# Managing Live Music Bands via Laptops using Max/MSP

Yehiel H. Amo *
syndrome, Israel
amo@syndrome.nu

Gil Zissu †
Uni. of the Arts, london
college of comunication, UK
gilzissu@gmail.com

Shaltiel Elul ‡
Oxford University, UK
shaltiele@gmail.com

Eran Shlomi
syndrome, Israel
ericw@syndrome.nu

Dima Schukin
syndrome, Israel
dima@syndrome.nu

Almog Kalifa
syndrome, Israel
almog0@gmail.com

## ABSTRACT

We use the Max/MSP framework to create a reliable but flexible approach for managing live performances of music bands. This approach allows an easy and low cost way to apply innovative music interfaces for live performance, without losing the professionalism required on stage. In the approach, every 1-3 players are plugged to a unit consisting of a standard sound-card and laptop. The units are in charge of auto-changing presets to manage virtual instruments, effects, and gestures for each player. All the units are then remotely controlled by a conductor patch which in charge of the synchronization of all the players and background samples in real time, as well as providing sensitive metronome and scheduling visual enhancement. Last but not least, we can take the advantage of using virtual instruments and effects in Max/MSP to manage the mix and routing the audio. This provides metronome and monitor system to the players' ears, and also virtual live mixing. This privilege almost eliminates the dependence on the venue's equipment, and in that way, the sound quality and music ideas can be brought directly from the studio to the stage.

## Keywords

Liveness, interactive interfaces, synchronization, intiligent performer, Max/MSP, live bands, low cost.

## 1. INTRODUCTION

In live performance, the ability to deliver the same music or sound concept as in the studio is a challenging aspect. Previous works [6] debate the issue of loss of performativity in live performing with laptops due to lack of gestures and visual expressions from the performer [11].

Music bands have another significant challenge on bringing ideas from the studio to the stage. In the studio, almost unlimited variations of sound modulations and effects can be assigned together to each instrument. Yet in live performance this process cannot be easily achieved. It is usually limited by the band's equipment and relies on the equipment available at the concert venue (amplifiers, monitors, and mixer). Therefore, the sound of each player and the output mix is significantly shifted from the composer's original intention. In certain cases, this shift is harmless or even contributes to the liveness of the performance. However, for some bands, the sound and recorded samples are an integrated part of the composition (for example music that involves "concrete music" or background samples [6]). In these cases, this shift can significantly reduce the experience in a live performance [11].

These two challenges can be reduced greatly or even eliminated by using a high cost music production. However, this is not the case for the standard music band who playing at a standard venue. Hence, many bands in a range of music genres have this limitation of bringing their own sound from the studio to the audience in an interactive and convincing way.

Max/MSP specializes in creating an interactive interface between sound, visual and physical playing in real time[8, 1, 10]. Nowadays, there are also many commercial or open source virtual plug-ins of audio digital effects that can replace and simulate analog amplifiers and effects for electric guitars, bass guitars, and vocals [9, 12, 3]. Governing these new technological advantages, we can take the advantage of Max/MSP [8] to manage and create the virtual plug-ins as well as the mix, audio routing in order to reduce and almost eliminate the dependency on the venue's equipment. As a result, the sound and music ideas can be taken out from the studio to the stage.

## 2. RELATED WORKS AND TARGETS

Many works have been done in the area of combining laptops in live performances to synchronize music and coordinate media [4, 5]. For example, works on automatic synchronization[10] and accompaniment systems for classical music performance [2], and recently, an important work by Dawen at al. suggested a framework of media synchronization with intelligent tempo detecting on 'real time' [7]. Here in this work, we focus on controlling the whole environment for live performances in a practicable way. The aims and considerations of the approach are as follows:

The first aim is to enhance the interaction between the live playing and the electronic music by; synchronizing samples and visual enhancement (by static tempo), ability to use metronome, and automation for changing sounds and effects in live. In this stage, we still wish to consider 'known' tempo for media synchronization instead of using intelligent tempo detector for practical reasons as will be discussed later. The second aim is to control the audio signal and audio-routing of the band. This approach uses laptops and

*Inter alia, sound integration and performance tests

†Inter alia, graphic art

‡Corresponding Author


*NIME'14,* June 30 – July 03, 2014, Goldsmiths, University of London, UK.

sound-cards to manage the audio routing to eliminate the dependence on the venue's equipment ("plug and play"). There are some considerations that took into account while developing the approach. The reliability along the performance is in high priority, keeping the CPU and latency low enough to get smooth performance at every concert. Still, it is also important to use standard quality of sound cards and computers to achieve high performance result in low cost equipment.

