом и незамкнутом пространствах, генерации шума, исследования волновых процессов).



Пространство №1

Пространство №2

Пространство №3

Рис. 7. Смоделированные в программе Blender™ геометрические пространства замкнутого типа, внутри которых может быть успешно генерирована волна пружинно-гребенчатой реверберации.



Рис. 8. Пространство № 1, выбранное нами для проведения акустических экспериментов над реверберацией звука.

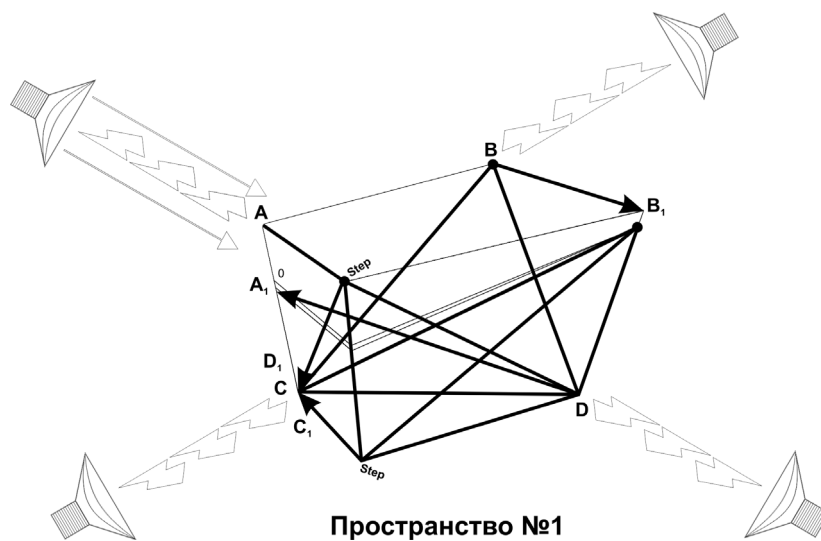


Рис. 9. Схема траектории распространения аудиосигнала в замкнутом пространстве.

и используется (*вполне применима*) для создания уникального акустического дизайна аудиодорожек.

Также пружинно-гребенчатая реверберация может быть полезна при реставрационных процедурах аудиоматериала и его дальнейшем мультисессионном премастеринге, наглядным примером здесь могут служить процессы, связанные с дереверберацией звука, то есть, когда необходимо убрать излишние хвосты реверберации из аудиосигнала либо (по возможности) заглушить реверберацию полностью. В подобных случаях пружинно-гребенчатая реверберация может быть использована в качестве «снимка» — образца для извлечения из панорамы аудиосигнала ненужных артефактов.

Для того, чтобы воспроизвести подобный эффект в редакторе Audacity®, нужно создать Nyquist-модуль, который описывал бы алгоритм такого типа реверберации. В первую очередь на данном этапе необходимо создать прототип Nyquist-модуля, который бы вобрал в себя все технические инструменты по настройке и применению к аудиоматериалу пружинно-гребенчатой реверберации (рисунок 5).

На прототипе показано, что основными инструментами управления модулем являются фейдеры (бегунки), регулирующие общие пропорции смешивания сигнала, применяемые к миксу. Общая схема работы, составленного нами алгоритма, будет выглядеть следующим образом (рисунок 6).

Сначала происходит ввод данных — это аудиоматериал, который изначально уже предзагружен в редактор Audacity® и отображается в виде сонограммы в его интерфейсе (программа регистрирует события). Затем производится ввод команд через приглашение Nyquist (Nyquist Prompt).

Написанный нами сценарий вызывает опции интерфейса модуля, которые передают управление сценарию вычисления времени пружинной реверберации (вычисляется время наивысшего шага задержки сигнала, измеряемого в секундах).

После этого, программа (когда оператор процесса обработки данных определится со временем пружинной реверберации) выполнит маршрутизацию по отношению к шагу гребенчатой реверберации (устанавливается процентное соотношение времени шага гребенчатой реверберации). Одновременно с этим процессом устанавливается соединение со сценарием общего процесса микширования аудиосигнала (интерфейс модуля регистрирует данные).

Далее предлагается зарезервировать значения шаговых пропорций микширования аудиосигнала (опция устанавливает шаговую пропорцию хвоста гребня реверберации). Затем все действия обрабатываются языком Nyquist и превращаются в пропорционально обработанные данные (П.О.Д.), которые передаются редактору Audacity® для применения пропорций к аудиоматериалу, а результирующие значения преобразовываются в обновлённые данные, которые формируют документ и обновляют графическое содержимое сонограммы. После применения данной процедуры, аудиосигнал будет насыщен данным типом реверберации.

Пружинно-гребенчатую реверберацию проще всего представить в виде замкнутого пространства, имеющего нестандартную геометрию. Для этих целей автор сгенерировал в программе Blender™ несколько нестандартных геометрических конструкций (виртуальных пространств), внутри которых отклик данного типа реверберации звучал бы более насыщенно (рисунок 7).

Итак, перед нами три пространства, имеющие схожую геометрию, но отличающихся между собой углами развёртки. Пространство № 1 — характеризуется наивысшей степенью отклика пружинной гребёнки. Пространство № 2 — олицетворяет плотную пружинно-гребенчатую реверберацию, при которой её хвост будет превышать уровень плотности отклика полезного сигнала. Пространство № 3 — характеризует рассеянную пружинно-гребенчатую реверберацию, отклик которой, при перемещении сигнала от угла одной стенки к другой, будет рассыпчатым.

Больше всего представляет интерес пространство № 1, поскольку его геометрия позволяет нам прочувствовать весь спектр гармоник данного типа реверберации.

Рассмотрим модель пространства № 1, которая обеспечивает самый чувствительный уровень отклика пружинной гребёнки.

Примечание: Рисунок демонстрирует распределение откликов аудиосигнала в геометрическом пространстве № 1.

Уровень отклика пружинной гребёнки определяется оператором обработки звука. Поскольку мы имитируем процесс пружинно-гребенчатой реверберации (в целях прозрачности и понимания применения подобных процессов), автор данного мини исследования вправе самостоятельно определить траекторию распространения аудиосигнала внутри разработанного нами пространства.

$$\begin{aligned}
 &1) A = C(D_1 + B_1) + (B, C \in \text{step}_1) + D \downarrow D_1 \Rightarrow \{C_1 + B_1\} \\
 &\quad (C[\downarrow; (D_1, C_1)] \leftrightarrow \text{step}_1[B] \rightarrow (C = C)) \\
 &2) B = D(C_1 + D_1) + (D, A \in \text{step}_2) + A \downarrow A_1 \Rightarrow \{C_1 + D_1\} \\
 &\quad (D[\downarrow; (D_1, A_1)] \leftrightarrow \text{step}_2[C] \rightarrow (D = D)) \\
 &3) C = B(D_1 + C_1) + (A, C \in \text{step}_3) + B \downarrow A_1 \Rightarrow \{C_1 + A_1\} \\
 &\quad (C[\downarrow; (B_1, A_1)] \leftrightarrow \text{step}_3[D] \rightarrow (C = C)) \\
 &4) D = A(C_1 + D_1) + (C, D \in \text{step}_4) + C \downarrow B_1 \Rightarrow \{C_1 + B_1\} \\
 &\quad (D[\downarrow; (D_1, C_1)] \leftrightarrow \text{step}_4[B] \rightarrow (D = D))
 \end{aligned}$$

Рис. 10. Формулы, определяющие маршрутизацию сигнала в геометрическом пространстве, по рисунку 7 (пространство № 1) и рисунку 9 (пространство № 1).

