Synchronizing Spatially Distributed Musical Ensembles

Aristotelis Hadjakos, Axel Berndt, Simon Waloschek
Center of Music and Film Informatics (CeMFI)
University of Music Detmold / University of Applied Sciences OWL
{hadjakos,berndt,waloschek}@hfm-detmold.de

ABSTRACT

Spatially distributed musical ensembles play together while being distributed in space, e.g., in a park or in a historic building. Despite the distance between the musicians they should be able to play together with high synchronicity and perform complex rhythms (as far as the speed of sound permits). In this paper we propose systematic support of such ensembles based on electronic music stands that are synchronized to each other without using a permanent computer network or any network at all.

1. INTRODUCTION

First attempts to explicitly design spatiality in music can already be found in the Renaissance and Baroque area. Giovanni Gabrieli (1557-1612) positioned trumpet players on the side galleries of his church and at times alternated between the trumpet groups [1]. In the further course of music history such spatial concepts where artistically explored again and again, from Berlioz (1803-1869) in his Symphonie fantastique [2], where an oboist enters the concert hall while playing, up to todays artificial spatiality through the use of surround sound.

In this paper we explore how to support spatially distributed musical ensembles. Such ensembles could, e.g., play in a park, making it possible for the audience to explore the piece by moving around. Or the ensemble members could be placed in co-located rooms in a building or spread out in the lobby of a concert hall during the intermission. Due to the distance, the musician's own instrument will usually drown out the sound of the other ensemble members, making it difficult to play synchronously. Furthermore, the musicians may not be able to see each other, making visual cues impossible. Previous realizations have relied on conductors that were visible for all ensemble members (e.g., to synchronize the musicians in the orchestra pit and the performers on stage) or they have relied on click tracks that were transmitted over wireless headphones.

To understand how click tracks are currently created, we performed informal interviews with musicians, composers, and electronic music artists. They used a variety of non-

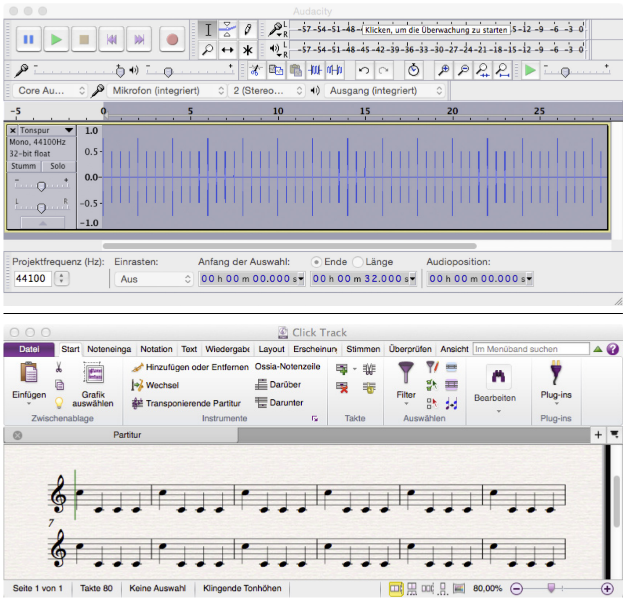


Figure 1. Audio editing (top) and music notation tools (bottom) are commonly used to create click tracks.

specialized software tools. In particular they used audio editing tools like Audacity or music notation tools for that purpose (Figure 1). We created WebMaestro, a web-based click track editor and player (see Section 3) to make it easier to create click tracks and provide better support for rehearsal situations. In addition to auditory cues, we wanted to provide the musicians with a visual display that gives them a representation of the musical beat and the current position in the piece. Since the musicians do not hear each other well enough at all times, this makes sure that the performance does not fall apart when, e.g., one musician miscounts rest bars. Our interactive music stand, the "M-Sync Player" (see Section 4) provides visual as well as auditory cues for synchronizing spatially distributed ensembles.

As a wired or wireless network may not always be present (e.g, outside in a park, in a historic building) or accessible (e.g., in a big concert venue), we were interested in synchronizing the M-Sync Players without having to rely on a network. We discuss (Section 5) and evaluate (Section 6) different synchronization strategies.

2. RELATED WORK

Many very different projects are faced with the situation of a distributed music-making and its key challenge of affording inner-ensemble communication. Besides the use of synchronized click tracks and low-latency audio transmission, this situation motivates the augmentation of traditional music stands and the use of networked digital music stands as platform to mediate communicative cues between the players. This section pinpoints some representative works in the field of distributed music-making to give an impression of the variety of scenarios. Then it introduces digital music stands and related research.

2.1 Networked Music-Making, Performance, Tuition

Networked music-making often requires a more or less complex hard- and software setup. With the JamBerry Meier et al. present a very compact stand-alone device, which is based on the Raspberry Pi, extended by a high-quality audio interface and a touchscreen [3]. The JamBerry focusses on the low latency audio transmission. Further means for communication between the players are not implemented so far.

