Nonlinear Real-Time Emulation of a Guitar Chain with Machine learning techniques

Thomas Schmitz, Jean-Jacques Embrechts

T.Schmitz@ulg.ac.be, jjembrechts@ulg.ac.be

INTELSIG Laboratory, Montefiore Institute, University of Liège, Belgium

Abstract

Numerous **audio systems** for musicians are expensive and bulky. Therefore, it could be advantageous to model them and to replace them by **computer emulation**. Their **nonlinear behavior** requires the use of complex models. Recent progresses made in the nonlinear system identification field have improved the ability to emulate nonlinear audio systems such as tube guitar amplifiers.

1. Goal

Typical guitar setups (see Fig. 1) should be replaced by computer emulations (see Fig. 2) to ease their displacements and to reduce their costs. The main challenge is the accuracy of such nonlinear emulations.

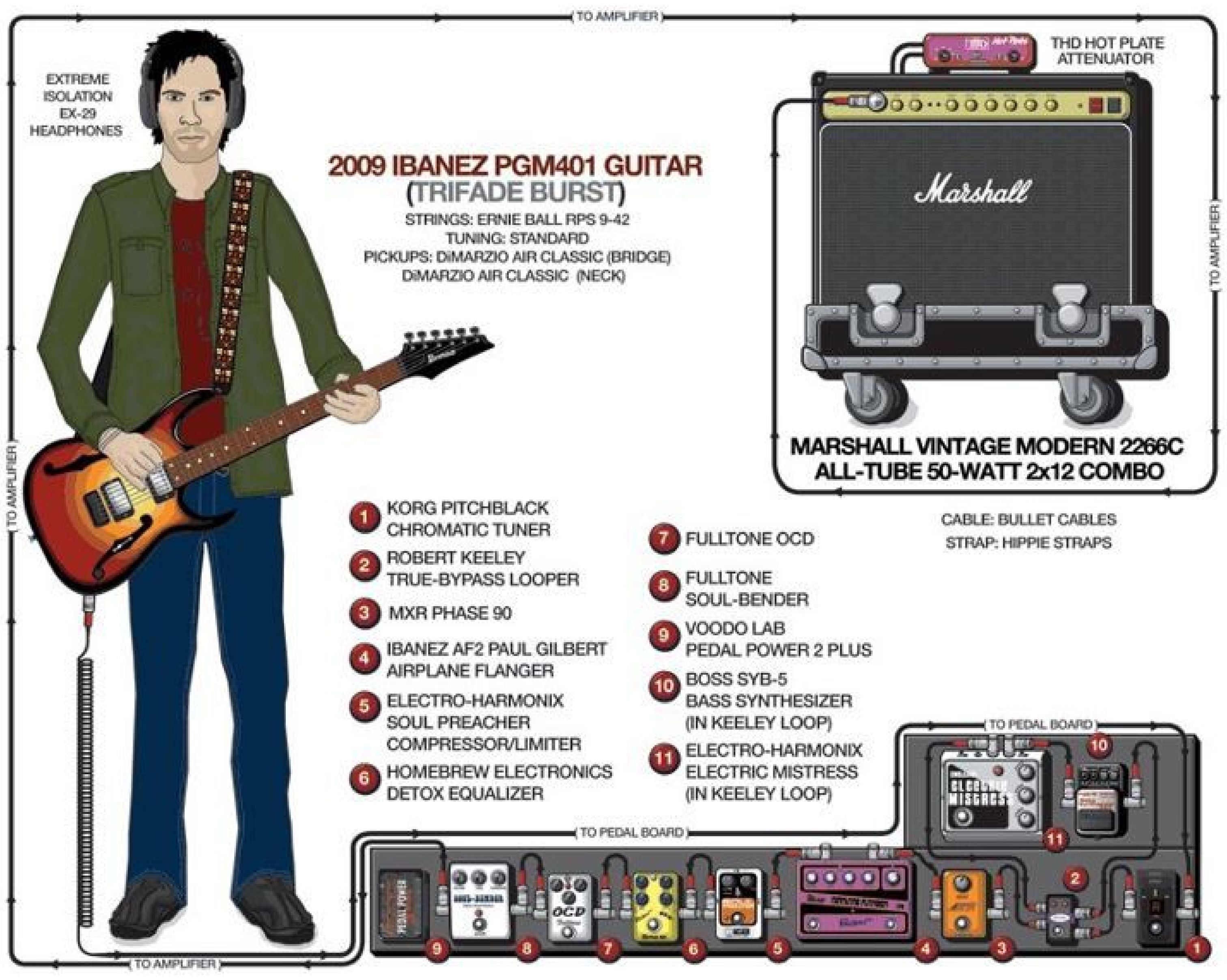


Figure: A typical guitar setup

2. Approach overview

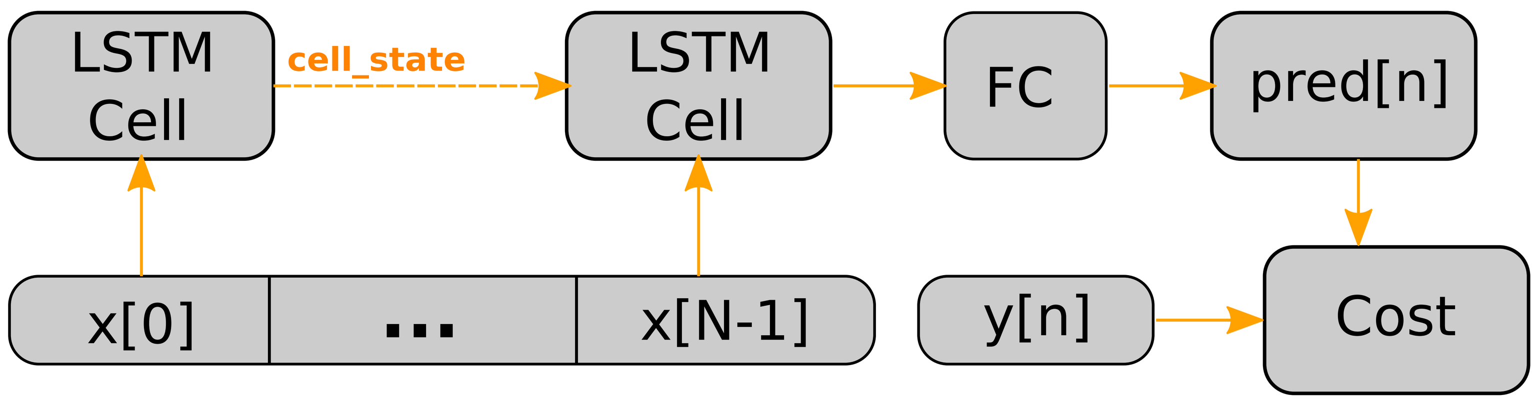
Our method is based on machine learning techniques, a signal $x[n]$ coming from a guitar is sent through the tube amplifier. The resulting signal $y[n]$ is recorded and used to adapt the *bias* and *weights* of a Neural-Network such as the