Dominik Schlienger and Victor Khashchanskiy

University of the Arts Helsinki
Sibelius Academy
PO Box 30, FI-00097 Uniarts
dominik.schlienger@uniarts.fi,
victor@victorx.eu

Immersive Spatial Interactivity in Sonic Arts: The Acoustic Localization Positioning System

Abstract: The Acoustic Localization Positioning System is the outcome of several years of participatory development with musicians and artists having a stake in sonic arts, collaboratively aiming for nonobtrusive tracking and indoors positioning technology that facilitates spatial interaction and immersion. Based on previous work on application scenarios for spatial reproduction of moving sound sources and the conception of the kinaesthetic interface, a tracking system for spatially interactive sonic arts is presented here. It is an open-source implementation in the form of a stand-alone application and associated Max patches. The implementation uses off-the-shelf, ubiquitous technology. Based on the findings of tests and experiments conducted in extensive creative workshops, we show how the approach addresses several technical problems and overcomes some typical obstacles to immersion in spatially interactive applications in sonic arts.

The technological developments in global navigation satellite systems; indoor and local positioning systems using radio signals, ultrasound, optical motion tracking; inertial reference units like gyroscopes and acceleration meters; and localization services on smartphones, etc., present an array of conceptual possibilities for use in sonic and performance art. Tracking technology for musical interaction is a perennial subject in works presented at conferences like the International Conference on New Interfaces for Musical Expression (NIME) and the International Conference on Live Interfaces. Suitable technologies like motion capture are expensive, however. Generally available technologies, like the global positioning system (GPS) and hybrid approaches using smartphone technologies, have, to our knowledge, rarely been developed to provide the low latency required for musical expression. In practice this has not hindered developers and musicians from coming up with a plethora of creative solutions, from "circuit-bending" interactive video gaming consoles to building case-specific hardware from scratch.

The project that engendered this article and provided its rationale was an art installation in which the spatial position of a participant controls a musical parameter. The installation took place in June 2010 at the 470 Degrees Graduate Show, held at the University of the West of England in Bristol. By changing the participant's position within the room, some parameter of sound or musical parameter is affected. Parameters of sound are, for instance, amplitude, pitch, duration, timbre, and overtones. Musical parameters can include melody, harmony, rhythm, texture, expression, dynamics, tempo, and articulation. As the project was initiated by technophile artists rather than by engineers, and because it was financed on a shoestring, only readily available, ubiquitous technology was ever considered. After fruitless experiments with Bluetooth signal-strength measurements, the most promising technology appeared to be techniques using acoustic localization (AL) for sound just above the frequency range audible to humans, for the reason that in the envisaged setup, loudspeakers would already be part of the system, and unobtrusive small microphones could easily be introduced at low cost.

Acoustic localization techniques can be used for tracking and positioning purposes by measuring the time it takes a sound to travel from a source to a sensor. As the speed of sound through air is known (343 m/sec), the measurement can be used to estimate the distance, according to the relationship $d = c \times t$, where d is the distance a sound wave travels, c is the speed of the sound wave, and t is time. Yet, despite the simplicity of the principle, it has rarely been implemented for artistic purposes. Intrigued by the clear potential for low-cost solutions and following Occam's argument that entities should not be multiplied beyond necessity,

Computer Music Journal, 45:2, pp. 24–47, Summer 2021
doi:10.1162/COMJ_a_00605

we found the choice of AL to be utterly compelling, if tracking is required in a system constituted by loudspeakers and microphones.

Subsequently, a plan formed to develop a tool, based on these principles, that would not only meet the requirements for said artistic idea, but also be a more generally useful utility for other artistic projects in the field of sonic and or performance arts. The tracking system and the applications presented here were developed concurrently. This happened in an experimental artistic practice which came into being as a part of the development process. From within this practice of participatory design, it was possible to define more general requirements, culminating in the concise notion of the *kinaesthetic interface* (discussed in more detail below), which now provides a benchmark of sorts: It allows testing against clear qualitative and quantitative parameters.

Does the technology developed fill the gap encountered when looking for the suitable technology for the artistic project we presented in Bristol in 2010? And more generally, does it provide a useful tool for the community of artists working with spatial interaction in the sonic arts? In answer to these questions, we aim to demonstrate how the algorithm, which we call the Acoustic Localization Positioning System (ALPS), in conjunction with the Autopan Max patch we developed, helps to solve a series of problems often encountered in spatially interactive improvisation: Whereas participants with acoustic instruments can use their spatial position within a performance space as a musical parameter, electronic instruments generally are statically bound to the position of the loudspeakers. By automatically panning the electronic sound source to the position of players, their location becomes an interactive element too, on par with the players of acoustic instruments. What is more, this approach also allows spatial trajectories to become musical narratives in electronic improvisations.

Background

The positioning technology commonly applied for spatially interactive sonic arts is optical tracking, for example, motion capture. Some examples thereof are described in Nymoen, Skogstad, and Jensenius (2011), Dobrian and Bevilacqua (2003), and Bazoge et al. (2019). Hacks of the interactive video gaming consoles Kinect (Şentürk et al. 2012; Trail et al. 2012) and Wii (Peng and Gerhard 2009) also use optical means of tracking. For applications that do not require absolute positions, "dead reckoning" methods (i.e., methods of calculating one's current position by using a previously determined position, or fix) can be used, as can inertial methods using accelerometers, gyroscopes, and magnetic compass. These tend to map smaller movements to a process, as does the data glove by Mitchell and Heap (2011). The proceedings of the annual conference on Indoors Positioning and Indoors Navigation provide examples for many possible technologies (see http://ipin-conference.org for details). Besides the classifications in Schlienger and Tervo (2014), a summary of positioning technologies can be found in Hightower and Borriello (2001) and more recently in Brena et al. (2017).

Every year, the proceedings of the NIME conference provide examples of applications using positioning or tracking technology. An extensive summary of these can be found in a review of over 80 applications presented at NIME between 2001 and 2013 (Schlienger and Tervo 2014). By comparing these applications' requirements with the theoretical possibilities of AL, it is demonstrated that AL could have provided feasible alternatives for many of the applications at lower cost.

In summary, for this project, AL was chosen for the following reasons: First, AL works with off-the-shelf loudspeakers, which, in the scenario in question, are already present for the diffusion of content audio (e.g., music or speech). Besides a microphone, little additional equipment is needed, as the same loudspeaker can be used for tracking and content audio. Second, when maintaining the identity of the tracked device is required, AL—along with other sender–receiver technologies like radio frequency identification (RFID)—can provide this more reliably than optical systems. Third, the tracked device is unobtrusive: Omnidirectional condenser microphones of the lavalier type (worn by actors on stage, musical theater, and opera)

Table 1. Quantitative Requirements on ALPS

Accuracy of localization	≤0.3 m
Update rates	>20 Hz (perceptually continuous)
Coverage	10–1000 m^2
Latency	<20 msec

often measure under 5 mm in diameter. Finally, in comparison to motion capture, there are fewer line-of-sight (LOS) issues in AL: Obstructions in the path between a loudspeaker and a microphone need to be large in comparison to the dimensions and directionality characteristics of the loudspeaker, so that the direct path of the radially propagating sound wave is entirely occluded, whereas a marker in a motion-capture setup can be occluded by another object the size of the marker. For the application by Lopes et al. (2006), which tolerates location errors of up to 70 cm, LOS effects are described as negligible. In our own experience, these errors were only a minor issue, albeit not entirely negligible. That is, for autopannning with ALPS, a smaller error margin of up to 30 cm is tolerable. Seob Lee and Yeo (2011) even successfully hid their microphone behind a curtain.