Inner-ensemble communication is a complex and often very subtle combination of visual and auditory cues. Typical examples are facial expressions, body movements and breathing. Schober [4] provides an overview of such coordinating cues and discusses their translation into virtual environments where players can be collocated even if physically distant. Distributed music rehearsal systems are presented by Konstantas et al. [5] and Alexandraki et al. [6].

Duffy & Healey [7] compare music tuition in collocated and video mediated situations. Among other observations, they point out the importance and efficiency of gesture interaction on the shared music score which gets lost in the video mediated setup: "The importance of the shared score to lesson interaction was evidenced by problems managing interaction such as turn control when participants were separated and could no longer share the same physical representation of the music." They motivate "to involve an interactive visual layer over a digitised representation of the physical score, which shows the separated participants where each person is gesturing on the music. Ideally both participants should be able to mark their layer in a way which allows the student to take an annotated copy away, and return with it for the next lesson. There should be a way for the tutor to communicate intent to interrupt the student's performance through visualization of gestures on the music."

A dynamic digital medium such as a digital music stand can display not only static scores. Brown [8] generates the score live at its performance, which requires the human player to have great sight-reading skills. Freeman's [9] interactive realtime score generation and distribution to live performing players goes even a step further. Here, the audience can interactively influence the score generation process while it is being performed by human players. Not only can the composer be replaced by virtual instances but also parts of the ensemble, letting humans play together with computer instruments. A fully automated digital and spatially distributed music ensemble is described by Kim et al. [10]. In today's concert practice such cooperative human-computer music performances are typically coordinated by click tracks. These force the human to follow the computer. Some approaches also make the virtual performer responsive to human musicians, such as Liang et al.'s framework for coordination and synchronization of media [11].

2.2 Digital Music Stand Technology

The typical functionality of electronic music stands, besides score display, comprises the management of a sheet music database, the possibility of adding annotations and performance instructions, metronome and pitch tuner integration, and hands-free page turning (a key feature of electronic music stands, traditionally triggered via foot pedal).

Commercial products and patents exist for more than a decade now, like the *eStand Electronic Music Stand* [1] (a review of the eStand is given by Cross [12]), the *MusicPad Pro* and its successor the *MusicOne* stand [2], and patents like Kumarova's digital music stand [13]. Besides these commercial instances several academic research projects deal with the development of electronic/digital music stands and related issues, like the *Espresso* digital music stand of Bell et al. [14]. In one of the first concept papers on digital music stands Graefe et al. [15] introduced the *muse* concept that never came to a full technical implementation but inspired many subsequent projects.

The MICON system is a music stand for interactive conducting of orchestral audio and video recordings [16]. The system is part of an exhibit with a focus on non-professional users. The exhibit implements a conducting gesture recognition which is connected to video and audio time stretching so that the music and the video of the orchestra react to the user's gestures. The MICON features several different score visualizations, automatic page turning animations, and an animated visual cueing system that indicates the current playback position within the score. In his study, Picking [17] already noted that such visual cues are very popular. MICON's beat visualization is a potential candidate for a visual click track.

With their Multimodal Music Stand, Bell et al. [18] introduced an augmented traditional music stand that seamlessly blends into a musical instrument. Equipped with microphones, cameras, and electronic field sensors the stand "augments the performance space, rather than the instrument itself, allowing touch-free sensing and the ability to capture the expressive bodily movements of the performer" [18]. The sensor data may provide a prospective starting point to integrate a new approach to inter-player communication.

Communication capabilities within the orchestra, i.e., with other music stands, were already part of the muse concept [15]. The MOODS (Music Object-Oriented Distributed System) is designed to equip a whole orchestra [19] and features corresponding networking capabilities. It interfaces with a score database, automatically generates parts, allows cooperative editing, managing/versioning, and distribution of the scores throughout the orchestra. Similar networking capabilities are described by Romero

[1] published by eStand, Inc., `http://www.estand.com` (last access: Apr. 2015)

[2] both, MusicPad Pro and Music One, are published by SightRead Ltd., `http://www.sightread.co.uk` (last access: Apr. 2015)

& Fitzpatrick [20] and Connick [21]. Laundry's developments on the music typesetting and annotation of music scores complements this work [22].

2.3 Further Contextual Research and Studies

Contextual studies on electronic/digital music stands has been performed by Picking [17] amongst others. Picking compares music reading on paper with music reading on screen (static and animated). He notes that the study participants preferred an animated score presentation over the static and paper presentation. The use of cursor-like markings that indicated the current (playback) position in the music turned out to be very popular among the participants. Here, research on automated score following and music alignment provides the potential technical complement [23–25]. These indications are most interesting for player synchronization tasks and serve as a replacement of traditional visual click tracks.