# 3. METHODOLOGY

The approach is programmed in Max/MSP. All the audio signals from the music instruments are controlled by standard 13" laptops with 2.4Ghz, Intel core 2 duo processors and external standard portable sound-card contains 4 audio inputs , 4 audio outputs, and a standard S/PDIF or optic audio connection. Each laptop is connected to one audio card and can be assigned to typically two players without overloading the CPU and keeping the latency low enough (performance tests are detailed later in section 5).

The approach contains three main patches. A scheme of the connections between the patches and laptops is provided in Figure 1. Massages between the laptops are remotely transmitted via UDP massages (around 1ms latency for UDP massages in inner network). The first patch, named *'Conductor'*, is used for triggering samples and synchronizing all the players. The second patch named *'Child'* is an interface that provides all the virtual environment for every player on stage. The third patch, named *'Virtual mixer'*, allows virtual mixing of all the live music instruments. All the patches are described further in this section.

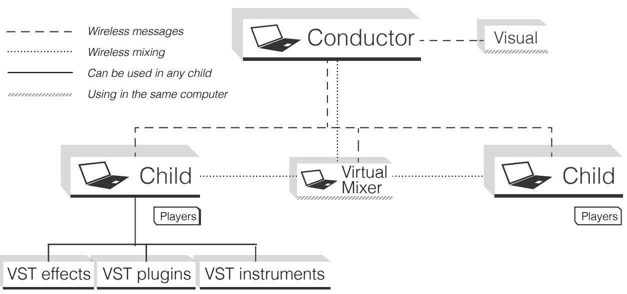


**Figure 1: Scheme of the communication between the leptops and patches.**

## 3.1 *'Conductor'*

This patch is made to conduct the songs and the whole performance. It is designed to trigger background samples and loops via MIDI massages in order to accompany the song in live playing, and to send UDP massages to trigger the other patches (*Child*, a visual patch, and *Virtual mixer*). Usually it is favourable to control the *conductor* patch by the keyboard player, so the player can control some music parameters, such as dynamics and filters with sliders and knobs, in order to increase the interactivity of background samples in real time. The *conductor* also consists a time synchronization part, to track the time (bar and beat) of each *child* patch. This is done by feedback massage of the time (bar and beats) from the other laptops.

### *3.1.1 Metronome*

Additional purpose of the *Conductor* patch is to send sensitive metronome massages. A sub-patch of metronome is embedded in each *Child* patch to provide metronome for every player via "in-ears" headphones. The headphones also provide the stage monitor system for the performers, as will be described later in the audio routing section. The *Conductor* adjusts the metronome's volume relative to the sound signal and sends these massages to the *Child* patches. This automatic volume balance makes the metronome convenient for all the player, which is crucial in order to avoid leakage of the metronome click to the audience in very quiet music phrases, or on the other hand, to be high enough (especially for the drummer) in 'full' volume playing phrases.

### *3.1.2 Visual Accompany*

In this approach it is also possible to trigger and control visual art, movies, or pictures during the performance. We built a Max/MSP/Jitter patch that provides the visualization and also controlled by the *Conductor*. It receives from the *Conductor* presets massages as well as audio signal values to make interactive visual, stimulated by the music.