Literature Review and Previous Work

Some similar applications use ubiquitous technology: Janson, Schindelhauer, and Wendeberg (2010) present an iPhone application using ambient sound signals for tracking and synchronizing phones via Wi-Fi; Filonenko, Cullen, and Carswell (2010) investigate ultrasonic positioning on mobile phones in general; and Mandal et al. (2005) write about the possibilities of using audible frequency signals for tracking. All three methods indicate that an implementation using standard loudspeakers could be achieved, but it is difficult to infer if they could meet all the requirements listed by Schlienger (2016c), summarized in Table 1. In these applications, Janson and coworkers demonstrated the positioning of distinct sound events, taking a measurement of a moving person every 10 to 20 meters, whereas Filonenko and colleagues work with updates at 1 Hz. The use of audible sound by Mandal and associates is interesting: It would be convenient to use content audio directly as a measurement signal. For reasons explained below, a distinct, inaudible measurement signal was deemed more appropriate for ALPS. The Active Bat (Harter et al. 2002) could provide the required functionality, but as it uses purpose-built ultrasound transmitters, it does not comply with our decision to use ubiquitously available technology only.

More recently, Aguilera et al. (2017) presented a system for smartphones as mobile receivers and a minimal number of custom-built senders, for use in small rooms (3 × 3 m), applying a broadband signal in the audible frequency range. They do not provide any indications regarding latency, but they achieved update rates up to 2 Hz, with accuracy of around 10 cm.

Acoustic localization has only rarely been implemented in spatially interactive sonic arts, despite the documented feasibility. A notable exception is the Sonicstrument by Seob Lee and Yeo (2011), using analysis of Doppler shifts to track a pair of off-the-shelf earbuds with one microphone. One bud controls the pitch and the other controls the amplitude of a synthesized sound, mapped to a performer's gestures. They further describe a larger-scale, interactive dance performance, covering an area of approximately 10 × 10 m. They chose Doppler over time-of-flight measurements, as the latter suffer from "limited precision due to the irregular time delay of the system process" (Seob Lee and Yeo 2011, p. 25). By applying an astonishingly simple technique patented by Medvedev, Sorokin, and Khashchanskiy (1989), however, ALPS avoids these issues (see the Signal Processing in ALPS section for details of the technique). Accordingly, the Doppler effect was not explored further, although it would have been an equivalent approach.

Latency is a systemic issue in all time-difference-of-arrival (TDOA) approaches to tracking, as time elapses during the measurement. In AL, which measures comparatively slow signals, it has considerable impact. Applications for which low latency is essential, such as gestural control of musical instruments, tend to map smaller distances than do

larger-scale applications like the 2010 Bristol project. Thus, the scalability of AL facilitates this to some extent, as smaller distances need less time to be measured.

Latency issues, other than the ones unavoidably induced by measurement, need to be dealt with separately, however. They concern the processing time required after the measurement, as well as time used by other processes on the same central processing unit (CPU) that might be given priority over audio tasks in a multipurpose processor like a laptop or desktop computer. Jack, Stockman, and McPherson (2016) provide a concise summary of what latency is acceptable for gestural control of musical instruments. They suggest that if jitter (the change in latency) is small, gestural control at 30–50 msec latency is possible, as musicians can anticipate the delay as long as it stays constant, although also mention studies that set 10 msec as an upper tolerance. Note, however, that they concluded that the majority of commonly used platforms for electronic music are worse than this benchmark. As to the amount of latency that can be experienced by musicians in gestural response, they state that even 4 msec may be noticeable, particularly as jitter. This is consistent with our experience.

Previous Work as Part of the Present ALPS Project

We demonstrated the viability of AL for positioning or tracking in artistic uses at the Klingt gut! Symposium on Sound in Hamburg in 2016: "Leluhelikvartetti" (Finnish for toy helicopter quartet), a homage to Karlheinz Stockhausen's Helicopter String Quartet, uses toy drones that spatialize the sound of the Free Improvisation String Quartet, using an algorithm based on the proof of concept (PoC) described by Schlienger (2016a). A short clip of the performance is available on video from https://doi.org/10.5281/zenodo.5608818. The performance was given an award for Excellence in Art, Design, and Production of Sound by the AES Student Section Hamburg. The AL tracking system used audible noise (around 15 kHz) for tracking, which was masked by the high-pitched whirring noise of the toy drones' four sets of propellers. Responsiveness was adequate for this performance, as the drones moved at moderate speeds. Although the principle remained roughly the same, the current implementation uses inaudible noise for tracking and is more responsive with considerably lower latency.

Earlier work explored how tracking technology could be applied to overcome limitations experienced with audio technology in practical application scenarios. Namely, the disconnection experienced by musicians when using electronic and electric instruments without built-in acoustic sound sources, and the gestural limitations of many electronic instruments are discussed in earlier publications (Schlienger 2016a,c), based on field notes from a free-improvisation workshop. (Other authors also describe the gestural limitations inherent to electronic instruments, e.g., Dean and Paine 2012; Mitchell and Heap 2011; Robinson et al. 2015; Salazar and Armitage 2018.)

The first author conceptualized the notion of the *kinaesthetic interface* in order to generalize the requirements for interfaces for spatially interactive sonic arts:

> It records kinetic events at the right resolution, over the necessary distances, at sufficient speeds, and with the necessary accuracy to make them relevant enumerations and encodings as parameters correlated to kinaesthetic experiences (Schlienger 2016c, p. 6).

Besides this qualitative notion, quantitative requirements summarized in Table 1 were identified with the help of an online survey, asking professionals in the field about their expectations of tracking systems. These quantitative requirements also provide the benchmark for the current implementation, as discussed below.

Schlienger (2016a) provided a PoC for an implementation for AL as a tracking device for automated panning (autopan) at the 2016 NIME conference, where it was successfully demonstrated. The demonstration also determined various application scenarios in which tracking technologies could help to create panning trajectories for moving sound sources in sonic arts. In the paper, four conceptual

possibilities are identified. The following is an updated summary of the scenarios.

