

Realtime Guitaramplifier model

Roope Kiiski	217974
Niko Lindvall	350051
Alvar Wegelius	296623

Reaaliaikainen kitaravahvistinmalli

Roope Kiiski
Niko Lindvall
Alvar Wegelius

Tiivistelmä

Työn tavoitteena oli ohjelmoida C++:lla Juce-ympäristöä käyttäen reaaliaikaisesti toimiva kitaravahvistinmalli VSD-pluginina. Työssä on mallinnettu kitaravahvistimen eri putkiasteet sekä ekvalisaattori. Lisäksi laskostumisen minimoimiseksi käytettiin mallissa up- ja down-samplaysta. Lisäksi kokeilimme toteuttaa kaiutinmallin, mutta sen tosimme toimimattomaksi.

1 Johdanto

Kitaravahvistin on laite, jota käytetään sähkökitaraa soittaessa luomaan soittimelle ja soittajalle ominainen ääni. Historiallisesti putkivahvistimia on käytetty erittäin paljon, koska monet soittajat kokevat niiden äänen olevan miellyttävämpi ja monipuolisempi kuin transistorivahvistimien. Putkivahvistimet ovat kuitenkin painavia, herkkiä kolahduksille ja usein myös hyvin kalliita laitteita. Kuitenkin koko ajan musiikkimaailma on siirtymässä kohti digitalisia versioita soittimistaan sekä vahvistimistaan, joten on tärkeää että myös kitaravahvistinmalleja luodaan, ja että ne kuulostavat hyvältä. Digitaaliset kitaravahvistinmallit ovat merkittävästi halvempia, kestävämpiä sekä monipuolisempia kuin perinteiset vahvistimet, joten jos niistä saadaan riittävän hyviä, todennäköisesti ne yleistyvät nopeasti. Tämän työn tarkoituksena on ohjelmoida kitaravahvistinmalli, joka kuulostaa oikealta, sekä jossa on kaikki kitaravahvistimen toiminnan kannalta tärkeät osat.

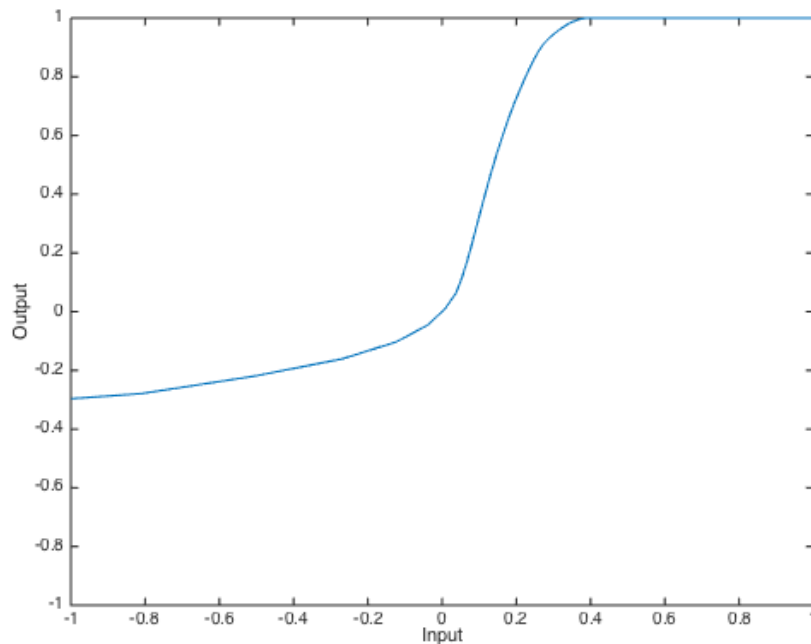
2 Metodit

2.1 Teoria ja algoritmit

Kitaravahvistimen pääasiallinen tarkoitus on vahvistaa sähkökitarasta tuleva sähköinen signaali, mutta myös muokata sitä erilaisin tavoin, kuten esimerkiksi lisäämällä säröä. Yksi merkittävä ero kitaravahvistimen ja geenerisen vahvistimen välillä on se, että kitaravahvistimen taajuusvaste ei edes pyri olemaan lineaarinen. Säröä kitaravahvistin lisää joko leikkaamalla signaalia, tai vahvistamalla signaalia epälineaarisesti. [2]