Bell et al. [26] investigate two further core aspects of the visual score presentation: page turning animation and image size. A user study compared six page turning variants, including cascaded blending, horizontal scrolling, and vertical scrolling [27], of which the participants preferred to keep control over changes instead of fully automatic animations. A similar experiment is described by Blinov [28]. In their image size study Bell et al. did not observe significant differences in the participants' performances while proofreading on large and small scales. But the participants favored the larger scale for convenience reasons.

Kosakaya et al. refined their page turning scheme via time delays based on glance analyses [29]. The muse concept employs a microphone for audio-to-score alignment to estimate appropriate page turning moments automatically. Research and development on page turning are continued until today [27, 30, 31].

3. WEBMAESTRO

WebMaestro [3] is a web-based click track editor and player. It is a self-contained application and can be used to edit and play back click tracks instead of using non-specialized software like audio editors or music notation editors for this task (see Section 1). WebMaestro can further be used as a pure editor, preparing a representation that is played back by synchronized M-Sync Players (see Section 4). An overview of WebMaestro's user interface is shown in Figure 2. In 2014 WebMaestro was used at the Internationale Ferienkurse für Neue Musik in Darmstadt for the rehearsal and performance of the piece à tue-tête for nine spatially distributed wind players by Fabien Lévy. The piece was performed by the ensemble "Klangforum Wien". [4]

3.1 Models and Tempo

Our solution uses two models: the editor model and the playback model. The editor model represents the parts that are relevant for editing a click track, including time signatures, tempo, accelerando and ritardando, etc. This is the model that the composer generates and modifies with the help of WebMaestro's user interface. The playback model on the other hand addresses the timed succession of events. In particular, tempo and tempo changes are boiled down to delta times (time differences or inter-onset intervals) between successive events.

The editor model represents the piece as a sequence of sections. A section is a sequence of bars with the same musical meter and the same tempo. The following code example represents a section with a tempo change:

```
{
  "bars": "5-8",
  "signature": "3/4",
  "bpm": "60-96",
  "tempoChange": {
    "begin": "6:2",
    "end": "9:1",
    "curve": "Natural"
  }
}
```

The section extends from bar 5 to bar 9, with a 3/4 time signature, and a tempo change that begins on the second beat in measure six, with a "natural" (quadratic interpolation) tempo curve. We use quadratic interpolation by default as this has been shown to be close to what musicians typically do [32].

The playback model is a representation that is simple to render. Similar to the MIDI file format it is based on delta times, i.e., time differences between subsequent events. Each event is defined by its delta time, its type and its content. The following representation denotes the second beat in a 4/4 time signature. It has a delta time of 1 second to the previous event:

```
{
  "delta": 1,
  "type": "beat",
  "content": "2/4"
}
```

Having a separate playback model greatly simplifies the implementation of the M-Sync Player, since tempo calculations are already contained in the delta times. This is important as we plan to port the M-Sync Player to different platforms including Windows, Android and iOS.