Two Spaces: Stage and Auditorium

This is the typical "concert hall" situation: Moving sound sources (i.e., musicians with instruments or singers) are in a different space from the audience. This other space can be a stage, in which case the automated panning reproduces (mirrors) the positions of the sound sources for the listeners in the auditorium. Even in the case of a classical music concerts, in which musicians remain seated throughout, it makes sense to replicate their spatial positions. In a broadcast situation, the listening space is even further removed from the musicians, but the logic remains the same. Here, and particularly in case of opera, musical theater, and other more spatially dynamic musical practices, automated panning can simplify the task of the sound engineer who otherwise would have to pan these sound sources manually. Conceivably, the scenario is the same if the musicians are in a studio. The difference in this case is that the position of a musician in the room does not need to be replicated, in fact it can be, and usually is, arbitrarily set without causing conflicting spatial impressions in the audience, who have no visual cue to the studio's setup. (So, in contrast to broadcasting and amplified live performance, automated panning would only have limited use in a typical recording studio situation.)

One Space: The Commons

A moving sound source is reproduced and amplified in the same space as the audience. For example, assuming a multiple-loudspeaker or surround-sound setup, and a very quiet, acoustic sound source with a closely placed microphone, which produces a signal that is simultaneously played back on the loudspeaker closest to it, or panned to its phantom position between multiple loudspeakers. Here "very quiet" can also apply to a laptop or another electronic or electric instrument. By locally amplifying it, the quiet instrument becomes the acoustic equal of a loud instrument, which probably does not need amplification and inhabits a distinct, localized space, with its sound radiating from it. Until now, the quiet instrument would have been amplified at a fixed pan position on one or several loudspeakers, possibly misrepresenting the instrument's actual position, particularly when the sound source is moving. Here, automated panning vastly increases the spatially interactive possibilities for a range of musical practices—even enabling new ones, in the fields of participant performances, performance art, festival commons, art games, and many more.

Virtual Sound Source

This is a typical example of a spatially interactive installation: A panning trajectory is created in real time (e.g., by visitors issued with a tracking device), so that sound follows their movements. The audio content being produced elsewhere or was prerecorded. A further possibility would be that a participant triggers certain events at certain positions. If the space is large enough for audio content to be heard selectively, depending on the participant's location, a panning trajectory can create narrative meaning: Participants will hear different sequences of events depending on their relative positions to each other. Last but not least, this is also a way to map offline content in real time to an online trajectory in space.

Trajectory and Sound Event Temporally Separated

Here the panning trajectory is produced in advance, and later used as a map for a musical event, which might also have been produced in advance. Although the trajectories could be generated by other means, tracking technology could provide an easy and organic way to achieve this.

The four application scenarios defined above are for spatial reproduction of moving sound sources in spaces from approximately 9 to 144 m^2. In earlier work it was shown how the same approach can also be scaled to smaller spaces, resulting in higher update rates, allowing gestural control of musical parameters that depend on quasi-real-time interaction (Schlienger 2016b). A Theremin-like pitch-control software device was demonstrated,

along with a rudimentary percussive instrument—rudimentary being the operative word here. Yet theoretical possibilities are evident and the current research paves some inroads towards improving low-latency gestural control with AL.

System Overview

This section first introduces the ALPS algorithm and its implementations in the ALPS software Audio1 and al-Qt, written in C++ by the second author and available at https://doi.org/10.5281/zenodo.5602869. To achieve audio panning—the adjustment of relative amplitudes of audio signals on loudspeakers to achieve the impression of a phantom sound source lying somewhere between the loudspeakers—some information as to the position of that virtual sound source is required. Such positions are provided in ALPS by using AL techniques. Evidently, to use ALPS for audio panning makes sense as both processes, positioning and tracking with AL and audio panning, require an infrastructure composed of loudspeakers. But ALPS could be adapted to provide positioning data for a variety of uses other than panning, wherever loudspeakers are present as part of a setup (multimedia, virtual reality, auditoriums, surround sound, museums, etc.), for instance, to trigger items in an audioguide relating to museum exhibits, or similar.

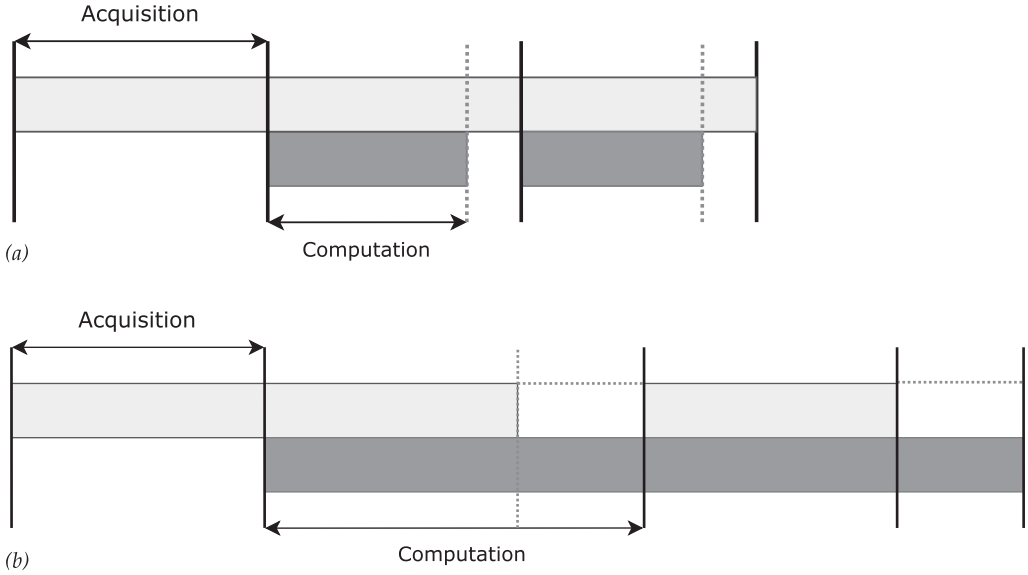
We then introduce the Autopan Max patch (https://doi.org/10.5281/zenodo.5607121), which utilizes ALPS to automatically pan content audio to the position of a person carrying a receiver (microphone) while moving in a room equipped with multiple loudspeakers. As such, it is a specific application of ALPS, which solves the seemingly technically orthogonal problems of *panning* and *positioning* by aligning them through the use of distance-based amplitude panning (DBAP, cf. Lossius, Baltazar, and de la Hogue 2009).

The ALPS Algorithm

To estimate positions of acoustic signal receivers (microphones) and senders (loudspeakers), ALPS uses AL. Acoustic localization techniques are applied in sonar, fish finders, and parking aids in cars, but also for medical purposes (e.g., sonography). Typically these technologies use dedicated hardware and ultrasonic signals (sound in the frequency range above human hearing). As it is known that sound moves typically at 343 m/sec, the distance between a microphone and a loudspeaker can be calculated by measuring the time delay of a known audio signal at the microphone in comparison to its original on the loudspeaker: $d = c \times t$ where d is the distance a sound wave travels, c is the speed of the sound wave, and t is time. Using several loudspeakers, or several microphones, the position of a sound source can be trilaterated, using the TDOA method. Trilateration calculates positions from distances, describing the points of intersections of spheres. To estimate the position of a moving object in 3-D space with trilateration, four synchronous distance measurements from known points are necessary. If the position to be tracked can be assumed to be on a plane—the earth's surface in GPS, for example—some simplifications are possible. Triangulation, in contrast, estimates a position from known angles between objects, without any knowledge of their relative distances and hence intrinsically inapplicable to TDOA methods.