signal $x[n]$ passing through the Neural-Network model (the `prediction` signal $pred[n]$) matches the `target` signal $y[n]$. The dataset containing the input/output signals used for this research is available at `http://www.montefiore.ulg.ac.be/services/acous/STSI/downloads.php`. The measured accuracy is given by the following Performance Index:

$$PI = \sqrt{\frac{\sum_{k=0}^{K}[y_{amp}[k] - pred[k]]^2}{\sum_{k=0}^{K} y_{amp}[k]^2}} \qquad (1)$$

Figure: Replacement of the guitar setup by computer emulation

3. LSTM method

Long Short Term Memory (LSTM) cells are used in our network. An input buffer of size N is used to compute the output state of the LSTM Cell. The predicted output sample $pred[n]$ is obtained from this state by adding a Fully Connected (FC) layer of Neurons. This predicted sample is then compared to the `target` sample $y[n]$ to compute a *cost* function. The Back Propagation Through Time (BPTT) algorithm computes the weights and bias of the network to minimize this cost.



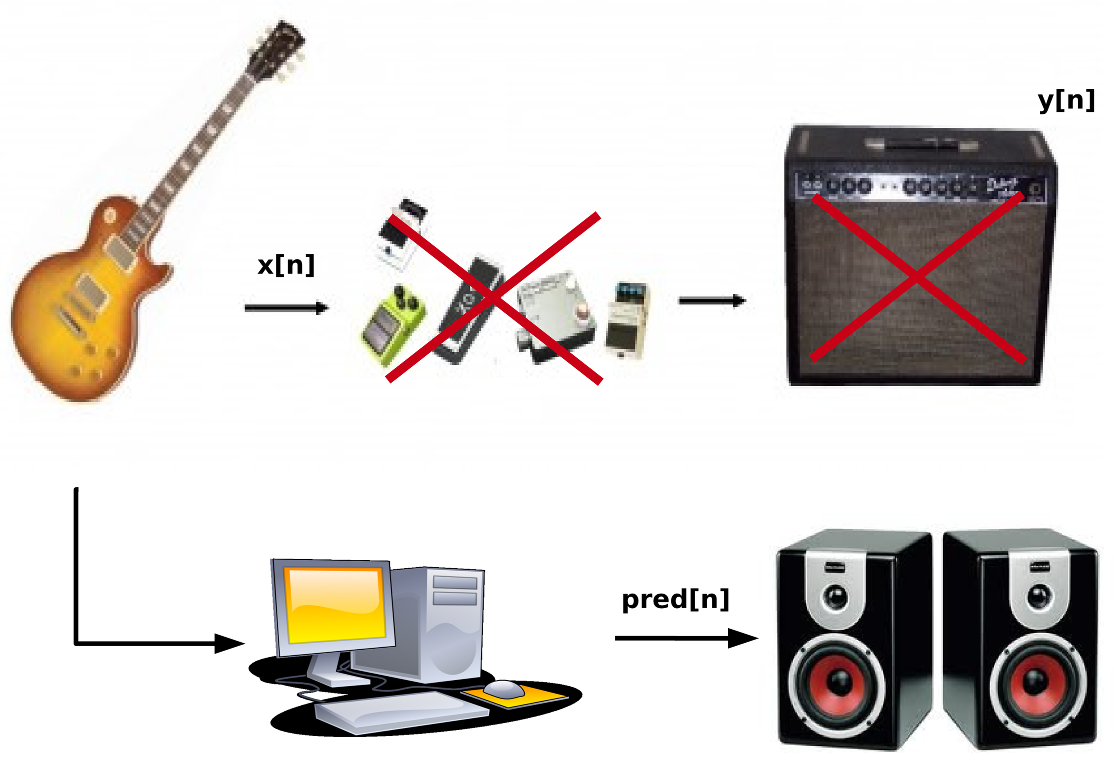
The size of N is chosen to take into account the frequency and amplitude nonlinear dependencies. More informations on this model can be found in `http://hdl.handle.net/2268/227161`.