Putkivahvistimet erityisesti tunnetaan omanlaisesta säröstään, joka johtuu vahvistinputkien siirtofunktion epälineaarisuudesta sekä siirtofunktion epäsymmetrisyydestä. Tämä on havainnollistettu kuvassa 1, jossa on esitetty ensimmäisen vahvistinasteemme putken siirtofunktio. [2]

Vahvistinputkessa särö syntyy, kun erivahvuuksiset sisääntulevat signaalit vahvistuvat eri verran, jolloin signaalin muoto vääristyy. Erityisesti siirtofunk-



Kuva 1: Vahvistinputken siirtofunktio, josta nähdään ulostulevan ja sisäänmenevän signaalin suhde.

tion epäsymmetrisyys, eli se, että positiiviset sisääntulon arvot vahvistuvat eri tavoin kuin vastaavat negatiiviset arvot, aiheuttaa sen, että vahvistinputken särö kuulostaa erilaiselta kuin transistorivahvistimen. Transistoreiden siirtofunktio on usein symmetrinen, ja sen seurauksena niiden muodostama särö on vain parittomia harmonisia, kun taas putken muodostama epäsymmetrinen särö sisältää myös parillisia harmonisia komponentteja. [2]

Tässä työssä vahvistinputkea simuloidaan käyttämällä valmiiksi mitattujen vahvistinputkien siirtofunktiota.

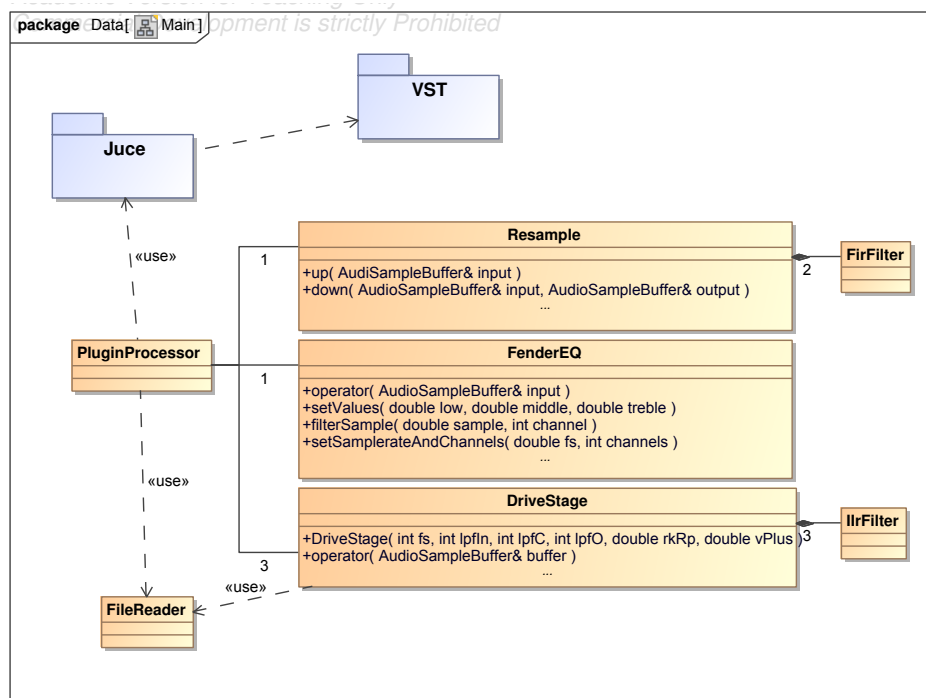
2.2 Työkalut

Kitaravahvistinmalli toteutettiin C++-kielellä käyttäen Juce-ympäristöä. Käytännössä joitain algoritmeja testattiin Matlabissa, ja kun ne todettiin toimiviksi, siirrettiin ne C++-koodiksi. Yksinkertaisemmat algoritmit toteutettiin suoraan C++:lla

3 Toteutus

3.1 Rakenne

Toteutuksen rakenteessa pyrittiin noudattamaan olio-ohjelmoinnin periaatteita. Ohjelman luokkakaavio on esitetty kuvassa 2. Luokkakaaviossa on kuvattu vain oleelliset funktiot. PluginProcessor on VST-pluginin pääluok-



Kuva 2: Luokkakaavio.

ka, joka hoitaa eri osien yhteenliittämisen. Resample-luokka sisältää ylös- ja alasnäytteistyskseen. FenderEQ:ssa on vastaavasti toteutettu ekvalisaattori. Suodattimet on toteutettu myös erillisiksi luokikseen. Luokkiin toteutettiin funktio-operaattori, jolloin niiden käyttö on helppoa muusta ohjelmakoodista.