The TDOA methods trilateration and multilateration are not discussed here in greater detail, as the application of ALPS discussed in this article, namely, autopanning, does not require them: The relation between distance measurements obtained through AL and the panning laws applied in DBAP means that trilateration is not necessary—this is perhaps surprising. The distance measurements, 1-D positioning so to speak, suffice. Two-dimensional estimates of relative positions were only necessary to validate the accuracy of the ALPS software in the experimental setups B1 and B2 discussed below: Positions were defined through Cartesian coordinates on a single plane to estimate the position of a device moving along a known trajectory. This was done with the help of the Pythagorean theorem implemented in the ALPS error calculator MATLAB script (https://doi.org/10.5281/zenodo.5607528).

Figure 1. Acquisition (light gray) and computation (dark gray). When computation is faster than acquisition, computation waits for the end of acquisition, thus the process happens in quasi–real time (a). On the other hand, when computation lasts longer than acquisition, acquisition waits for completion of computation, which results in additional latency but guarantees that computation always refers to the most recent acquisition (b).



Signal Processing in ALPS

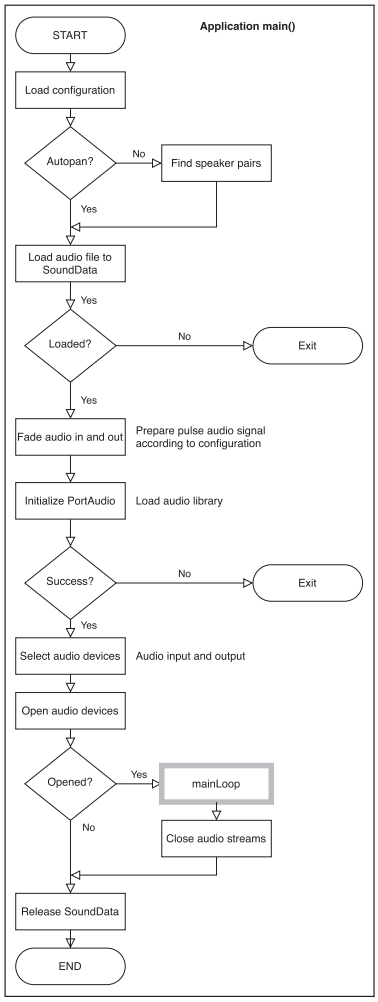
Using the principle of AL, a measurement signal (e.g., band-limited white noise) is played on a loudspeaker and compared to an audio recording of it, a *capture block*, made on a microphone at a distance d. A sampling frequency and a window length in samples are chosen. The larger the window, the larger the area that can be covered with the system, but the longer it will take to obtain a measurement. Each sample has an index number, sequentially from first to last sample of a window. The next step is to take one window's length of the measurement signal and calculate its correlation with the signal of the same length that was recorded on the microphone at the same time. In the resulting correlation signal, the sample with the highest amplitude is found at the index number corresponding to the delay between the measurement signal and the signal recorded on the microphone. According to the relation $d = c \times t$, the distance can be estimated between the microphone and the loudspeaker.

The data acquisition records the signal for the length of time of the window, while it computes the previous window's correlation signal. If this computation lasts longer than the acquisition of the following length of audio, the system delays the acquisition of the next length until computation is completed (see Figure 1). If the computation takes less time than the acquisition time, the system runs in quasi–real time. As ALPS uses pulsed signals stepping through the loudspeakers one by one, the effective latency of the system is given by the length of the pulse multiplied by the number of loudspeakers, even when processing is achieved in quasi–real time. This is discussed in greater detail in the following section, The Measurement Signal.

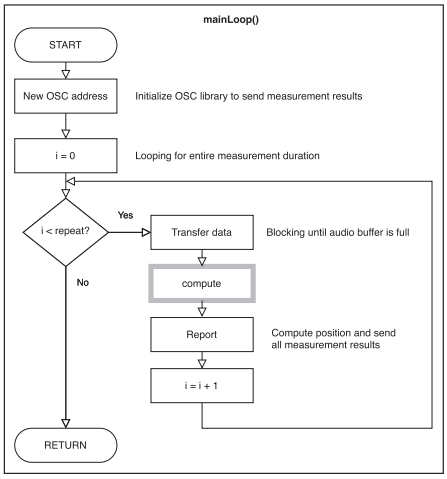
Additional latency needs to be accounted for, introduced by concurrent processes on general-purpose computers, which tend to vary over time and make measurements unreliable (Lopes et al. 2006; Seob Lee and Yeo 2011). For ALPS, the following simple solution was applied (cf. Medvedev, Sorokin, and Khashchanskiy 1989): By physically connecting one output of the sound card to a reference input on the same card, round-trip latency (RTL) is measured for every window, covering all delays due to analog-to-digital conversion, operating system, and concurrent processes on the computer. In every window, ALPS deducts RTL from the length of the acquisition signal, so that the remaining length consistently corresponds to the time it took the sound to travel from the loudspeaker to the microphone.

For a structural schema of the ALPS code, see Figure 2.

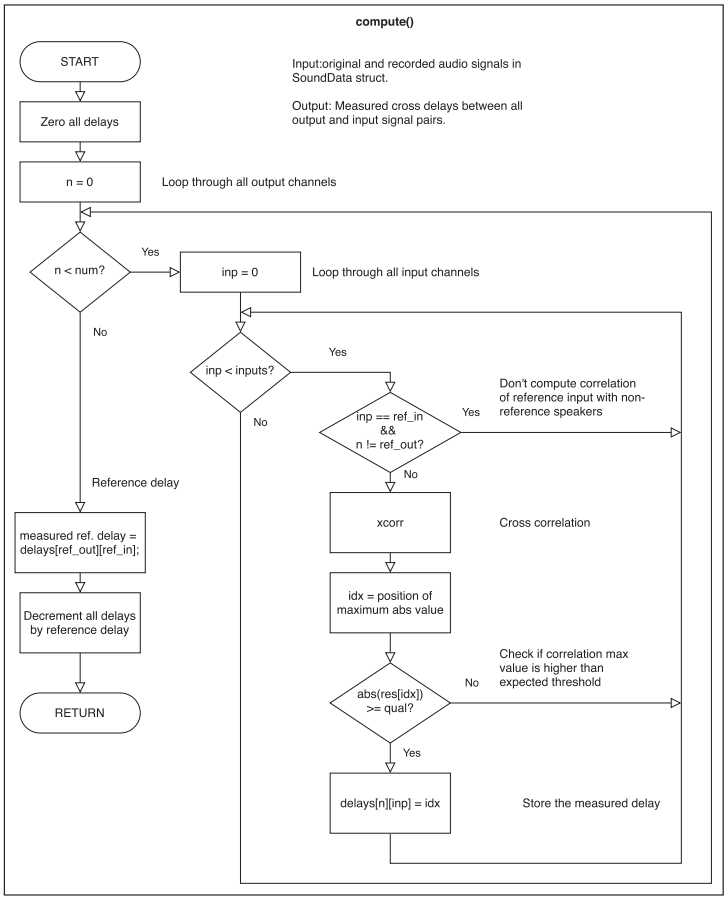
Figure 2. Block diagram of the code used in the Acoustic Localization Positioning System (ALPS). The function main() sets hardware according to user defined configurations (a). It uses the function mainLoop(), which sends values via OSC and handles buffer blocking (b). This, in turn, uses the function compute(), which computes correlation and compensates for reference delay (c).