На рисунке № 9 представлена схема траектории распространения аудиосигнала в замкнутом пространстве. Источник звука выведен за границы схемы для удобства иллюстрации направления аудиопотока. Вхождение аудиосигнала в акустическое пространство, представленной геометрической конструкции, осуществляется с левой стороны в верхнем углу и обозначается точкой «А». Динамик, расположившийся над позицией входной точки, двумя стрелками показывает направление ввода сигнала¹. Аудиосигнал, попадая

в замкнутую конструкцию, начинает отражаться от стенок замкнутого пространства. Цикл отражения обозначается стрелками и выделен жирными линиями.

Радиальными элементами размечены углы соединения стенок виртуального пространства, места их рассредоточения указывают на точки диффузии моделируемого пружинного отклика с откликом гребёнки. Отметка «0» указывает на то, что замкнутое пространство имеет небольшой регулируемый допуск выхода аудиосигнала за его пределы.

Поскольку мы занимаемся имитационным моделированием распространения аудиосигнала, такой допуск необходим для наглядного представления о подмешивании в чистый сигнал артефактов реверберации. step^2 — обозначает шаг гребенчатой реверберации.

¹ Прим.автора. Основным источником звука в нашей схеме является динамик над точкой «А», поток аудиоинформации, следующий от динамика к точке «А», обличён двумя стрелками, которые показывают направление стереофонического сигнала внутри смоделированного геометрического пространства. На иллюстрациях (рисунки № 8,9) динамики расположены с внешней стороны, хотя пространство является замкнутым. Это сделано для наглядности понимания процесса, поскольку объём дизайн-элементов внутри разработанной автором конструкции, утяжелял бы её и не позволял воспринимать её целостно. Во всех спроектированных автором пространствах, присутствует небольшой надрез (в виде периферической серой линии), горизонтально разделяющий каркас трёхмерного пространства на две части. Это сделано для того, чтобы читатель обратил внимание на возможность настройки данного алгоритма в незамкнутом пространстве. Конечно, в реальных (физических) условиях ставить подобные эксперименты с динамиками и «сундуком» — назовём нашу геометрическую конструкцию именно так, поскольку она его очень напоминает, было бы хлопотно и проблематично. Особенно, если отделять верхнее пространство (шапку) от основного каркаса конструкции по градусной траектории, синхронно перемещая динамик (динамики — если необходима многопоточность стерео → квадро) по соответствующей шапке траектории отклонения. Однако программно-вычислительные эксперименты позволяют нам имитировать такого рода процессы, осуществляя процедуры обработки аудиосигнала в виртуальном пространстве. Автору важно показать своему потенциальному читателю, что эксперименты в данном направлении можно и нужно развивать, а действия, направленные на моделирование подобных процессов, могут иметь многовекторный характер. Для компьютерного аудиоинженера или программиста по звуку, либо же просто талантливого экспериментатора, такой подход откроет новые грани в возможностях по моделированию физико-акустических процессов с применением такого универсального (открытого) программного средства, как Audacity®.

Хочется отметить ещё один важный фактор (условие) для моделирования — это материал самой конструкции, где должен воспроизводиться звук. В данном случае (по умолчанию) мы моделируем металлический тип реверберации (т.е. стенки нашего виртуального пространства как бы покрыты металлом), это объясняется тем, что металл имеет более громкий отклик, способствующий резонансному отражению звука, что для пружинной реверберации, имеющей гребенчатый хвост, очень важно. Материалы для условий моделирования могут быть разными: различные полимеры, дерево, металлы разного вида, бетонно-каменные плиты. Под каждый из этих процессов можно задать свой алгоритм отклика моделируемой реверберации, для этого достаточно иметь образец частотных характеристик звука, отражаемых в замкнутых пространствах с отличающимися друг от друга элементами покрытия стен.

² Прим.автора. В соответствии с правилами русского языка после точки, следующее предложение начинается с большой буквы. Но в нашей статье приходится отступить от установленных филологических правил и писать имена функций, операторов и прочих управляющих элементов в области программирования согласно синтаксисам программно-технических языков (в нашем случае с маленькой буквы). Поскольку такие языки часто бывают регистрозависимыми — отступление от написания их элементов может привести к ошибкам, и путанице при воспроизведении и отладке программного кода на вычислительных устройствах.

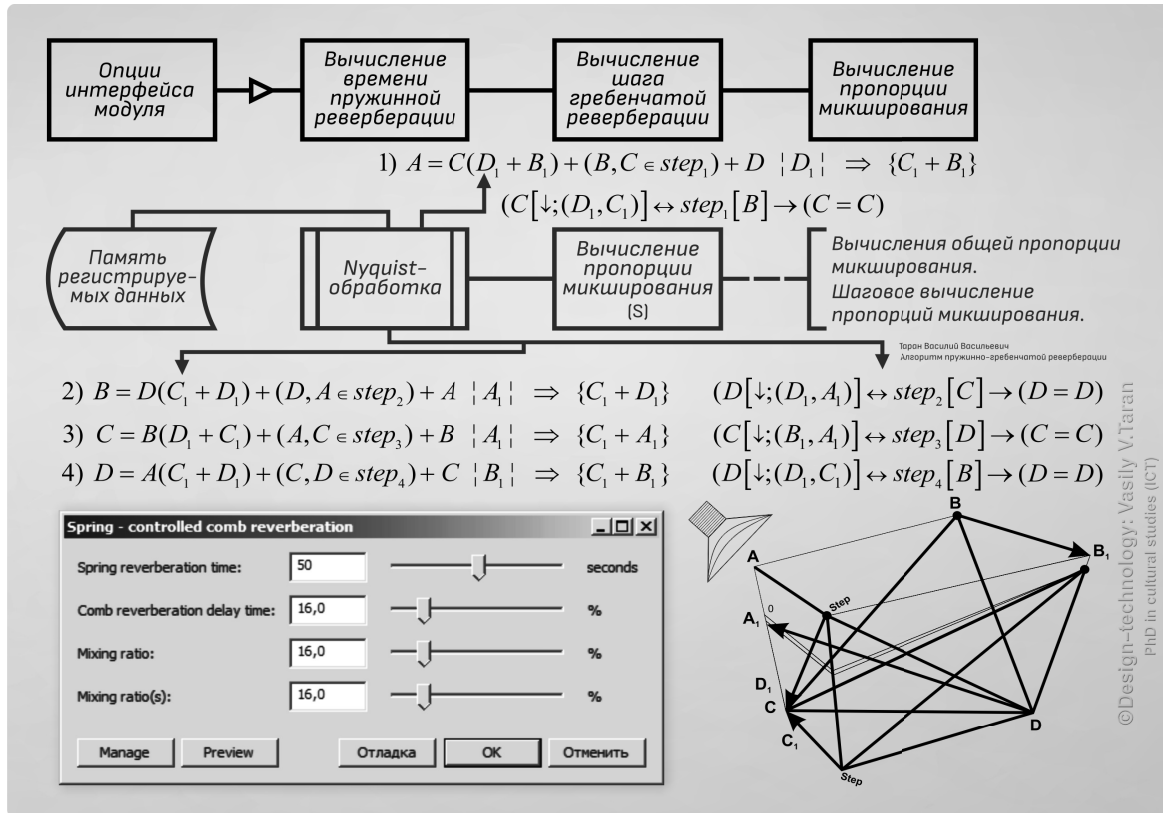


Рис. 11. Фрагментарная, общая схема обработки аудиоданных, демонстрирующая заданную оператором геометрию отклика для имитации виртуального пространства.