## 3.2 *'Child'*

The *Child* patch is designed to manage the audio signal of the instrument for each player. In that manner the musical instruments (electric guitars, bass guitar, and microphones) should be plugged to sound-cards (typically 2 players per one sound-card and *Child* patch). The audio signal from the instrument is generated by VSTs (see figure 1) controlled via Max/MSP in the *Child* patch. The VST can be either a commercial plug-in of virtual amplifiers and effects, in order to simulate the sound of an analogue amplifier for the electric guitar and bass or a 'homemade' Max/MSP effects and virtual instruments. The role of the *Child* is therefore,to give an automatic control of the sound in live. The player chooses his sound in the desired VST, adjusts his effect and volume. Then set a preset to a specific time in the music track (bar and beat). In real time, when a trigger massage is sent from the *Conductor* to the *Child*, a specific song's plan is generated and the presets are automatically changed. Figure 2 shows an example of presets that was planned for a song with a metronome of 97bpm. This automation allows the player, such as the guitarist or

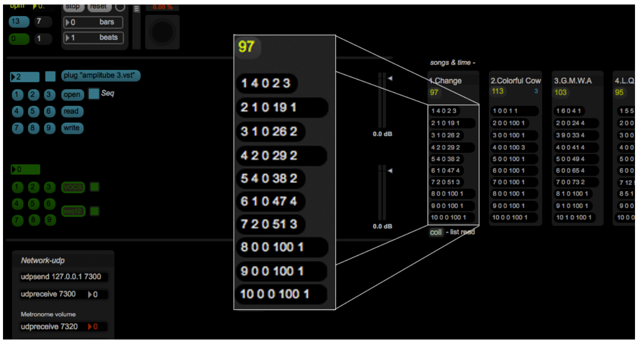


**Figure 2: The *Child* patch that made for automations of presets (close-up view on one song plan).**

the singer to get many variations along a song in real time, and reach a similar sound in every performance.

### *3.2.1 Using 'Home-made' Virtual Instruments*

In order to show the ability to combine interactive interfaces in the approach, we demonstrate three patches of virtual instruments that we built in Max/MSP. These patches are made for modulating sounds and adding effects, sampling and granulating sounds, and creating loops. A short demonstration of the virtual instruments is provided[1]. With this

[1] www.youtube.com/watch?v=FfRsaDV-z4A

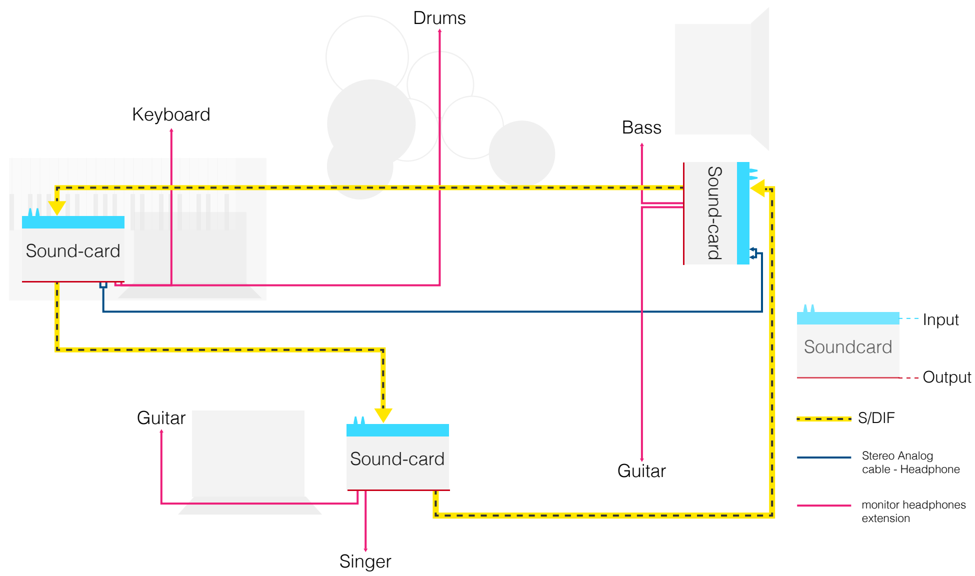


**Figure 3: Bird's eye view scheme of the audio routing between the players on stage.**

approach it easy to combine such instrument patches in the main *Child* patch to get automatic control by presets or by the instrument's player via MIDI massages.