(a)

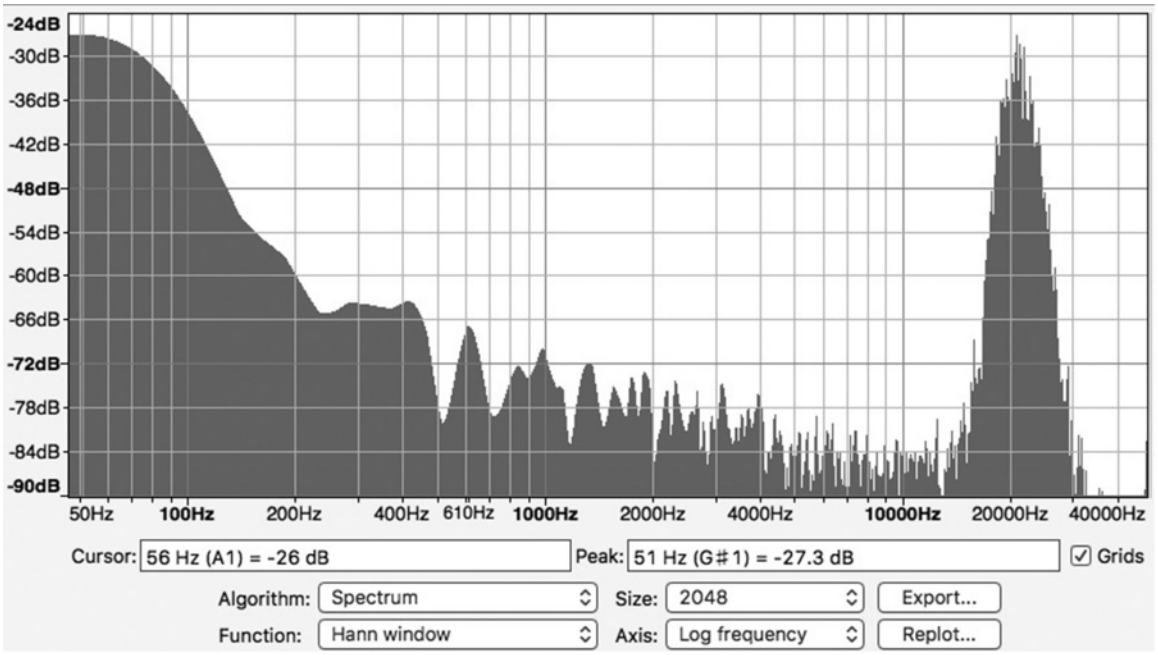


(b)



(c)

Figure 3. Measurement signal played on Genelec 1029A loudspeakers and recorded on DPA 4061 microphones.



The Measurement Signal

Arguably, the content audio itself could be used as a measuring signal. Yet, for the application in question, the position of the tracked device is also needed when there is no content sound present—for example, in musical pauses (silence). For that situation an inaudible signal is required. Thus, a signal is applied that is above the frequency range audible to the human ear, but still within the range of off-the-shelf loudspeakers. A pulsed measurement signal of random noise between 19 and 30 kHz is used in ALPS. If a loudspeaker's specifications state 20 kHz as an upper limit, this commonly refers to the frequency above which the response rolls off. Essentially, the frequencies in the roll off are still there, they are just quieter. Many off-the-shelf loudspeakers reach 30 kHz even before a significant roll off occurs. But we recommend creating a test signal for each hardware setup. In Figure 3, for example, one can see that there is indeed sufficient detectable signal for Genelec 1029A loudspeakers, recorded on a DPA 4061 microphone. If a loudspeaker's frequency range does not extend to 30 kHz, or the roll-off above the nominal upper limit is too steep, a narrower band could be chosen, fort example, 19 to 21 kHz, or a lower band, for example, 16 to 19 kHz. This might result in audible noise for some listeners, particularly younger ones. That the signal at higher frequencies has less energy does not constitute a problem, above the audible frequency range its power can be increased at will as long as the amplification does not cause distortion. For the experiments here, a white-noise signal was generated, with high- and low-pass filtering applied repeatedly and normalization in between. The resulting file was saved as a single multichannel audio file. The number of channels has to correspond to the number of loudspeakers in the system, and the file needs to be of sufficient length to cover at least a complete cycle across them. The pulse lengths and cycle duration are set in a configuration file.

In most cases, the same loudspeaker can be used both for localization and for content audio, as the measurement signal lies distinctly above the frequency band of content audio. To improve the signal-to-noise ratio, the content signal can also be

"cleansed" in the band in question by using low-pass filters at a corresponding cutoff, for instance, 19 kHz. For the PoC shown by Schlienger (2016a), the measurement signals were sent to all loudspeakers at the same time, so that the correlation could be measured from a single capture block. This resulted in an abysmal signal-to-noise ratio: For every calculation, all other signals constituted noise. By pulsing the signal, that is, by taking turns for each loudspeaker, this is avoided here. The reversed approach is less problematic: When calculating the distance between multiple microphones and a single loudspeaker, only one measurement signal is necessary—in the same way that in everyday verbal conversations additional listeners do not make noise, only additional speakers do. For many applications this single-loudspeaker approach is therefore preferable. Unfortunately, the primary design here is for a multiple-loudspeaker situation.

Although Lopes et al. (2006) show that LOS issues are negligible, the effect of sound diffracting around objects is limited: The wavelengths at frequencies above 20 kHz will be shorter than 17 mm, and consequently reflect from larger objects. Still, assuming plane-wave propagation, the occluded wave is still available next to the obstacle: It will still be detected by an omnidirectional microphone, only delayed slightly with respect to the point source assumed for distance estimation. This additional error causes an inaccuracy in measurement, not a loss of signal. But this only applies to situations in which a plane-wave model is applicable. Zhang et al. (2017) presume the complete loss of LOS with every introduced obstacle. This might be sensible for their scenario, where the sound sources are small loudspeakers in mobile phones, because even relatively small obstacles create a near-field situation.