Точки A, B, C, D — обозначают проход сигнала в чистом виде без артефактов моделируемого нами эффекта реверберации.

Точки A_1, B_1, C_1, D_1 — обозначают первичные характеристики отражательной способности акустического пространства.

Итак, мы имеем акустическое пространство, в котором пружинно-гребенчатая реверберация может распространяться в различных вариациях. В целях экономии места и в соответствии с жанром изложения научного материала (научная статья), остановимся на одной из вариаций, которая будет имитировать объёмную пружинно-гребенчатую реверберацию, узлы которой будут обеспечивать плавную взаимодиффузию для обеспечения эффекта скользящего звука¹. Назовём её скользящей пружинно-гребенчатой реверберацией.

¹ Прим.автора. В имитационных процессах реверберации эффект скользящего звука характеризуется как кратковременное эхо, отклик которого осуществляет вторичный плавный переход между первичным источником сигнала и его отражателем. Кратковременное отражение может быть применено к аудиодорожке многократно, таким образом человеческий слух начинает воспринимать воспроизводимый эффект как скользящее импульсивное эхо, реверберация которого

Для достижения качественного эффекта скользящей пружинно-гребенчатой реверберации, с физической точки зрения нужны правильно расставленные *отражающие границы* внутри замкнутого пространства. Минимальный сдвиг таких границ способен *изменить траекторию отклика звука* и привести к более *коротким его затуханиям* на определённых участках, что может заметно сказаться на воспроизводимом эффекте. Воспроизведение эксперимента является одной из важнейших составляющих естественнонаучных дисциплин, компьютерные науки здесь не являются исключением.

будет иметь хвост, который плавно будет переходить от одного отражателя к другому. Отражателями здесь могут являться любые атрибуты акустического пространства (углы комнаты, стенки комнаты, и т.д.). Степень насыщенности эффекта здесь может зависеть от материалов, из которых сделан отражатель, обычно такой материал делится на две группы: звукопоглощающий материал и материал, имеющий высокую степень отражающей способности. В качестве звукопоглощающего материала можно привести в пример полиуретан, а в качестве материала, которому присущи отражательные характеристики, может являться железо. Для эффекта скользящего эхо подходит второй тип материала. Чтобы имитировать звук «классического железа», необходимо увеличить плотность взаимодиффузии на узлах, которые располагаются перпендикулярно друг другу на участке цепи всей реверберационной схемы.

```

set sprs-sum += ch[t] * s  —> set sprs-sum += ch[m] * s
set total-sprs += ch[t]   —> set total-sprs += ch[m]
    
```

Рис. 12. Фрагмент программного кода, демонстрирующий переключение таблицы одного из каналов в монорежим.

```

1. ;; создание (нового) левого субканала
2. (defun tab (slabel s1)
3.   (let((s1 (s1 leftch 300))
4.         (s2 (feedback-delay(s1(hz start 500)900)(* s1 0.1) 0))
5.         (s3 (s1(start left 800)1200))
6.         (s4 (feedback-delay(s1(s4 left 1200)2400)(* s1 0.08) 0))
7.         (s1 (s1 left 2400)))
8.     (sim s1 s2 s3 s4 s1)))

                                     ©Design-technology: Vasily V.Taran
                                     PhD in cultural studies (ICT)

1. ;; левый субканал
2. (unite sp (acc 0 vol) (col 0 snp))
3. (unite sp (acc 1 vol) (col 1 snp))
4. (unite sp (acc 2 vol) (col 2 snp))
5. (unite sp (acc 3 vol) (col 3 snp))
6. (unite sp (acc 4 vol) (col 4 snp)))
    
```

Рис. 13. Фрагмент программного кода для обработки данных на одном из каналов.

Поэтому для постановки нашего эксперимента требуется установить границы его исполнения. Такие границы мы определим с помощью схематических рекурсивных формул, адресно показывающих маршрутизацию отклика сигнала в построенном нами геометрическом пространстве (рисунок 10).

Рекурсивные формулы демонстрируют каналы распространения звукового сигнала в замкнутом пространстве. Буквенные обозначения, это точки его отражения. Точками отражения мы считаем углы замкнутого пространства геометрической конструкции. Они же образуют узлы кратковременных откликов, характеризующих реверберацию.

Отклики каждого из приведённых выше узлов, образуют колорит звучания реверберации, создают её акустический образ. Взаимоотношения узлов характеризуются шагами, обеспечивающими проход сигнала на определённом отрезке пути. Наша схема реверберации состоит из четырёх шагов. На примере первого шага расшифруем последовательность действия нашего алгоритма.

$$A = C(D_1 + B_1) + (B, C \in step_1) + D \{D_1\} \Rightarrow \{C_1 + B_1\}$$

Первая часть формулы $A = C(D_1 + B_1)$ — определяет влияние звукового потока на точку C . $(D_1 + B_1)$ — образуют узлы первичной свёртки реверберации. $(B, C \in step_1)$ — показывают принадлежность опорных точек реверберации к шагу (шаг 1). $D \{D_1\} \Rightarrow \{C_1 + B_1\}$ — переход сигнала к результирующей фазе затухания¹. С помощью языка Nyquist попробуем представить нашу схему в действии и вывести её основные параметры в виде плагина. Приводим значения, в соответствие с прототипом и схемой (рисунки № 5, № 6), по которой настраивается алгоритм виртуализации пространства (рисунок 11).