3.2 Ylös- ja alasnäytteistys

Ylösnäytteistyksessä signaalin näytteiden väliin lisätään nollia. Tämän jälkeen saatu ulostulo suodatetaan interpoloivan alipäästösuodattimen läpi.

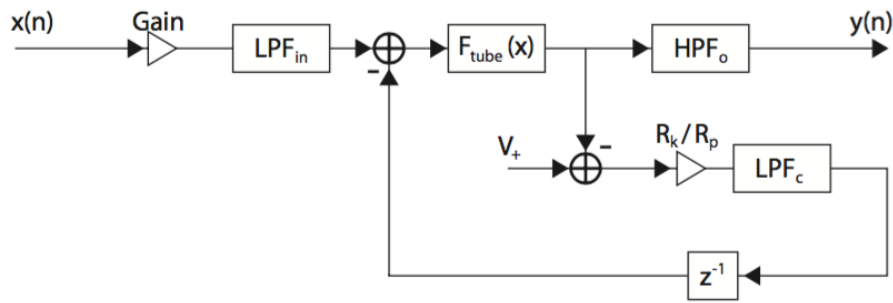
Alasnäytteistyksessä signaali ensin alipäästösuodatetaan ja tämän jälkeen siitä poimitaan joka toinen arvo.

Up- ja downsamplayksessä käytettävät FIR-suodattimet osoittuivat ongelmallisiksi soittaessa aiheuttaen pauketta ja muita artefakteja. Korjausyrityksistä huolimatta suodattimet eivät toimineet halutulla tavalla joten päätimme ottaa ne pois käytöstä.

3.3 Vahvistinasteet

Kitaramallimme toimii kolmella vahvistinasteella, joista toinen ja kolmas ovat putkimalliltaan identtiset, ja ensimmäinen erilainen, mutta muilta parametreiltaan kaikki kolme mallia ovat erilaisia. Tarkoituksena on simuloida vahvistimissa esiintyviä eri vahvistinasteita ja näin saada aikaan realistisempi ääni. Kuvassa 3 on esitetty yksittäisen vahvistinasteen lohkokaavio.

Lohkokaaviossa LPF-lohkot ovat alipäästösuodattimia, F_{tube} -lohko on putki-



Kuva 3: Vahvistinasteen lohkokkaavio.

malli, sekä HPF-lohko on ylipäästösuodatin.

3.4 Ekvalisaattori

Ekvalisaattori päätettiin tähän työhön toteuttaa hydyntäen Fenderin '59 Bassmanin digitalisointia. Tarkoituksena on tällöin saada mahdollisimman aidon kuuloinen vahvistin, sekä todellista vastaava ekvalisaattorin toiminta. Kitaravahvistimien ekvalisaattorit eivät kuitenkaan koskaan ole ideaalisia, joten käyttämällä alipäästö-, kaistanpäästö, sekä ylipäästösuodattimia ekvalisaattorina, olisi lopputulos ollut keuhno. [1]

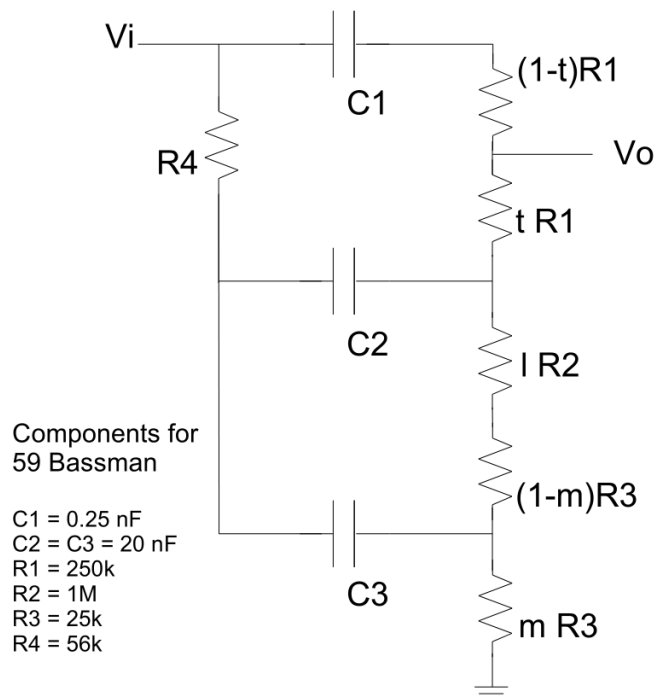
Käytännössä Fenderin ekvalisaattori on toteutettu kuvassa 4 esitetyn mukaisesti, ja se voidaan erilaisin matemaattisin operaatioin muuttaa muotoon joka on esitetty kaavassa 1. Kaavan kertoimet ovat tarkemmin avattu alkuperäisessä julkaisussa. [1]