To play and record a random noise signal between 20 and 30 kHz, a sampling rate of at least twice the highest required frequency needs to be applied (Nyquist rate must be twice the highest required frequency to satisfy the Nyquist sampling criterion). So for a 30-kHz signal, the 41.1- or 48-kHz sampling rates commonly used in audio applications will not be sufficient. Owing to hardware restrictions in the initial phase of the project, all tests were run using a sampling rate of 96 kHz. In hindsight, 88.2 kHz could have improved processor performance and thus reduced computing time, at a minor loss in precision and update rate.

The ALPS Software: Audio1 and al-Qt

We implemented ALPS as a stand-alone application in C++ based on the PoC presented in an earlier publication (Schlienger 2016a). It was developed in cooperation with the participants of the workshop on Music, Space, and Interaction (MSI), and coded for this project in collaboration with the authors. There are two versions available. Audio1 forms the basis of most of the discussion here, unless indicated otherwise. It was tested under macOS 10.12 on a single-processor 2011 MacBook Air. For greater detail than what was shown in Figure 2, the reader is encouraged to examine the source code, available as a GitHub repository (https://github.com/spatmus/alps). The second version, al-Qt, which relies on a multiprocessor architecture, was tested for macOS 11.6 and provides the necessary update rates and low latency required for quasi-real-time applications. A precompiled binary of al-Qt tested for macOS 10.12 and macOS 11.6 is available from https://github.com/spatmus/alps/releases.

In ALPS, we use ubiquitous technology in the form of commercially available audio loudspeakers and audio microphones designed for frequencies within the human hearing range, that is, for speech and musical content. The processing power of a fairly recent laptop or desktop computer is adequate. This makes ALPS a straightforward choice for tracking or positioning in situations where loudspeakers and microphones are readily available, for example, surround-sound systems, virtual-reality applications, conferencing, live sound, and home theater.

Settings and Configurations

The settings for both instantiations of the ALPS software can be controlled via a configuration file (see config.ini in the GitHub repository for a generic example). In the al-Qt version, the configuration file can also be accessed via the dialog panel. The

choice of adjustable parameters is intended to help in debugging and experimentation for application-specific performance. As a stand-alone application for 3-D tracking, ALPS uses trigonometrical Euclidean distance calculations for possible pairs of delayed signals. The positions of all loudspeakers need to be given as $x/y/z$ Cartesian coordinates in the positive quadrant. The three-dimensional position estimates are, however, not essential for use of ALPS in conjunction with the autopanning provided by the Autopan Max patch; the distance readings alone are sufficient.

To run the ALPS software with the Autopan Max patch, autopan needs to be enabled. This sends the distance readings directly via the User Datagram Protocol (UDP) and Open Sound Control (OSC) to a network port in Max, whose IP address and port number are also set here. In Max, the messages from ALPS are received via a `udpreceive` object. ALPS and Max can run on the same processor or over a network using OSC. Further settings are for the time offsets as well as for lengths and fade times of the pulses. The overall duration of the sequence can be set to compensate for the total latency of all processes involved and how many times the sequence will be repeated. In debug mode, the last repeat of the pulses, as well as the last instances of the signals on the selected inputs, are recorded as an audio file in the subdirectory set here. Generally, the challenge is to find the right settings for a balance between update rate, area covered, and latency for a particular situation: If the tracked device is ten meters away, around 0.03 seconds are needed before computation, but if it is only one meter, 0.003 seconds meet the requirements. The setting for maximum distance can limit erroneous outliers by setting it to a value smaller than the longest possible distance in the room. But it can also be set to a value arrived at by trial and error to mark the distance above which readings are no longer reliable, even if these values lie well within the room dimensions, for example, if the room is large. The quality setting allows one to filter readings that have a small amplitude and are hence more likely to be wrong. This effectively deals with early reflections, whose amplitudes are unlikely to be equal to or higher than the direct signal.

Autopan Max Patch

The Autopan Max patch utilizes the ALPS software to automatically produce the panning of a sound source in a multiple-loudspeaker setup by tracking a moving object that represents the position of that sound source. This has the advantage that virtual sound sources can be turned into spatially discrete sound objects that can change position, just as most acoustic instruments can. Examples of virtual sound sources are mobile devices, such as laptops and tablet computers, as well as electric and electronic instruments, but also many novel musical interfaces. Autopanning distinctly improves immersion in spatially interactive sonic arts, especially when virtual sound sources are played live along with acoustic instruments, as will be discussed in the Tests in Situated Use: Qualitative Evaluation section.

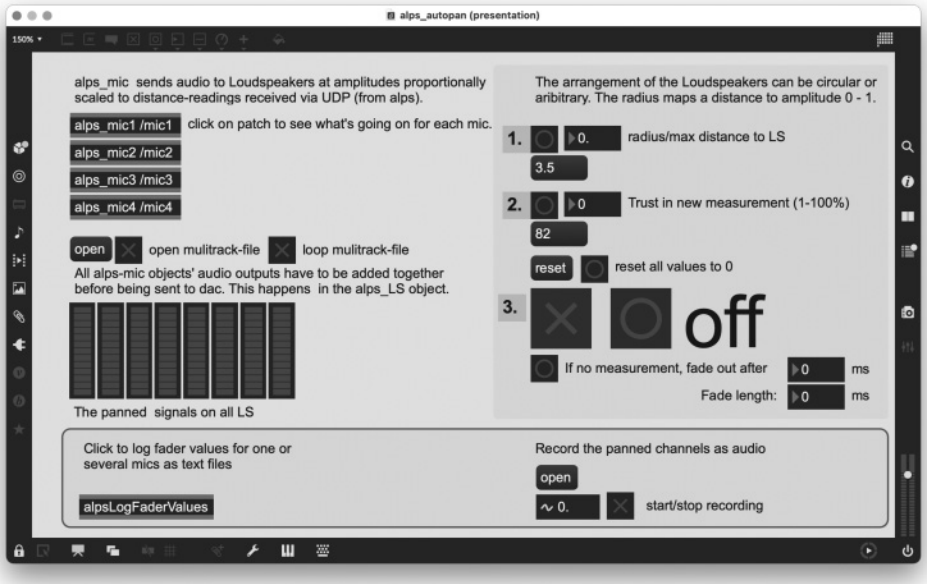
The Autopan Max patch and associated subpatches (tested for Max 7 and Max 8), as well as a standalone Max application that should work "out of the box" on all platforms, are available from https://doi.org/10.5281/zenodo.5607121.

The patch's main interface (see Figure 4) allows one to record the panned audio files and to save a log of the calculated trajectory in the form of error-corrected distances. A separate alps-player patch can recreate the distance readings. An additional alps-recorder patch is provided to write the raw data as received on the UDP port to a log file.

The smoother subpatch provides rudimentary filtering and error handling. When a distance reading fails to be updated from ALPS, the content audio starts to fade out on the loudspeaker in question (controlled via fade time and fade length). Once a new reading is available, the audio fades in again. This reduces responsiveness slightly but provides a high-pass filter for outliers during periods between signal updates. The filter is adjustable, in degrees of confidence in newly acquired values, by adding a user-defined percentage of the old measurement to the newest one.