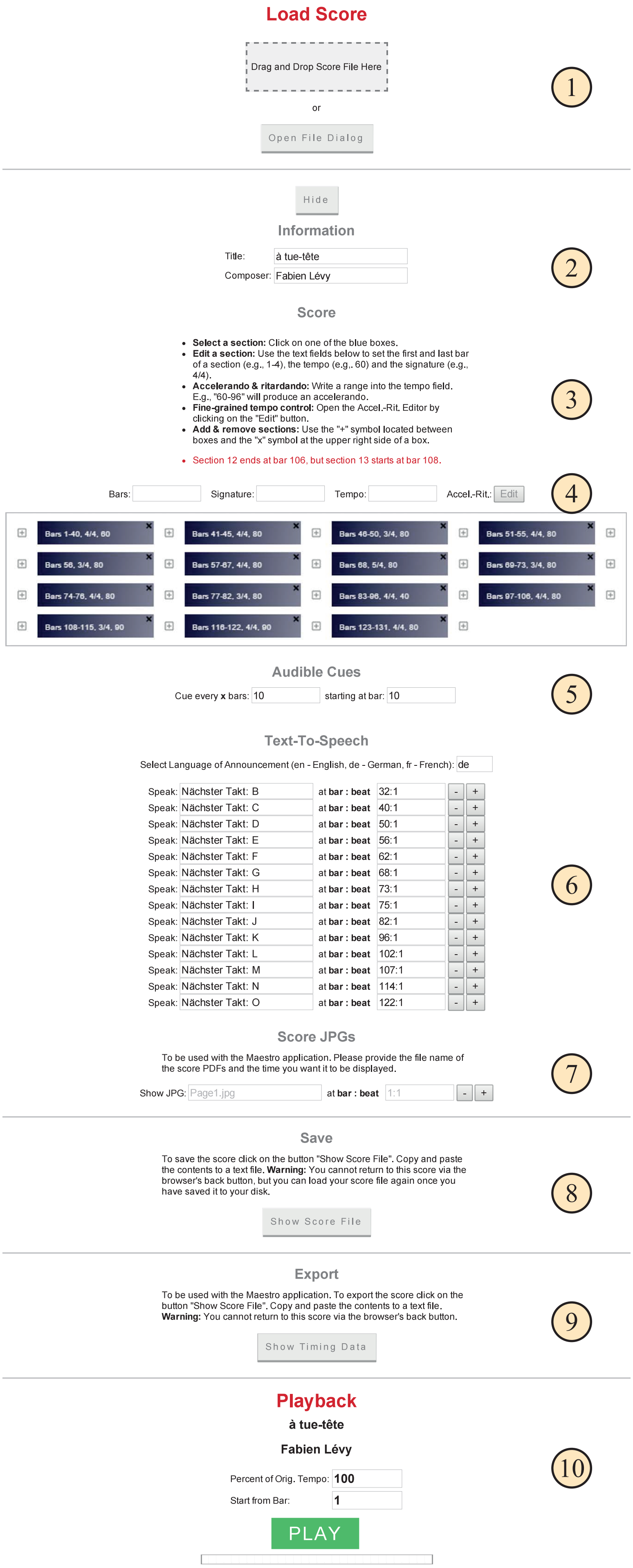
3.2 Implementation

WebMaestro's audio output was realized using the Web Audio API. Audio samples and JPEGs were encoded with base64 directly as JavaScript strings contained in the HTML file. This makes WebMaestro usable without network as a single self-contained HTML file, which is sometimes useful in rehearsal situations without network access.

4. M-SYNC PLAYER

The M-Sync Player (see Figure 3) displays and advances the score, visualizes the musical beat and also plays the sound of a metronome. All of that is done synchronously

[3] http://zemfi.de/downloads/WebMaestro.html
[4] http://www.klangforum.at/



Explanations:

1. Score files can be loaded either by dropping them onto the marked box area or by using a dialog.

2. The title and the name of the composer can be entered.

3. A short manual is provided that describes the basic functionality of the editor. Common user errors are displayed immediately as red warnings.

4. Sections are defined by providing the bar numbers as well as the signature and the new tempo. Accelerando and ritardando can be specified by providing a tempo range, e.g., 60-96, in the tempo field. An Accel.-Rit.-Editor provides fine-grained control over the tempo change.

5. Audible cues can be generated periodically every n bars.

6. A web-based speech synthesis may be used to give vocal cues at given bars and beats.

7. The M-Sync Player is able to display the score with automatic page turning if the corresponding JPEG files are provided.

8. Edited click tracks can be saved as plain text files and loaded at a later time again.

9. The timing data can be exported for later usage with the M-Sync Player.

10. Finally, the click track can be played directly in the web browser. For rehearsals, the user can select from which bar to start and change the overall tempo of the entire playback.

Figure 2. The user interface of the WebMaestro

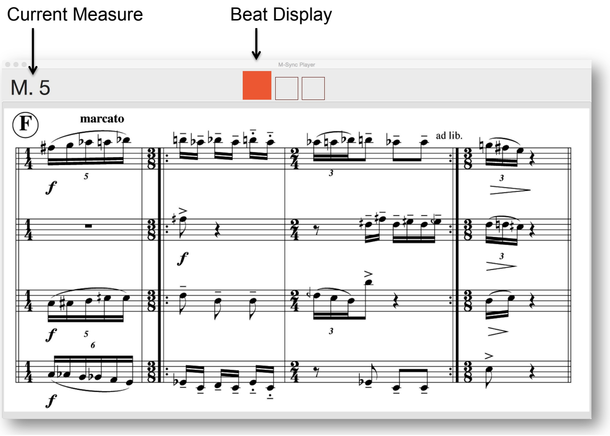


Figure 3. The M-Sync Player

on all computers using one of the synchronization methods described in Section 5. To display the content, the M-Sync Player uses the playback model generated by WebMaestro. The M-Sync Player provides a beat display that indicates the current beat with a filled box while all other beats are shown as outlined boxes. The boxes are relatively large to make it easier for the musician to follow those visual cues in peripheral vision while looking at the musical score below. At the upper left, the current beat is displayed. Together with the automatic advancement of the score, this ensures that the performance does not fall apart if one player looses track of the current position, e.g., by miscounting rest bars. Such errors can otherwise be difficult to spot since the ensemble members may not hear each other well enough in the targeted distributed situations. In addition to the visual cues, the M-Sync Player also generates auditory cues with separate sounds for the first and the following beats of a bar.

Furthermore, the M-Sync Player generates OSC messages that can be received by other applications on the same machine. This can be used to synchronize electronic music, e.g., generated by a Max patch, or a visualization, e.g., generated by Processing, to the performance of the ensemble.