Как видно из рисунка № 11, блок «Nyquist-обработка» отвечает за выполнение инструкций, изложенных на языке Nyquist. Для того чтобы алгоритм адекватно обрабатывал реверберацию в режиме стерео, пред-

¹ Прим.автора. Результирующая фаза затухания определяет характер перемещения дробного сигнала, т.е. сигнала который был перераспределён в первом участке прохождения. Результирующая фаза состоит из четырёх шагов A, B, C, D, каждый из которых имеет свою оригинальную карту прохода. Опорной точкой карты прохода является буквенное обозначение, занесённое в «{,}_». После стрелки, следующие буквенные значения в фигурных скобках обозначают обратный проход сигнала, формируемый в результате его отталкивания от точек первоначальной ретрансляции.

```

;nyquist plug-in
;version 1.0
;name "Spring - controlled comb reverberation"
;test-version 2022

;control time "Spring reverberation time" real "seconds" 2.5 0.1 10
;control damping "Comb reverberation delay time" real "%" 50 0 100
;control mix "Mixing ratio" real "%" 20 0 100
;control mix "Mixing ratio(s)" real "%" 20 0 100

;; создание (нового) левого субканала
(defun tab (slabel s1)
  (let((s1 (s1 leftch 300))
        (s2 (feedback-delay(s1(hz start 500)900)(* s1 0.1) 0))
        (s3 (s1(start left 800)1200))
        (s4 (feedback-delay(s1(s4 left 1200)2400)(* s1 0.08) 0))
        (s1 (s1 left 2400)))
    (sim s1 s2 s3 s4 s1)))

;; создание (нового) правого субканала
(defun tab (slabel s1)
  (let((s1 (s1 leftch 300))
        (s2 (feedback-delay(s1(hz start 500)900)(* s1 0.1) 0))
        (s3 (s1(start right 800)1200))
        (s4 (feedback-delay(s1(s4 right 1200)2400)(* s1 0.08) 0))
        (s1 (s1 right 2400)))
    (sim s1 s2 s3 s4 s1)))

  set sprs-sum += ch[t] * s
  set total-sprs += ch[t]
;; левый субканал
(unite sp (acc 0 vol) (col 0 snp))
(unite sp (acc 1 vol) (col 1 snp))
(unite sp (acc 2 vol) (col 2 snp))
(unite sp (acc 3 vol) (col 3 snp))
(unite sp (acc 4 vol) (col 4 snp)))

  set sprs-sum += ch[t] * s
  set total-sprs += ch[t]
;; правый субканал
(unite sp (acc 0 vol) (col 0 snp))
(unite sp (acc 1 vol) (col 1 snp))
(unite sp (acc 2 vol) (col 2 snp))
(unite sp (acc 3 vol) (col 3 snp))
(unite sp (acc 4 vol) (col 4 snp)))

```

©Design-technology: Vasily V.Taran
PhD in cultural studies (ICT)

Рис. 14. Полная структура программного кода алгоритма пружинно-гребенчатой реверберации.

усмотрены две таблицы, регулирующие хвост пружинно-гребенчатой реверберации, который будет отражаться на каждом из двух каналов. Если необходимо провести обработку аудиоматериала в режиме моно, то можно просто отключить таблицу одного из каналов, поменяв значение [t] на [m] как показано на рисунке 12, либо воспользоваться нулём.

Ниже приведён фрагмент программного кода, осуществляющий обработку данных одного из каналов (рисунок 13).

Здесь символьная функция `defun`¹ (для подстановки значений можно использовать формальный список параметров — список лямбды) определяет фрагменты значения таблицы левого субканала, её можно использовать для определения других функций, также как в Common Lisp [13,14]. Субканал — это виртуальный канал, который создаёт огибающие для физического канала аудиоданных. На субканале происходит приращение предварительно обработанного сигнала к полезному

¹ Прим. автора. Рисунок № 13. (в списке вторая строчка, первый блок).

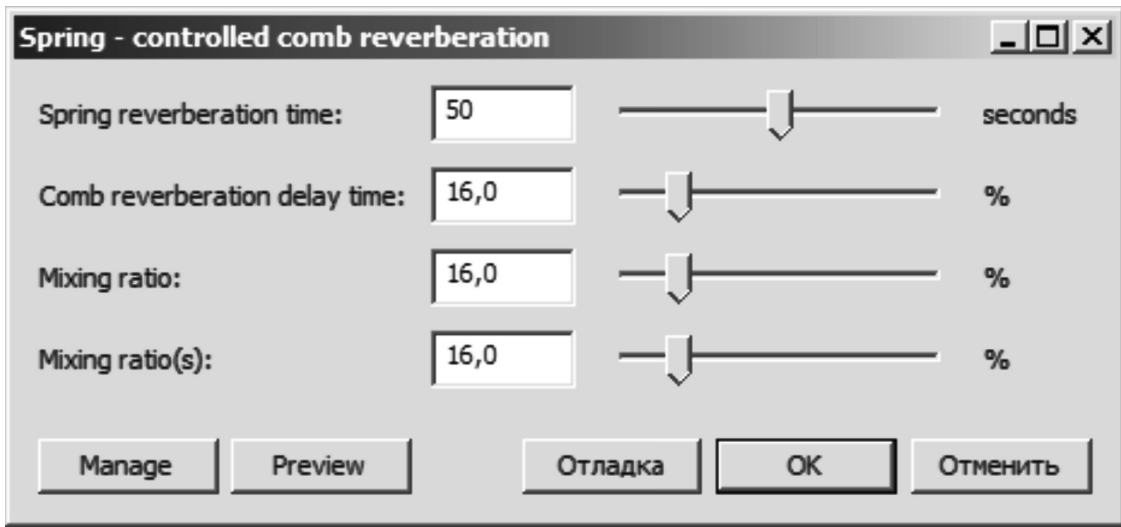


Рис. 15. Модуль обработки аудиоматериала (плагин), имитирующий эффект пружинно-гребенчатой реверберации.