The alps-mic subpatches (see Figure 5) show the smoothed levels of the distance readings in meters, measured for each loudspeaker. Further, the panned

Figure 4. The main window of the Autopan Max patch has three sections: The panel on the left provides access to the subpatchers of all tracked devices (microphones); selection of content audio in multitrack format and visual monitoring of levels on all loudspeakers. In the bottom panel, fader values can be logged as text files and the panned content audio recorded as a mutlitrack file. The panel on the right provides 1. Controls for loudspeaker layout, 2. Filter adjustment, and 3. A panic button (ON/OFF) and fade-out times for content audio in the absence of measurements.



content-audio source can be set here. The options are

1. on/off;
2. player, for a mono audio file which can be chosen via the open button;
3. audio in, to connect to the input of a connected sound card; and
4. mutrach, to choose one channel of a multitrack file selected from the autopan patch main window.

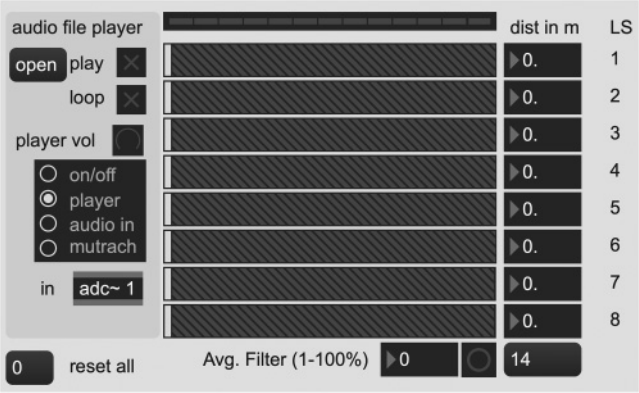
The reference microphone can also be monitored here (according to configuration setting for refIn in the ALPS software) and then, for example, routed to compensate for jitter.

Distance-Based Amplitude Panning

In an ideal world, every sound source to be reproduced could be represented by a real sound source in its place. In a multiple-loudspeaker setup this can be approximated by representing it by the nearest loudspeaker. If the ideal position lies between loudspeakers, it can be approximated on multiple loudspeakers according to amplitude panning principles. This is a simplified description of the DBAP algorithm by Lossius, Baltazar, and de la Hogue (2009). It lends itself to the use with ALPS, because the distance measurements are inversely proportional to the required amplitudes. In that sense, panning succeeds even if only one distance measurement is available, as usually happens when the distance between the tracked device and a loudspeaker is small. In that case, a single-loudspeaker representation is adequate. Presuming that rolloff $R = 6$ dB (in decibels per doubling of distance) equals the inverse-distance law for sound propagating in a free field, a direct mapping of distance measurements estimated by ALPS with the amplitude of the panned audio signal becomes possible.

The ALPS software can also calculate absolute positions in a Cartesian system. Although this is the sensible thing to do for most general positioning and tracking tasks, it is not necessary for audio panning with DBAP, as outlined in the The ALPS Algorithm section. This is in contrast to angle-based panning paradigms, such as vector-base amplitude panning (VBAP, cf. Pulkki 1997) or Ambisonics (Malham

Figure 5. The alps-mic subpatch provides visual monitoring of the measurement signals on each loudspeaker as "distance in meters." In the "audio file player" section the content audio to be panned can be selected: mono files, audio input, or the specific multitrack channel to be panned (of the multitrack file selected in the autopan main window); and individual adjustments of the averaging filter.



1998): To arrive at the required panning positions in Cartesian or polar coordinates, three known points are needed for trilateration in 2-D and four points in 3-D. With DBAP, the absence of a fourth, third, or even second measurement makes the situation less than ideal, but not undefined. Consequently, less hardware (i.e., fewer loudspeakers) is necessary when using DBAP. This is why the seemingly orthogonal problems of acoustic panning and acoustic localization align well here and inform each other in parallel.

Methods

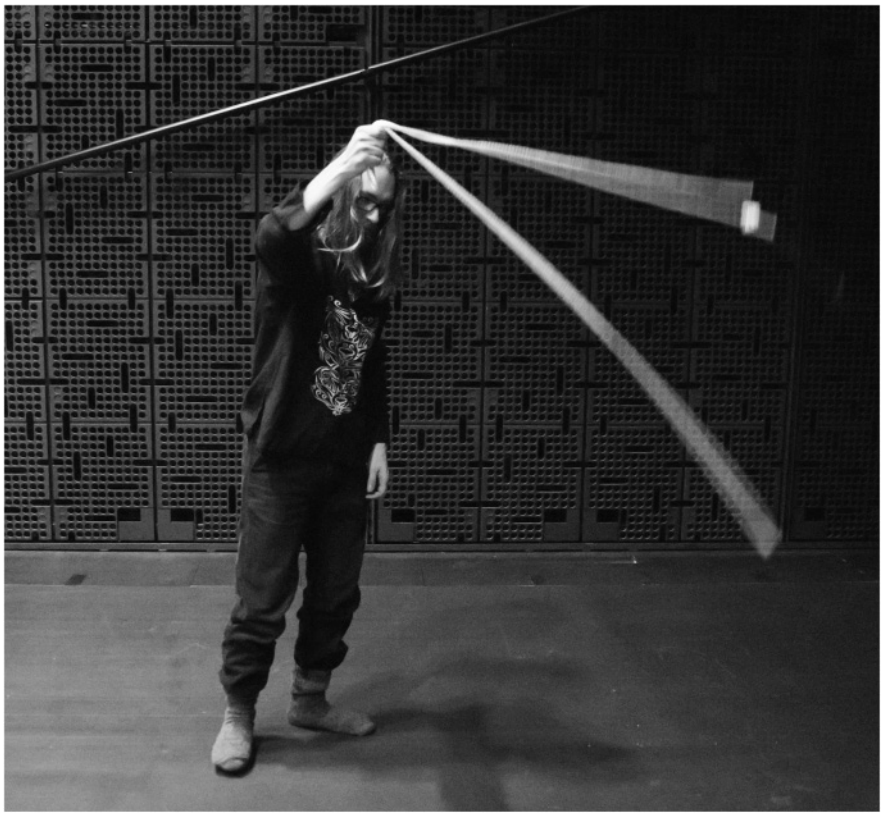
The development method of the ALPS project is an extension of participatory design principles though the addition of interdisciplinary improvisation (setup A). Besides this artistic-research approach, quantitative methods are also applied (setups B1 and B2), to gauge the extent to which the technology meets the requirements defined and evaluated in its artistic use.