5. SYNCHRONIZATION

The performance of spatially distributed music can take place in parks, historic buildings or big concert venues where it may be difficult to get access to a wired or wireless computer network. Therefore, we examine different synchronization options that require no (Distributed Button, radio time signals, GPS) or no permanent (NTP) network connection. In order to display the score and play the click track simultaneously on multiple computers, their clocks have to be synchronized with great accuracy. However, computer clocks may not only deviate by a static time interval but the clocks may also drift due to slightly different speeds (see Figure 4).

We distinguish one-shot synchronization and continuous synchronization. In one-shot synchronization, the systems are synchronized once, i.e. before the performance has started. In continuous synchronization, the computers are

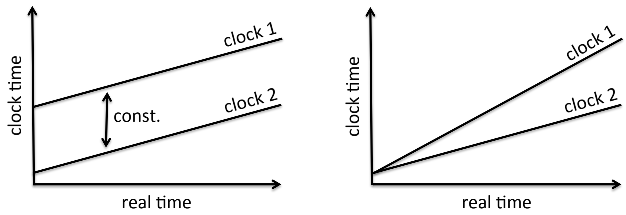


Figure 4. Clock offset (left): the clock readouts differ by a constant amount. Drift (right): Although the clocks are initially synchronous they continuously drift apart since one clock runs faster than the other.

connected to an external clock that corrects the computer clock in regular intervals.

5.1 One-Shot Synchronization

5.1.1 NTP

The Network Time Protocol (NTP) is a protocol to synchronize computers via the Internet. Clock synchronization is acquired by exchanging four messages. For each exchanged message the sender and the receiver measure the send and reception time with their local unsynchronized clock. Based on this information, the offset between the two clocks and the transmission delay can be calculated, making it possible to adjust the client's clock to the right time. However, the transmission delay has to equal in both directions (or close to equal) for NTP to work properly.

5.1.2 Distributed Button

For the user, the Distributed Button is a big box with USB connectors and a button on top (see Figure 5). First, the users connect their computers to the box and then one user presses the button on top. This event is received on all computers simultaneously and used to synchronize all M-Sync Players.

Figure 5. The Distributed Button

5.2 Continuous Synchronization

5.2.1 Radio Time Signals

Radio time signals transport an encoding of the current time over radio waves. Typically, amplitude or frequency modulation is used to encode the bit representation of date and time on long, medium or short waves. Radio time signals are available all over the world.

Being located in Europe, we used DCF77 signals. DCF77 is a long wave radio time station located near Frankfurt, Germany. It provides radio time signals that can be received in large parts of Europe. DCF77 uses amplitude modulation and generates pulses of varying length each second. The bits are encoded by changes in pulse lengths: A 100 ms pulse is a zero and a 200 ms pulse a one. The bits encode the current date and time and also provide parity bits, which provide error detection to single bit errors. No pulse is sent on the last second of each minute. Then the next pulse indicates the beginning of a new minute. We used an Arduino shield with a DCF77 receiver [5] (see Figure 6).

5.2.2 GPS

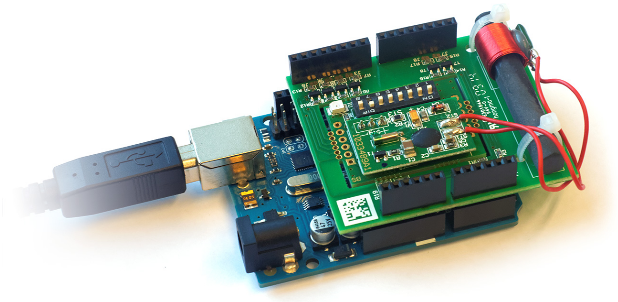
The Global Positioning System (GPS) is based on a multitude of satellites orbiting Earth. Each satellite sends its position in space together with a highly accurate time stamp obtained from an onboard atomic clock. The signal spreads out with the propagation speed of light and eventually reaches the receiver. The intersection of those signal spheres from multiple satellites determines the position of the receiver. In order to calculate this intersection point however, the receiver needs to have a very accurate clock in order to determine the distance from a satellite as a function of the time stamp from the satellite and reception time. Since GPS receivers need to be cheap, a clock signal is reconstructed from the satellite signals. In essence, four-dimensional hyper-spheres originating from multiple satellites are intersected to calculate 3D position and the current time. While GPS users are typically more interested in the position signal, the time signal can be used to synchronize spatially distributed musical ensembles.

GPS receivers are available at relatively low cost and compatible to popular physical computing platforms. We used two GROVE GPS sensor modules [6] (version 1.1 and 1.2) and interfaced them to an Arduino Leonardo using a SPINE shield (see Figure 6).