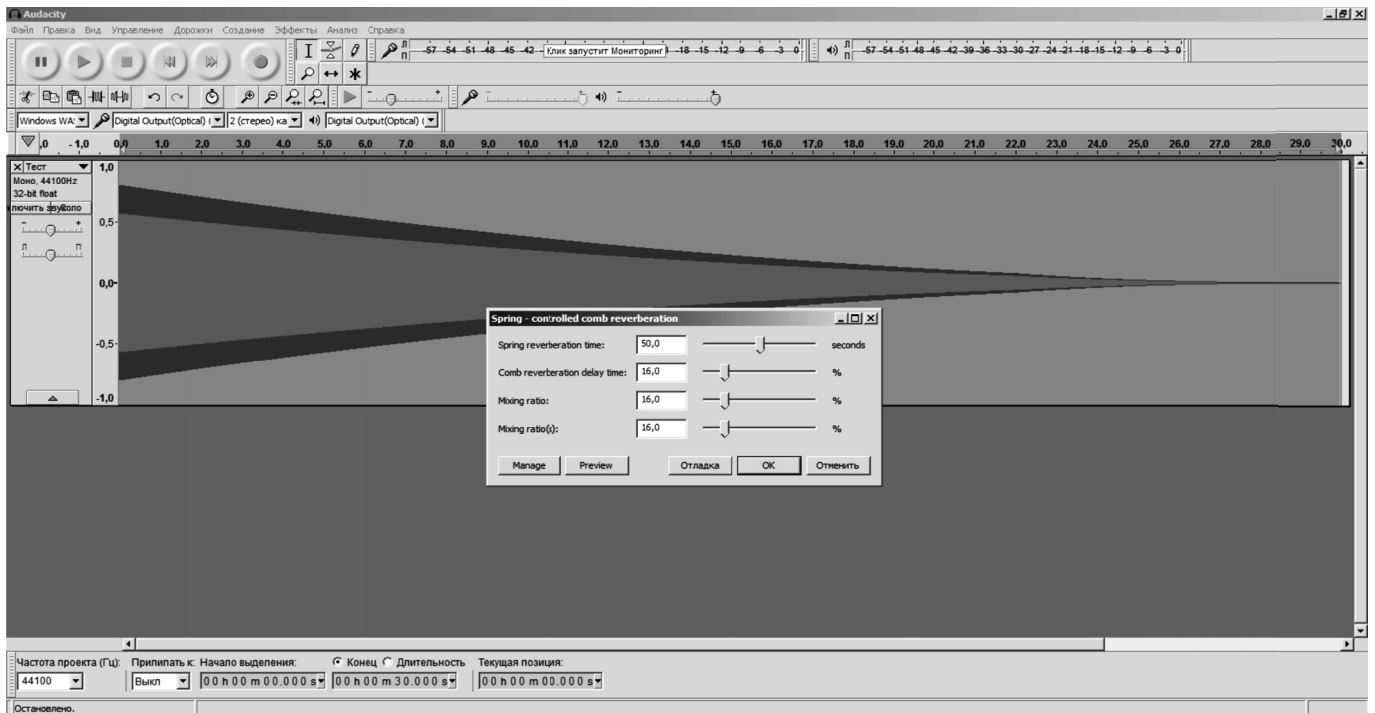


Рис. 16. Работа компьютерного редактора Audacity® с модулем эффекта пружинно-гребенчатой реверберации.

сигналу. Два субканала (левый и правый) образуют стереосигнал, каждый из них по отдельности будет являться моносигналом. slabel — создаёт временный маркер, отмечающий точку диффузии между оригинальным и предварительно обработанным аудиосигналами, где s1 — граница отклика В, С, которые принадлежат перво-

му проходу вектора реверберации ($step_1$). Контрольная конструкция let¹ устанавливает локальную связь leftch с обратным откликом (хвостом) пружинно-гребенчатой реверберации, и в левом, и в правом каналах сохра-

¹ Прим. автора. Рисунок № 13. (в списке третья строчка, первый блок).

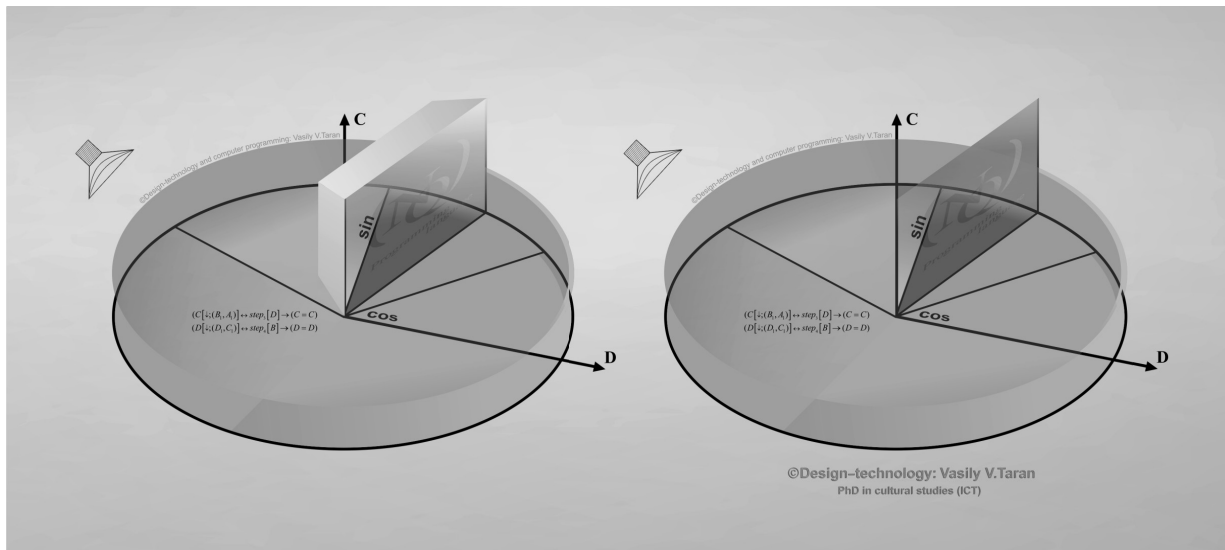


Рис. 17. Горизонтальная ротация эффекта реверберации, переходящая из одного канала в другой (по типу эффекта Лесли).

няется привязка leftch¹. feedback-delay — производит задержку обратной связи от границы отклика (в нашем случае s1=500 Гц, с обратным микрооткликом s1 0.1.). sim² — характеризует структуру времени на точках прохождения эффекта реверберации A, B, C, D. Данные таблицы³ и её заголовок пересекаются с меткой стерео-образа (leftch) и образуют следующую цепочку

unite sp → acc 0 vol → col 0 snp

Объединение сообщения, выполняющего цикл по траектории заданного нами алгоритма unite (square post) и передача этого сообщения сигналом acc⁴ в нижеследующую часть таблицы происходит поочередно, поскольку при этом заголовок выполняет функцию предпрокачки сигнала. Объем сообщения будет нулевым, также как и индекс заголовка прохода. До перехода к следующим позициям таблицы происходит надрез

¹ Прим. автора. leftch — это метка стерео образа левого канала, максимальное значение растяжения установлено в 300 мс. Это означает, что при одновременном воспроизведении двух каналов (и соответствующих им субканалов — стерео) источником извлечения центральной фазы будет выступать именно левый канал. Таким образом, на правый канал распространяется стереобаза левого канала (при воспроизведении взаимосвязь между субканалами превращается во взаимосвязь физических каналов), а хвост реверберации (при растяжении стереополя) может быть вписан в правый канал (см. рисунок № 11 — создание правого субканала).

² Прим. автора. Рисунок № 13. (в списке восьмая строчка, первый блок).

³ Прим. автора. Рисунок № 13. (в списке вторая строчка, второй блок).

⁴ Прим. автора. Сигнал «acc» — (Accordion), обозначает передачу сообщения ступенчато, с визуальной точки зрения это похоже на элемент конструкции меха аккордеона. Сигнал, распространяемый таким способом, легко возвратим к исходным позициям.

(рассечение) сигнала на несколько частей с целью дублирования одних и тех же данных на разные потоки, в том числе и на потоки обратной связи. Значение sp отвечает за разделение сигнала и передачу его содержания от таблицы к субканалам. Значения по вертикали — это точки отклика аудиоматериала, эти же значения присутствуют и по горизонтали в левом и правом субканалах, отклик внутри каналов имеет структуру времени, измеряемую в микросекундах.

Приведём полную структуру программного кода нашего алгоритма (рисунок 14)

Интерфейс нашего модуля будет выглядеть следующим образом (рисунок 15).

Наш модуль при обработке предгенерированного аудиосигнала в редакторе Audacity® (рисунок 16).