$$H(z) = \frac{B_0 + B_1 z^{-1} + B_2 z^{-2} + B_3 z^{-3}}{A_0 + A_1 z^{-1} + A_2 z^{-2} + A_3 z^{-3}} \quad (1)$$

Käytännössä siis ekvalisaattori toteutetaan kolmannen asteen IIR-suodattimena, jonka kertoimet muuttuvat aina kun käyttäjä muuttaa ekvalisaattorin asetuksia. Tarkempi toiminta, kuten taajuusvasteet, on esitetty alkuperäisessä julkaisussa. [1]

3.5 Kaiutinmalli

Kaiutinmalli on myös erittäin keskeinen osa kitaravahvistimen ääntä, mutta meidän työssämme jouduimme jättämään sen pois, koska emme saaneet sitä toimimaan järkevästi. Käytännössä kaiutinmalli on erittäin pitkä suodatin, ja suodattimen pituus aiheutti lukuisia ongelmia kuten pauketta, jota emme saaneet korjattua järkevällä työmäärällä.



Kuva 4: Fenderin '59 Bassman ekvalisaattorin kytkentä. [1]

4 Tulokset

Lopputulokseltaan kitaravahvistinmallimme toimii varsin hyvin. Särö kuulostaa varsin hyvältä, ekvalisaattori toimii kuten sen kuuluukin ja malli on reaaliaikainen. Kuitenkin ääni jättää paljon toivomisen varaan, sillä siinä on havaittavissa paljon erilaisia artefakteja, joita siinä ei pitäisi olla. Etenkin kun äänikortin bufferi asetetaan pieneksi, kasvaa ylimääräiset särökomponentit sekä naksunnat merkittävästi. Lisäksi myös jotkin filterimme aiheuttavat ääneen havaittavaa ylimääräistä säröä. Nämä ongelmat kuitenkin vaikuttavat olevan sellaisia, että niiden korjaaminen vaatii hyvin paljon perehtymistä ja aikaa, eikä se siten ole tämän kurssin puitteissa järkevää.

Koodiin jäi ajanpuutteen vuoksi vielä joitain muotovirheitä. Lisäksi virhetarkistusta ei ole tehty käytännössä missään. Jos esimerkiksi tiedoston lukeminen menee pieleen, niin ohjelma kaatuu. Tämä ei valmiilta ohjelmalta olisi hyväksyttävää.

5 Johtopäätökset

Kokonaisuutena työ onnistui hyvin. Aikataulujen kanssa oli välillä haasteita, mutta kokonaisuudessaan saimme mallimme toimimaan kuten alunperin oli tarkoituskin, vaikka ääneen jäi lopulta parantamisen varaa. Työstä mielenkiintoisen teki myös se, että eri ryhmän jäsenille eri asiat olivat uutta: osalle esimerkiksi C++-ohjelmointi oli uutta, kuten myös osalle Gitin käyttö

versionhallinnointiin. Toisaalta myös itse reaaliaikaisuus toi ohjelmointiin uusia asioita. Signaalinkäsittelyllisesti työssä menttiin huomattavasti aiempia ohjelmointiprojekteja alemmalle tasolle. Virheiden etsiminen oli välillä todella haasteellista.

Käytännössä tulevaisuudessa, jos työtä jatkaisi, pitäisi eri suodattimia parantaa, jotta eri artefaktit saadaan minimoitua. Myös graafista käyttöliittymää voisi hioa, siitä voisi ajan kanssa tehdä vaikka kuinka hienon, mutta tämän kurssin puitteissa siihen ei ollut järkevää käyttää liikaa aikaa. Isoimpana ongelmana siinä on säätöarvojen skaalaus, jota ei keritty hioda hyväksi.

Viitteet

- [1] David T. Yeh and Julius O. Smith. Discretization of the '59 fender bassman tone stack. *DAFx-06*, 2006.
- [2] Udo Zölzer. Dafx: Digital audio effects, 2nd edition. 2011.