6. EVALUATION

6.1 Procedure

We wanted to assess the synchronization accuracy that can be achieved with the different synchronization methods. We employed the following evaluation procedure: Two M-Sync Players running on two different computers were synchronized with one of the said methods. The M-Sync Players were triggered to begin playing at a particular time and rendered a half-hour long steady 60 bpm pulse in 4/4 time. We recorded the audio output of the two M-Sync Players using a custom-made cable with two singal-in headphone connectors and one signal-out headphone connector. The signal-in connectors where connected to the headphone outputs of the two computers and the signal-out connector was connected to the line-in of a separate computer that



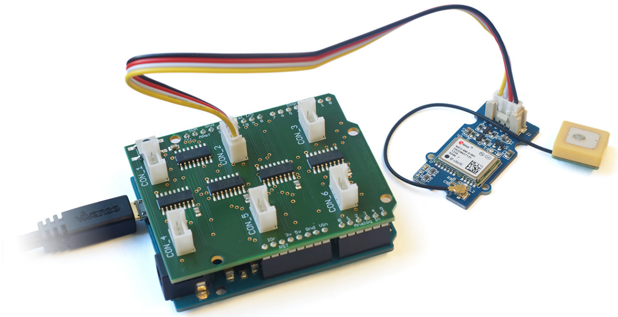


Figure 6. An Arduino with DCF77 shield (top) and an Arduino with a SPINE shield connected to a GROVE GPS module (bottom)

we used as a recording device. This provided us with a stereo signal where the left channel originates from the M-Sync Player of one computer and the right channel from the other. In the experiments we used a MacBook Pro (Retina, 15”, mid 2014) and a MacBook Pro (13”, mid 2012). Both computers where running OS 10.10.2.

6.2 Results

We examined the timing deviations between synchronized M-Sync Players. For this purpose the time difference of the onsets on the left audio channel and the corresponding onset on the right channel were determined with a MATLAB script, which detected beat onsets with thresholding.

One-shot synchronization: For NTP, we manually initiated the computers to synchronize themselves to an NTP server on the Internet before we started the M-Sync Players. While the Distributed Button provides more accurate clock offset compensation than NTP, i.e., the computers start out with less deviation, the computer clocks drift apart with increasing differences of about 3 ms/min (see Figure 7). This drift makes it problematic to perform pieces that are more than a few minutes long. Instead of using the clock offered by the operating system, we then measured time by counting the number of samples that are sent to the built-in sound card at a fixed rate of 44.1 kHz. The drift rate sank to about 1 ms/min. We then measured the overall deviation after 30 minutes and computed the (almost) constant deviation of the audio rates of the two computers. By compensating for that exact amount, we were able to achieve a drift rate of about 0.0367 ms/min between the two M-Sync Player (see Figure 8). Using this method, the two M-Sync Players drift only about 1 ms apart after 30 minutes, which is well below what is musically relevant.

[5] http://bit.ly/1CtXOR8
[6] http://www.seeedstudio.com/wiki/Grove_-_GPS

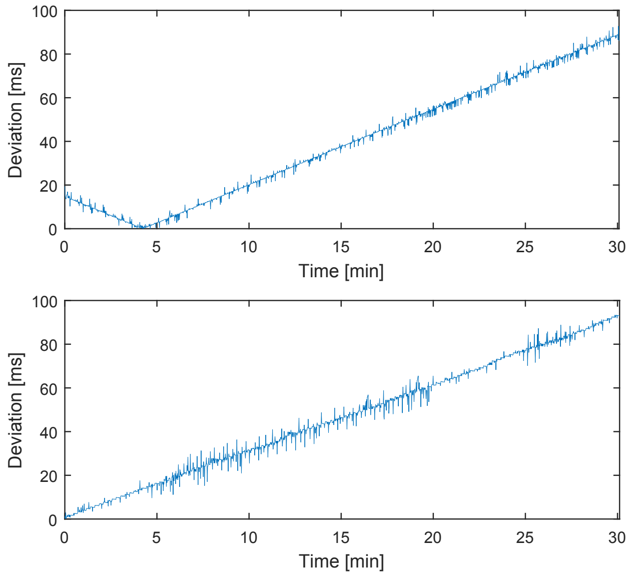


Figure 7. One-shot synchronization: NTP (top) and Distributed Button (bottom). The Distributed Button provides a better clock offset compensation than NTP.

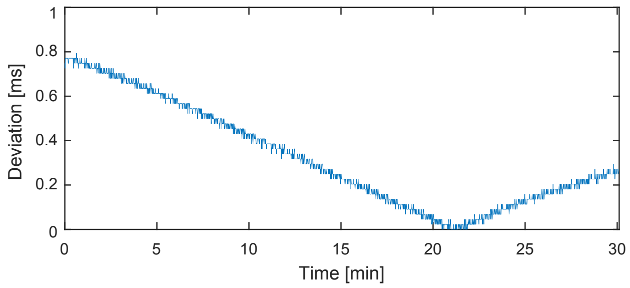


Figure 8. Interchannel deviation using the Distributed Button and the internal audio clock.