4. Listening test

For this listening test, a guitar signal (named `inputValidationSet`) is sent through several amplifiers. The signals named *X.Target* are the signals at the output of the amplifiers *X* while the signals named *X.Prediction* are their corresponding emulations from the Neural-Network.

4. Evaluation and results

The picture to the left presents a guitar signal $x[n]$ passing through the *amplifier* giving the *target* signal $y[n]$, or through our emulator giving the prediction signal $pred[n]$. The central picture presents the comparison between the prediction and the target signals. Finally, the picture to the right presents the accuracy of the model on different amplifiers.



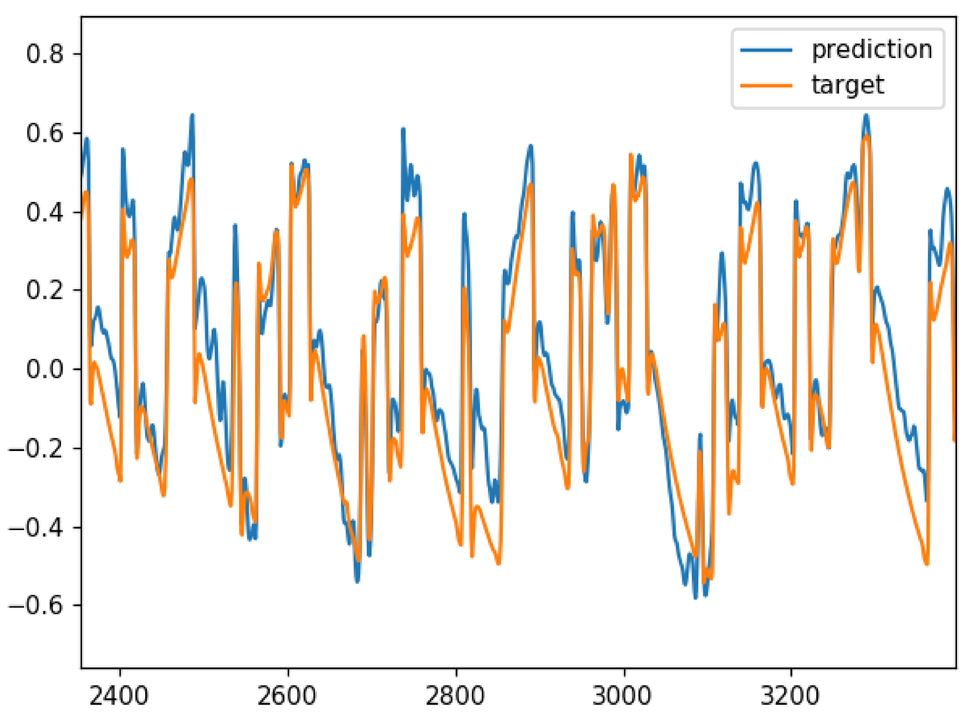


Table: Example of Performance Index:

Amplifier	Gain	PI
Engl retro tube disto	5	32%
MesaBoogie 550 clean	5	36%
MesaBoogie 550 crunch	5	29%
MesaBoogie 550 disto	5	32%
Ibanez TSA15 crunch	5	25%
MesaBoogie MarkV clean	5	4%
MesaBoogie MarkV crunch	5	13%
MesaBoogie MarkV disto	5	22%
Blackstar HT5M	5	28%

the 28^{th} international conference on Noise and Vibration Engineering, Belgium, Leuven, 17-19 September, 2018