Creating MIDI and physical gestures in these virtual interfaces is especially important to enhance the interaction. For instance, in the attached URL[2], we demonstrate one example of the use of the virtual instruments in a guitar solo. The pedal of the guitarist is assigned to modify the sound of the guitar (via pitch and filter modulation). It suggests an interactive bridge between the popular distorted sound of the electric guitar to digital sound modulation, maintaining the same expressions founded with playing a guitar.

### 3.3 *'Virtual Mixer'*

Since all the live instruments are connected directly to the sound-card and routed through Max patches, the live mixing can be made via UDP massages to the *Child* patch of each player. Therefore, we built the *Virtual mixer* patch to control the output sound (volume and filter) of each player without the use of physical mixing. In this way of virtual mixing, it is possible to maintain the same balance in rehearsals and in live performances.

## 4. AUDIO ROUTING

Since the audio signal is created with VSTs via Max/MSP, the audio routing which includes sending outputs and self monitors system for the players, can be managed directly through the laptops and the sound-cards. Figure 3 shows a scheme of the audio connections between the players and the outputs and inputs on stage. The audio signal is routed between the sound-cards via an S/PDIF cable which transmits two audio channels in one cable. In that way, the sound-cards are chained together and all the audio-signals are summed up to one stereo output (see figure 3) that is sent directly to the main output.

Routing the audio between the sound-cards also allows to use the headphones connections of each sound-card as a monitor for each player on stage (the monitors routing are shown by the red lines in figure 3). These monitors also transmit the metronome click for each player. This method where all the audio routing is managed by the band, allows the players to always play in high professional standards, regardless the stage's equipment and to achieve the same sounds and balance at every venue.

[2]`http://www.youtube.com/watch?v=es3Ue7_g5vk`

## 5. LIVE PERFORMANCE

Figure 4 shows an installation of a live performance and the organization on stage. In a typical performance, one laptop and sound-card unit was assigned to two players (guitar and bass), one unit to 3 players (vocal, guitar, electric drum set), and one unite to the *'conductor'* patch and another player (keyboard). The reliability of the approach in real-time is in high priority, and it was realized that after few rehearsals, the players were fully trusting the automation and the synchronization during the performance. This allows, the player can focus on the playing alone, and to deliver the music with natural expressions, and on the other side, giving the audience the ability to fully relate the performance for the following reasons: (a) The live playing are synchroized with background music. (b) Many of the synthesized sounds, and effects are assigned to physical gestures of the players and automatically changed without interrupting the players. (c) The sound and the mix is authentic and well identified with the band at every show.

### 5.1 Demonstrations

Two demonstrations of this approach are provided in this work. The first demonstration is a short scene recorded with a standard camera from the audience during a live show (`http://youtu.be/JBY-nHzlWpk`).

This authentic scene shows clearly how the band's playing and background samples are well synced with dramatic music style variations. During the scene, it is also shown how a virtual effect made via Max/MSP is scheduled automatically, and applied on the singer voice without a noticeable latency(in time 00m:48s).

Another video accompanies this article. This video shows how the patches communicate and trigger massages, and massages between the *Conductor* and the *Childs* patches (noting that in the video the *Child* patch is called *Autopress*, the previous name of the patch).

### 5.2 Performance Tests

Plugging the equipment (tested on a band with 6 players), takes 10-20 minutes. We made performance tests for the La-


**Figure 4: Installation on stage of the approach in live performance.**

**Table 1: Latency and CPU at background state, 1 guitar player, 1 vocal singer, and 2 players.**

| Buffer-Size | No vst<br>Latency(ms) / CPU(%) | 1 guitar player<br>(commercial vst)<br>Latency(ms) / CPU(%) | 1 vocal Player<br>(Virtual instrument)<br>Latency(ms)/ CPU(%) | 2 players<br>(2 vst instrument)<br>Latency(ms)/CPU(%) |
|---|---|---|---|---|
| 512 | 29 / 3-4 | 30 / 21-25 | 54 / 19-23 | 40.36 / 45 - 50 |
| 256 | 18 / 4-5 | 18.75 / 22-27 | 31 / 21-24 | 25.4 / 49- 57 |
| 128 | 14 / 4-6 | 16.6 / 23-30 | 19 / 25-30 | 17.6 / 51 - 60 |
| 64 | 9 / 5-6 | 13.5 / 26-32 | 13.3 / 29-34 | 15.2 / 53 - 64 |
| 32 | 7 / 9-10 | 9 / 32-42 | 11 / 38-43 | 11.6 / 55 - 70 |