Continuous synchronization: We then examined GPS and DCF77-based synchronization. The GPS modules we used did not drift but had substantial timing irregularities (see Figure 9, top), making them unusable for our purposes. However, we observed distinct differences between different GPS modules so that there probably is a suited GPS module, which we have not identified yet. DCF77 on the other hand provides good synchronization with a maximum deviation of 11.43 ms and a mean deviation of 2.4 ms without introducing any long-term drift (see Figure 9, bottom).

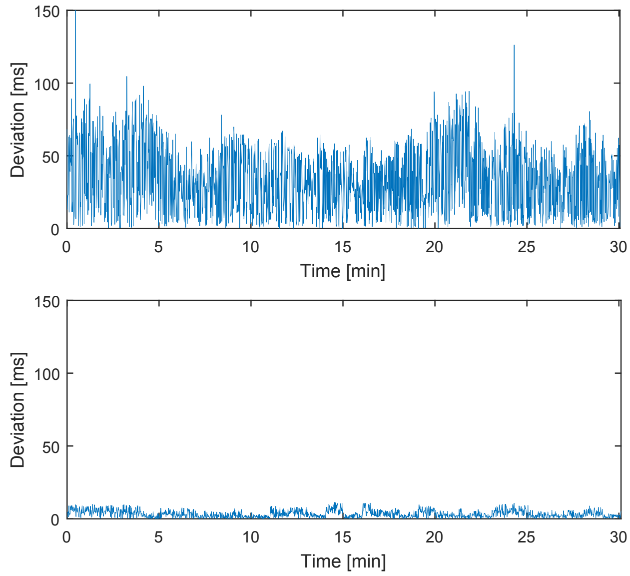


Figure 9. Continuous synchronization: GPS (top) and DCF77 (bottom). In our experiments, DCF77-based synchronization worked significantly better (about one order of magnitude).

7. CONCLUSION

In this paper, we have explored how to systematically support spatially distributed musical ensembles. The WebMaestro click track editor and player lets the user define and play back complex click tracks with changes in tempo & meter, accelerando & ritardando together with text-to-speech announcements. Furthermore, WebMaestro lets the user export a playback model, which can be used in conjunction with the M-Sync Player to visualize the musical score and provide visual cues for beats together with auditory metronome beats. In many places where one would want to perform with a spatially distributed musical ensemble, it is oftentimes difficult to get access to a wired or wireless computer network. Therefore, we explored and evaluated a variety of synchronization methods that can be realized without (or without permanent) network access. The Distributed Button (best one-shot synchronization) and radio time signal synchronization (best continuous synchronization) turned out to be the best options. Additionally, we experienced that the internal clock sources of audio interfaces built into computers are more accurate than regular system clocks.

8. REFERENCES

[1] G. Gabrieli, *Sacrae Symphoniae*. Venice, Italy: Apud Angelum Gardanum, 1597, vol. 1.

[2] H. Berlioz, *Symphonie fantastique*, N. Temperley, Ed. Kassel, Germany: Bärenreiter-Verlag, 2012.

[3] F. Meier, M. Fink, and U. Zölzer, "The JamBerry—A Stand-Alone Device for Networked Music Performance Based on the Raspberry Pi," in *Linux Audio Conference*, vol. 2014, 2014.

[4] M. F. Schober, "Virtual environments for creative work in collaborative music-making," *Virtual Reality*, vol. 10, no. 2, pp. 85–94, 2006.

[5] D. Konstantas, Y. Orlarey, O. Carbonel, and S. Gibbs, "The distributed musical rehearsal environment," *IEEE Multimedia*, vol. 6, no. 3, pp. 54–64, 1999.

[6] C. Alexandraki and D. Akoumianakis, "Exploring new perspectives in network music performance: the DIAMOUSES framework," *Computer Music Journal*, vol. 34, no. 2, pp. 66–83, 2010.

[7] S. Duffy and P. G. Healey, "Spatial co-ordination in music tuition," in *Proceedings of the 34th annual conference of the cognitive science society*. Cognitive Science Society Sapporo, 2012, pp. 1512–1517.

[8] A. R. Brown, "Generative music in live performance," in *Australian Computer Music Conference*. Brisbane, Australia: Australasian Computer Music Association, 2005, pp. 23–26.

[9] J. Freeman, "Extreme sight-reading, mediated expression, and audience participation: Real-time music notation in live performance," *Computer Music Journal*, vol. 32, no. 3, pp. 25–41, 2008.

[10] D.-H. Kim, E. Henrich, C. Im, M.-C. Kim, S.-J. Kim, Y. Li, S. Liu, S.-M. Yoo, L.-C. Zheng, Q. Zhou *et al.*, "Distributed computing based streaming and play of music ensemble realized through TMO programming," in *10th IEEE International Workshop on Object-Oriented Real-Time Dependable Systems, 2005. WORDS 2005.* IEEE, 2005, pp. 129–136.