В целях придания реверберации дополнительных артефактов, формирующих *пространственные эффекты*, можно поэкспериментировать с ротацией реверберации по заданному оператором алгоритмическому образцу.

Для того чтобы выполнить ротацию звукового потока хвоста пружинно-гребенчатой реверберации и сформировать её оригинальный алгоритмический рисунок, как показано ниже (рисунок 17), необходимо задать следующие параметры.

В нашем случае ротация будет происходить по двум фрагментам, используемых нами рекурсивных формул.

```

1. | (let ((d rotate) (max (+ d range)))
2. |   (make-symbol (< d max)
3. |   (if (<= (aref s d) diffphase)
4. |     (if (> (aref r (+ d 5)) diffphase)
5. |       (setq range (cons (+ d 5) phase))))
6. |   (if (<= (aref s (+ d 5)) phase)
7. |     (setq leftch(cons d phase))))
8. |   (setq d (5+ d)))) ©Design-technology: Vasily V.Taran
                          PhD in cultural studies (ICT)

1. | (unite sp (acc 3 vol) (col 3 snp))
2. | (unite sp (acc 4 vol) (col 4 snp))
2. | (unite sp (acc 4 vol) (col 4 snp))
1. | (unite sp (acc 3 vol) (col 3 snp))
    
```

Рис. 18. Портируемый фрагмент программного кода, осуществляющий ротационные функции согласно рисунку 17.

$$\begin{aligned}
 (C[\downarrow; (B_1, A_1)] \leftrightarrow step_3 [D] \rightarrow (C = C)) \\
 (D[\downarrow; (D_1, C_1)] \leftrightarrow step_4 [B] \rightarrow (D = D))
 \end{aligned}$$

Использование *третьего* и *четвёртого* шагов ($step_3 \rightarrow step_3$) объясняется направленностью реверберационного потока.

Первые опорные точки A, B (которые, по сути, являются физическими каналами) — по которым аудиосигнал проходит в чистом виде, используют левый канал в качестве извлекаемой стереобазы.

Генерация стереобазы, а также *гребень* хвоста пружинной реверберации происходят на субканалах A_l и B_l , поэтому период затухания отклика будет: $C = B_l, A_l$ (leftch) и $D = D_l, C_l$ (leftch). Именно это затухание¹ зарезервировано для нашего фазовращателя².

¹ Прим.автора. В период затухания кратковременного отклика аудиосигнала, для поддержания его чёткости и ясности воспроизведения (особенно на материале со смешанным содержанием), важен баланс стереополя всей обрабатываемой композиции. Поскольку мы производим чистый эксперимент и работаем только с программной оболочкой нашего модуля, то мы не используем сторонних плагинов, расширяющих стереополе, более того мы не задействуем и интерфейсно-ориентированную базу Audacity®, к примеру её удачную реализацию стереокомпрессии и расширения каналов стереобазы SC4. Мы задействуем только средства программного кода Nyquist, поэтому подчёркивание затухания отклика, который физически перемещается из канала в канал, происходит за счёт установленного количества проходов обработки участков отклика, на выделенном оператором хронометраже (всего 5 проходов).

² Прим.автора. Имеется в виду алгоритм вращения фазы аудиосигнала, формирующий эффект перехода воспроизводимого звука из одного канала в другой.

Вращение будет иметь ротационную форму согласно рисунку 17. Ячейки таблицы будут немного изменены, поскольку ротация будет переходить из одного субканала в другой. Приводим программный код данной процедуры (рисунки 17).

Здесь let^3 — контрольная конструкция, которая осуществляет локальную связь с точкой ротации «d», при этом max^4 — арифметическая функция, определяющая самое большое из списка чисел в диапазоне точки ротации [15,16]. $make-symbol^5$ — символьная функция, создающая неинтернированный символ⁶ в точке ревербе-

³ Прим.автора. let устанавливает последовательную привязку для переменных, каждая из которых может быть либо символом, который инициализируется как nil, либо списком, корнетка которого является символом (в данном случае cadr будет выступать выражением инициализации). $expr$ — выражения которые будут оцениваться далее. $returns$ — значение последнего выражения.

⁴ Прим.автора. max — осуществляет определение самого большого числа из списка чисел. Как правило, это стартовые числа. Работает в паре с операторами $expr$ и $returns$. При совместной работе, $expr$ — это выражения, подлежащие проверке, а $returns$ — это наибольшее число в списке.

⁵ Прим.автора. $make-symbol$ на языках SAL, LISP и Nyquist создаёт неинтернированный символ. Работает с операторами $rname$ и $returns$. $rname$ — строка имени печати символа, $returns$ — новый символ.

⁶ Прим.автора. Неинтернированный символ — это символ, используемый просто «как объект данных», без специальной каталогизации (он не принадлежит ни к какому конкретному пакету). Неинтернированный символ обозначается символом «#:», за которым следует его печатное название. Напротив — (интернированный символ) — это символ, который индексируется по его печатному наименованию в каталоге, называемом пакетом. Запрос на поиск символа с таким именем для печати приводит к тому же символу «ef» (проверка на идентичность выражения). Каждый раз, когда входные данные считываются с помощью функции «read» (осуществляет считывание аудиоданных, в режиме потока работает с переменной «fr», которая инициализирует дополнительные

рации, для оценки совокупности максимального числа из списка¹. `if2` — выполняет тестирование оценки выражения, лежащего в области функции массива (`aref`), связка вводится для оценки уровня диффузии реверберации на фазе субканала к которому применяется ротация [17]. `aref3` — функция массива, получает элемент массива для шины стереорасширения. `setq4` — символьная функция, осуществляет установку значения символа [18]. `cons5` — функция списка, осуществляет создание нового узла списка для точки ротации на фазе канала [19].

Чтобы поэкспериментировать с реверберацией и разобраться в опциональной настройке Nyquist-модулей, приведём программный код обычной реверберации, изложенный на языке LISP (приложение № 1), с дальнейшей его имплементацией на языке Nyquist (приложение № 2)⁶. Структура нижеприведенного про-

параметры для считывания) и появляется это имя печати, оно считывается как один и тот же символ. Это свойство символов делает их подходящими для использования в качестве имён для вещей и в качестве специализированных крючков, на которые можно цеплять постоянные объекты данных (например, используя список свойств).

¹ Прим.автора. Оценка совокупности приведёт к чтению первого выражения от файла «init.lisp». Выражение будет возвращено как результат функции «`read`». Больше выражений может быть считано из файла, используя дальнейшие обращения к функции «`read`». Когда данные заканчиваются, чтобы осуществлять дальнейшее чтение, функция «`read`» возвратится значением «`nil`».

² Прим.автора. `IF` — изначально оператор языка SAL, введённый в Nyquist для тестирования оценки выражений (`test-expr`), если выражение тестируется как истина, то используется оператор (`true-stmt`), если ложь — (`false-stmt`). Можно также воспользоваться оператором `begin-end` для оценки присутствия более одного оператора (использовать как функцию тогда → ещё).