tency and CPU on stage. The tests made for one and two players with a unit consists of laptop, 2.4Ghz core 2 duo Intel processor, and a portable Saffire pro 24DSP Focusrite sound-card. For the guitar player, we use a commercial VST to simulate physical amplifiers and Max/MSP 'home-made' virtual instrument (as done in live performance and in studio). For the vocal we use virtual effects made in Max/MSP (as used in live performance and in studio). The results are summed in table 1. The tests show that the latency can get to a maximum of 40.36ms on 512bit buffer size. Reducing the buffer size to 256bit allows comfortable performing with insignificant latency while keeping the CPU low enough to not cause sound glitches or distortions during the live performance.

## 6. CONCLUSION

We have demonstrated a reliable and low cost approach that utilizes laptops in live performance for live music bands. We have introduced the ability to bring the sound from the studio to the stage by synchronizing loops and samples, automatic changing presets of VST plug-ins , and managing 'Home-made' VSTs. Using the Max/MSP environment makes it possible to adapt this approach to every band and maintain the ability of the player to introduce new interactive interfaces which later can be combined in live performance. This helps the band to maintain creativity as well as professionalism on stage.

Moreover, by using a virtual environment of standard laptops and sound-cards, we show the ability to manage the audio signal routing at any venue with low costs and standard equipment (no need for mixer, monitor system, and amplifiers on stage). This advantage also makes the sound quality and volume balance close to the original intention of the band as in rehearsals or in the studio.

Despite the common use of Max/MSP in an experimental way, the performance tests shows that this approach is confidently can be used in live performance, and the "real life" test of many concerts in various venues suggests that this approach is reliable and practical for the standard band.

## 7. REFERENCES


[1] F. Blum. *Digital Interactive Installations*. VDM Verlag, 2007.

[2] A. Cont. Antescofo: Anticipatory synchronization and control of interactive parameters in computer music. In *International Computer Music Conference (ICMC)*, 2008.

[3] P. Cook. Sound production and modeling. *Computer Graphics and Applications, IEEE*, 22(4):23–27, 2002.

[4] R. Dannenberg. Current directions in computer music research. chapter Real-time Scheduling and Computer Accompaniment, pages 225–261. MIT Press, Cambridge, MA, USA, 1989.

[5] R. B. Dannenberg. New interfaces for popular music performance. In *NIME proceedings*, pages 130–135, New York, NY, USA, 2007. ACM.

[6] S. Emmerson. *Living Electronic Music*. Ashgate Publishing Company, Leicester, UK, 2007.

[7] D. Liang, G. Xia, and R. B. Dannenberg. A Framework for Coordination and Synchronization of Media. In *NIME proceedings*, pages 167–172, 2011.

[8] V. J. Manzo. *Living Electronic Music*. Oxford university press, New York, United states, 2011.

[9] J. Pakarinen and D. T. Yeh. A review of digital techniques for modeling vacuum-tube guitar amplifiers. *Computer Music Journal*, 33(2):85–100, June 2009.

[10] G. Sioros and C. Guedes. Automatic Rhythmic Performance in Max/MSP: the kin.rhythmicator. In *NIME proceedings*, pages 88–91, 2011.

[11] M. Stroppa. Live electronics or live music? towards a critique of interaction. *Contemporary Music Review*, 18(3):41–77, 1999.

[12] D. T. Yeh, J. S. Abel, A. Vladimirescu, and J. O. Smith. Numerical methods for simulation of guitar distortion circuits. *Computer Music Journal*, 32(2):23–42, 2008.