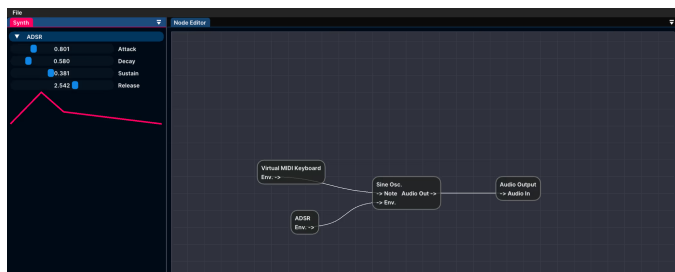# Synthesis of sound with nodes

Fig. 1. A similar project Node-Based Audio Synth [3], implemented in C++ programming language. The image displays a simple UI with a couple nodes that generate sound.

Abstract—**This project aims to create a tool for computer sound synthesis by only using nodes, which can be placed on a canvas and connected with wires. The main objective of the project is to create a tool for learning computer sound synthesis without the need for programming.**

I. Introduction

Computer sound generation became the norm in the current era of entertainment industries. Game, movie, music, and other industries have all adopted the computer sound generation, and practically all of the products these industries produce use some form of computer sound generation. This project's main objective is to implement a tool for computer sound synthesis with visual programming. The application's main goal is to help users learn the basics of computer sound generation without the need for programming. Figure 1 shows an example tool for a node-based audio synth.

II. Goals

The main goal of this project is the implementation of a tool for computer sound synthesis with visual nodes instead of the usual programming approach. The tool will provide a collection of nodes for sound generation, i.e., sine wave generator, file input; nodes for sound manipulation, i.e., adder, multiplier, passthrough, filters; and for sound output, i.e., players, file writers, and others. The user will be able to place the nodes on a canvas and connect them with a wire, resulting in a transfer of data from the output of a node to the input of another node. The tool's main objective is to improve the learning process of computer sound generation without the need for programming, which requires an intuitive and easy-to-use interface.

III. Implemetaion

The main programming language used in this project will be C++. For the input and output of sound, RtAudio [2], a lightweight, cross-platform API (Application Programming Interface) for realtime audio input and output, will be used. UI (User Interface) will be implemented using an open source library ImGui [4] for creation of GUIs (Graphical User Interfaces) in C++ in combination with ImGui Node Editor [5], a node editor implementation.

The end product will give the user a collection of nodes used to generate, manipulate, and output a sound. The node collection will be heavily inspired by the one provided with the Java API JSyn [1]. Example nodes are: sine wave generator, adder, multiplier, passthrough, different filters, and a standard output node. Those nodes will have inputs and outputs, defined with so-called **pins**, which are connected with **wires**. The user will be given the power to instantiate the node on a canvas and connect an output to an other node's input with a wire. With this process, any sound synth can be produced.

First, we will implement individual components (sine wave generator, adder, etc.) in C++ and define an API for creation, deletion, and connecting of those components. Then, we will use the API to implement a user interface with ImGui Node Editor [5].

Another feature of the tool will allow the users to implement their own nodes with extensions of the base classes provided with the API.

IV. Evaluation

The solution will be evaluated with the help of JSyn [1]. The same synth will be assembled in both tools and the output compared by looking at the raw data and by listening to the sound. The implemented tool will be evaluated based on the deviation in the data.

The performance will be tested in a similar way. Both tools will be used to create the same synth, and the time of synthesization will be measured.

References

[1] JSyn: API for developing interactive sound applications in Java https://www.softsynth.com/jsyn/
[2] RtAudio: API for realtime audio input/output in C++ https://www.music.mcgill.ca/ gary/rtaudio/
[3] Node-Based Audio Synth https://eilu.me/en/portfolio/node-based-audio-synth/
[4] ImGui: API for GUI in C++ https://github.com/ocornut/imgui
[5] Imgui Node Editor: Implementation of a node editor using ImGui https://github.com/thedmd/imgui-node-editor