³ Прим.автора. `aref` — функция массива, получает `n`-ый элемент массива (в нашем случае значение «`s`» — шина стереорасширения). `array` — собственно сам массив, `n` — индекс массива (целое число), `returns` — значение элемента массива.

⁴ Прим.автора. `setq` — функция устанавливающая значения символа. Перешла в Nyquist от одного из его прародителей LISP. Однако также близкой к Nyquist структуре SAL свойственна функция `set`. Работает с операторами `sum`, `expr` и `returns`. `sum` — это устанавливаемый символ (обычно заключаемый в кавычки). `expr` — новое значение, `returns` возврат нового значения.

⁵ Прим.автора. `cons` — функция создания нового списка, использует следующие операторы: `expr 1`, `expr 2` и `returns`. `expr 1` — каретка нового узла списка, `expr 2` — заголовок нового узла списка, `returns` — новый узел списка.

⁶ Прим.автора. В приложении № 2 описывается имплементация программного кода (язык Nyquist), показанного в приложении № 1. Имплементация осуществлена Стивом Далтоном (Steve Daulton) и выпущена под лицензией GPL v.2.0. Оригинальное изложение программного кода стандартной реверберации на языке LISP — Роджер Данненберг (Roger Dannenberg). Предупреждения, разработанные для данного модуля, являются интеллектуальным достоянием Уина Бента (Win Bent)*. Важно! В описываемом программном коде (приложение № 2) фильтр низких частот и стерео диапазон воздействуют только на `\n` реверберацию (субканалы воспроизведения), а не на сухой (полезный) сигнал.

* Уин Бент (Win Bent) — американский композитор, исполнитель и музыкальный экспериментатор, в том числе в области компьютерной музыки. Текущий адрес проживания: г. Остин, штат Техас.

граммного кода, проиллюстрированного в виде *приложений*, позволит лучше разобраться в тонкостях настройки классической реверберации. Таким образом, автор статьи произвёл анализ возможностей редактора Audacity® в области компьютерного моделирования эффекта реверберации интерфейсно-ориентированным и программным способами.

Эксперименты, поставленные автором в ходе исследования, подтвердили возможность использования программного комплекса Audacity® в качестве всестороннего развитого средства, обеспечивающего полный контроль над процессами обработки аудиоматериала.

Конечно, Audacity® как профессиональный редактор звука, продолжает совершенствоваться, выходят его новые версии, которые добавляют разносторонние функции, опции которых реализуются через интерфейсную оболочку программы, однако одно остаётся неизменным — это совершенствование мастерства работы над аудиоматериалом. В современных условиях это довольно непростой процесс, требующий комплексных, порой полидисциплинарных знаний разного рода научных дисциплин.

В первую очередь, это, конечно же, современная информатика и вычислительная техника, и соответствующие ей навыки работы с различными языками программирования, включая понимание структур данных, типы языков, алгоритмику и прочее.

Во вторую очередь, это дисциплины естественнонаучного профиля, которые позволяют ориентироваться в свойствах звука, такие как физика (акустика), медицина (аудиология) и другие уважаемые дисциплины.

В третью очередь, это научные дисциплины, позволяющие формировать творческие замыслы в данной области и воплощать их с помощью вычислительной техники — это культурология (культура возделывания аудиоматериала, его синтез и воспроизведение, культура работы с аудиоданными), психология (основы восприятия человеком звуковой информации, знание психоакустических свойств звука), а также некоторые другие, родственные им науки.

Статьи подобного толка призваны привнести инновационные методы в исследуемую нами область и дополнить современные техники компьютерного редактирования аудиоматериала.

Автор выражает надежду на то, что написанный им материал послужит хорошей вспомогательной базой для исследователей, занимающихся проблематикой работы со звуком, а также станет предметом научной дискуссии в этой увлекательной области исследований.

Приложение № 1

```

(defun reverb (x time)
  (multichan-expand "REVERB" #'reverb-mono
    '(((SOUND) "snd") ((NUMBER) "time"))
    x time))

(defun reverb-mono (ga irevfactor)
  (let (sr ilowpass idel ihz icsc acomb1 allp1 allp2 allp3 allow allp4 allp5
        arevout)
    (setf sr (snd-srate ga))

    (setf ilowpass 9000.000)

    (setf idel (list
      (✓ 1237.000 sr) (✓ 1381.000 sr) (✓ 1607.000 sr)
      (✓ 1777.000 sr) (✓ 1949.000 sr) (✓ 2063.000 sr)
      (✓ 307.000 sr) (✓ 97.000 sr) (✓ 71.000 sr)
      (✓ 53.000 sr) (✓ 47.000 sr) (✓ 37.000 sr)
      (✓ 31.000 sr)))

    (setf ihz (mapcar #'✓ idel))

    (setf icsc (list
      (* irevfactor 0.822) (* irevfactor 0.802)
      (* irevfactor 0.773) (* irevfactor 0.753)
      (* irevfactor 0.753) (* irevfactor 0.753)
      (* irevfactor 0.7)))

    (setf acomb1 (sum
      (comb ga (nth 0 icsc) (nth 0 ihz))
      (comb ga (nth 1 icsc) (nth 1 ihz))
      (comb ga (nth 2 icsc) (nth 2 ihz))
      (comb ga (nth 3 icsc) (nth 3 ihz))
      (comb ga (nth 4 icsc) (nth 4 ihz))
      (comb ga (nth 5 icsc) (nth 5 ihz))))

    (setf allp1 (alpass acomb1 (nth 6 icsc) (nth 6 ihz)))
    (setf allp2 (alpass allp1 (nth 6 icsc) (nth 7 ihz)))
    (setf allp3 (alpass allp2 (nth 6 icsc) (nth 8 ihz)))
    (setf allow (lp allp3 ilowpass))
    (setf allp4 (alpass allow (nth 6 icsc) (nth 9 ihz)))
    (setf allp5 (alpass allp4 (nth 6 icsc) (nth 11 ihz)))

    allp5
  ))