[11] D. Liang, G. Xia, and R. B. Dannenberg, "A framework for coordination and synchronization of media," in *Proc. of the Int. Conf. on New Interfaces for Musical Expression (NIME11)*, 2011, pp. 167–172.

[12] J. Cross, "eStand TM Electronic Music Stand (review)," *Notes*, vol. 60, no. 3, pp. 754–756, 2004.

[13] M. Kumarova, "Digital music stand," Aug. 2007, US Patent App. 11/587,180. [Online]. Available: http://www.google.com/patents/US20070175316

[14] T. Bell, D. Blizzard, R. D. Green, and D. Bainbridge, "Design of a Digital Music Stand," in *ISMIR*, 2005, pp. 430–433.

[15] O. Dasna, C. Graefe, J. Maguire, and D. Wahila, "muse: A Digital Music Stand for Symphony Musicians," *interactions*, vol. 3, no. 3, pp. 26–35, May/June 1996.

[16] J. Borchers, A. Hadjakos, and M. Mühlhäuser, "MICON: A Music Stand for Interactive Conducting," in *Proc. of the 2006 Int. Conf. on New Interfaces for Musical Expression (NIME06)*. Paris, France: IRCAM – Centre Pompidou, 2006, pp. 254–259.

[17] R. Picking, "Reading music from screens vs paper," *Behaviour & Information Technology*, vol. 16, no. 2, pp. 72–78, 1997.

[18] B. Bell, J. Kleban, D. Overholt, L. Putnam, J. Thompson, and J. Kuchera-Morin, "The multimodal music stand," in *Proceedings of the 7th International Conference on New Interfaces for Musical Expression*, ser. NIME '07. New York, NY, USA: ACM, 2007, pp. 62–65. [Online]. Available: http://doi.acm.org/10.1145/1279740.1279750

[19] P. Bellini, F. Fioravanti, and P. Nesi, "Managing music in orchestras," *Computer*, vol. 32, no. 9, pp. 26–34, 1999.

[20] E. Romero and G. Fitzpatrick, "Networked electronic music display stands," June 1998, US Patent 5,760,323.

[21] H. Connick, "System and method for coordinating music display among players in an orchestra," Feb. 2002, US Patent 6,348,648. [Online]. Available: http://www.google.com/patents/US6348648

[22] B. A. Laundry, "Sheet Music Unbound: A fluid approach to sheet music display and annotation on a multi-touch screen," Ph.D. dissertation, University of Waikato, 2011.

[23] N. Orio, S. Lemouton, and D. Schwarz, "Score following: State of the art and new developments," in *Proceedings of the 2003 Int. Conf. on New Interfaces for Musical Expression (NIME03)*. National University of Singapore, 2003, pp. 36–41.

[24] R. B. Dannenberg and C. Raphael, "Music score alignment and computer accompaniment," *Communications of the ACM*, vol. 49, no. 8, pp. 38–43, 2006.

[25] V. Thomas, C. Fremerey, M. Müller, and M. Clausen, "Linking Sheet Music and Audio—Challenges and New Approaches," in *Multimodal Music Processing*, ser. Dagstuhl Follow-Ups, Dagstuhl, Germany, 2012, vol. 3, pp. 1–22.

[26] T. Bell, A. Church, J. Mc Pherson, and D. Bainbridge, "Page turning and image size in digital music stand," in *International Computer Music Conference*, 2005.

[27] J. McPherson, "Page turning—Score Automation for Musicians," *Honours project report, Department of Computer Science, University of Canterbury, Christchurch, NZ*, 1999.

[28] A. Blinov, "An interaction study of a digital music stand," *Honours project report, Department of Computer Science and Software Engineering*, 2007.

[29] J. Kosakaya, Y. Takii, M. Kizaki, A. Esashi, and T. Kiryu, "Research and evaluation of a performer-friendly electronic music stand," in *Proc. of the Int. Conf. on Active Media Technology (AMT) 2005*. IEEE, 2005, pp. 11–15.

[30] J. Pagwiwoko, "Improvements To A Digital Music Stand," Master's thesis, University of Canterbury, Christchurch, New Zealand, Nov. 2008.

[31] I. Yasue, K. Susumu, and F. Yositomo, "Design and production of an electronic musical score system to reduce the load of page turning for wind orchestra," in *IEEE 13th International Conference on Cognitive Informatics & Cognitive Computing (ICCI* CC)*. IEEE, 2014, pp. 242–246.

[32] A. Friberg and J. Sundberg, "Does Music Performance Allude to Locomotion? A Model of Final Ritardandi Derived from Measurements of Stopping Runners," *The Journal of the Acoustical Society of America*, vol. 105, no. 3, pp. 1469–1484, March 1999.