```

Приложение № 2

```

;control time "Reverberation time" real "seconds" 2.5 0.1 10
;control damping "High frequency damping" real "%" 50 0 100
;control low-cut "Low cut filter" real "Hz" 200 0 1000
;control width "Stereo Width" real "%" 40 0 100
;control mix "Reverb mix proportion" real "%" 20 0 100

;limit values to valid range
(setq time (max 0.1 time))
(setq damping (max 0 (min 1 (/ damping 100.0))))
(setq low-cut (max 0 (min (/ *sound-srate* 2.0) low-cut)))
(setq width (max 0 (min 1 (/ width 100.0))))
(setq mix (max 0 (min 1 (/ mix 100.0))))

;; create new Left channel
(defun Lsplit (leftch df)
  (let((f1 (lp leftch 300)) ;Common Sub below 300
        (f2 (feedback-delay(lp(hp leftch 300)600)(* df 0.2) 0))
        (f3 (lp(hp leftch 600)1200)) ;Common MID - 600-1200
        (f4 (feedback-delay(lp(hp leftch 1200)2400)(* df 0.08) 0))
        (f5 (hp leftch 2400))) ;Common Presence
    (sim f1 f2 f3 f4 f5))) ;mix sounds

;; create new Right channel
(defun Rsplit (rightch df)
  (let((f1 (lp rightch 300)) ;Common Sub below 300
        (f2 (feedback-delay(lp(hp rightch 300)600)(* df 0.14) 0))
        (f3 (lp(hp rightch 600)1200)) ;Common MID - 600-1200
        (f4 (feedback-delay(lp(hp rightch 1200)2400)(* df 0.06) 0))
        (f5 (hp rightch 2400))) ;Common Presence
    (sim f1 f2 f3 f4 f5))) ;mix sounds

(defun stereo (s-in w)
  (let ((dlay (+(* w 0.5)0.2)))
    (if (arrayp s-in)
        (vector
         (sim
          (mult w (Lsplit (aref s-in 0) dlay))
          (mult w -0.5 (Rsplit (aref s-in 1) dlay))
          (mult (- 1 w) (aref s-in 0))))
         (sim
          (mult w -0.5(Lsplit (aref s-in 0) dlay))
          (mult w (Rsplit (aref s-in 1) dlay))
          (mult (- 1 w)(aref s-in 1))))
        s-in)))

```

ЛИТЕРАТУРА

1. Компьютерная программа Nyquist IDE v.3.15 / Файл директории (C:\Users\Name\Nyquist) // OS: MS Windows, GNU Linux /// Полная реализация — Jesse Clark, David Hovard, David Movatt, David Deangelis, Roger B. Dannenberg. — 2002–2018. [Электронный источник, компьютерная программа].
2. Компьютерная программа Audacity® v.2.1.3 / Файл директории (C:\Program Files (x86)\Audacity) // OS: MS Windows, GNU Linux /// Полная реализация — Gale Andrews, Arturo «Buanzo», James Crook, Roger B. Dannenberg, Steve Daulton, Vaughan Johnson, Greg Kozikowski, Paul Licameli, Peter Sampson, Martyn Shaw, Bill Wharrie. — 1999–2017. [Электронный источник, компьютерная программа].
3. Таран В.В. Язык программирования Nyquist: настоящее время и перспективы его развития в области компьютерной аудиоинженерии и аудиоинформатики / В.В. Таран // Современная наука актуальные проблемы теории и практики: Серия естественные и технические науки // Информатика, вычислительная техника и управление. — 2020. — № 4. — С. 135–153. [ISSN2223–2966]. (DOI 10.37882/2223–2966.2020.04.37).
4. Таран В.В. Корректировка аудиосигнала при монтаже аудиозаписей в программной среде Audacity®, используя мультифункциональные возможности языка программирования Nyquist / В.В. Таран // Современная наука актуальные проблемы теории и практики: Серия естественные и технические науки // Информатика, вычислительная техника и управление. — 2021. — № 3. — С. 155–202. [ISSN2223–2966]. (DOI 10.37882/2223–2966.2021.03.32).
5. Таран В.В. Цифровая библиотека чтения и записи аудиоданных Libsndfile: техническая структура, возможности использования и перспективы развития / В.В. Таран, Р.С. Гиляревский // Современная наука актуальные проблемы теории и практики: Серия естественные и технические науки // Информатика, вычислительная техника и управление. — 2021. — № 6. — С. 131–155. [ISSN2223–2966].
6. Psychoacoustics: perception of normal and impaired hearing with audiology applications / Jennifer J. Lentz. — Plural Publishing, Inc. 2020. (Editor-in-Chief for Audiology Brad A. Stach, PhD). — p. 238 [ISBN9781597569897; ISBN1597569895].
7. Sound Reproduction (The Acoustics and Psychoacoustics of Loudspeakers and Rooms) / Third Edition: Floyd E. Toole (Audio Engineering Society (AES). Third edition published 2018, by Routledge 711 Third Avenue, New York, NY10017–2018 (Taylor & Francis). — p.490 [ISBN9781138921368; ISBN: 978–1–138–92137–5; ISBN: 978–1–138–92136–8; ISBN: 978–1–315–68642–4].
8. Optimal utilization of a reverberation chamber / Leferink, F. and van Etten. — Euro EMC2000, Symposium on EMC, Brugge, pp. 201–206
9. Anechoic and Reverberation Chambers (Theory, Design, and Measurements) / Qian Xu and Yi Huang. — 2019 (John Wiley & Sons Ltd). p.380 [ISBN: 9781119361688].
10. Simoni M., Dannenberg R., Algorithmic Composition (A Guide to Composing Music with Nyquist) / Published in the United States of America by The University of Michigan Press Manufactured in the United States of America (e-book). — 2008, 2013. — p.249. [ISBN978–0–472–02905–1].
11. Audacity® 2.1.3 Manual, standalone (offline version: C:\Program Files (x86)\Audacity\help\manual\man\index_of_effects_generators_and_analyzers.html [дата обращения к электронному ресурсу: 14.09.2022].
12. Audacity® 2.4.2 Manual, WEB-version, URL: <https://manual.audacityteam.org/> [дата обращения к электронному ресурсу: 14.09.2022].
13. Touretzky, David S. Common LISP: a gentle introduction to symbolic computation /Carnegie Mellon University /// Copyright © 1990 by Symbolic Technology, Ltd.///Published by The Benjamin/Cummings Publishing Company, Inc.p.587 [ISBN0–8053–0492–4].
14. Seibel Peter. Practical COMMON LISP /APRESS — 2005, p.528 [ISBN1–59059–239–5].
15. Dannenberg R.B. Nyquist Reference Manual Version 3.19 // Carnegie Mellon University — School of Computer Science/ Pittsburgh, PA 15213, U.S.A. 2013–2022 WEB-version, URL: <http://www.cs.cmu.edu/~rbd/doc/nyquist/> [текущая версия учебника, дата обращения к электронному ресурсу: 14.09.2022].
16. Dannenberg R.B. Nyquist Reference Manual Version 3.16 // Carnegie Mellon University — School of Computer Science/ Pittsburgh, PA 15213, U.S.A. 2013–2020 WEB-version, URL: [http://www.cs.cmu.edu/~rbd/doc ar-3.16/nyquist/](http://www.cs.cmu.edu/~rbd/doc/ar-3.16/nyquist/) [архив, дата обращения к электронному ресурсу: 14.09.2022].
17. Dannenberg R.B. Nyquist Reference Manual Version 3.15 // Carnegie Mellon University — School of Computer Science/ Pittsburgh, PA 15213, U.S.A. 11.08. 2018, p.276.
18. Dannenberg R.B. Nyquist Reference Manual Version 3.09 // Carnegie Mellon University — School of Computer Science/ Pittsburgh, PA 15213, U.S.A. 27.12. 2014, p.297.
19. Dannenberg R.B. Nyquist Reference Manual Version 2.36 // Carnegie Mellon University — School of Computer Science/ Pittsburgh, PA 15213, U.S.A. 05.03. 2007, p.205.

© Таран Василий Васильевич (allscience@lenta.ru).

Журнал «Современная наука: актуальные проблемы теории и практики»