Free Improvisation, Artistic and Qualitative Research

Interdisciplinary improvisation is an experimental practice bringing together practitioners of various disciplines to seek common ground, reduce significant differences, and identify challenges (Andean 2014). The Research Group on Interdisciplinary Improvisation was launched in 2012 at the University of the Arts Helsinki. In collaboration with some of its members, the MSI workshop was brought to life in 2013, with the idea of applying the practice of interdisciplinary improvisation to technology development, specifically in the field of spatially interactive sonic arts (Schlienger and Olarte 2016). It combines concepts of interdisciplinary improvisation with participatory design, whereby all participants in a development project are involved at all levels (Simonsen and Robertson 2013). Interdisciplinary improvisation, as applied in MSI, helps to find unexpected, simple, and sustainable solutions by prototyping a situation in a problem area, rather than finding the solution to a problem. To use a simple example, the situation "everything you find in the room is an instrument" is given as a score for the improvisation. This means participants move about to explore the room and its objects, inventing "instruments" through improvisation. This method could be applied to other fields. For example, as a participating town planner suggested, imagine a new housing development, where footpaths between the buildings need to be planned. One approach would be to do nothing initially but survey what paths the inhabitants choose if left to roam free. If this is not possible, the situation could be enacted as improvised theater, as in "let's pretend here is the bus stop, here the shopping center, and here the motorway . . ." The difference is that in art, which seems to be already situated on some metalevel with respect to reality, the element of role play is not necessary. There is no point of "let's pretend to play the violin" if one has a violin. Artistic situations intrinsically constitute this "let's pretend" level, making improvisation as a development method for technology in the arts even more felicitous. Figures 6 and 7 show still images of improvisations in MSI.

This approach represents a counter model, an antidote to conventional "consumer evaluations," in which a handful of test subjects test a product for 20 minutes, after which they are asked to "tick boxes" to select choices from answers to leading questions. Rather, the idea of MSI was to form new ideas for development over time,

Figure 6. Participant exploring the space–sound relations of a found instrument in an improvisation in the workshop on Music, Space, and Interaction (MSI). Photo by author.



developing practices and techniques along with the technology. Typically, participants were musicians and composers, scenographers, multimedia artists, sound designers, dancers, painters, town planners, choreographers, and so on. The workshop was open to students and professionals, the latter making up around 30 percent of the participants. The workshop underpinned the ALPS project from the beginning, as detailed in earlier publications (Schlienger 2016a,b,c).

Experimental Setups

Experiments were conducted in two settings: In setup A, a medium-size performance area was used for qualitative experiments oriented towards artistic research, whereas experiments in setup B were of a quantitative nature. Setup B was further subdivided into two variants, B1 and B2, to compare between the single-processor implementation Audio1 in setup B1 and the multiprocessor implementation al-Qt in setup B2.

Setup A: Stage-Size Performance Area

This is the setup for experiments with the ALPS software and the Autopan Max patch in the MSI workshop. Audio1 and the Max patches were run on two midrange MacBooks communicating over WLAN, one MacBook from 2011 and the other from 2012. For tracking we used omnidirectional AKG CK 55L microphones, which meet the requirements for AL in the frequency bandwidths above 20 kHz, despite the moderate price in comparison to the DPA 4061 microphones we used to record the audio content. Both types are small, unobtrusive lavalier microphones. The sound card, a MOTU

Figure 7. A typical scene in an Interdisciplinary Improvisation session in the MSI workshop. Photo by author.

16A AVB, was chosen for its extremely low system latency (less than 2 msec) and flexible input/output options. The AKG microphones were connected to wireless senders and receivers that work in the analog radio-frequency band, with typical latencies in the microsecond range. For loudspeakers we used Genelec 1029As, which deliver sufficient volumes above 20 kHz, as was shown in Figure 3.

Setup A was located in a performance space approximately 10 × 16 × 3 m in size, treated with acoustic panels for classical music. First and early reflections were minimized, but the room was by no means dry. Four loudspeakers were spaced evenly around the center of the room at a radius of 3 m, on stands approximately 1.1 m high, making elevation negligible in relation to most acoustic instruments' positions in the room. Four to seven performers moved around freely with small acoustic sound sources, microphones, or electronic musical instruments lacking acoustic output. The electronic instruments' sounds were distributed via a second set of four loudspeakers positioned on top of the set of four dedicated to the tracking task. (For a schematic layout of the room, see Figure 8.)

The sessions in MSI usually took the form of exercises in free improvisation that were 10 to 40 minutes long, following a very loose score open to the participants' interpretation. For example, a group exercise might have the instructions "start in the middle of the room, spread out, but get quieter while you do so and listen to the others." After each exercise, the experience was discussed in the group. Based on these discussions, the techniques or technologies applied in the exercise were modified as needed, either immediately (adjusting playing techniques, "tweaking"), or between workshops (rebuilding instruments, recoding, developing).

For documentation, the audio content panned by the ALPS Autopan Max patch was recorded as an audio file. A selection of four-channel recordings can be downloaded from https://doi.org/10.5281/zenodo.5607027.

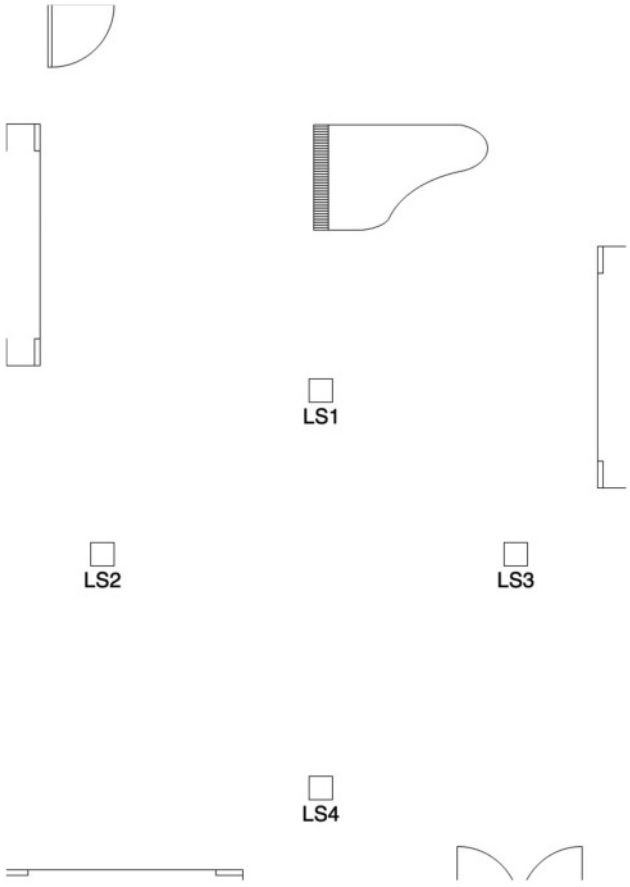


Figure 8. The room layout for setup A, a performance space of approximately 10 × 16 × 4 m, where we conducted the qualitative experiments. LS1–LS4 mark both the loudspeaker positions for measurements and those for content audio.

Setup B1: Performance Area Section

In setup B1, tests were run for situations having four microphones and four loudspeakers as well as having four microphones and eight loudspeakers, by running all equivalent computational processes but only actually distributing the measurement signals via three loudspeakers, and recording them on one microphone only. This way, a section of a larger performance area could be observed in detail, without having to completely set it up. The room is a living room, approximately 6 × 3 m in size, with laminate flooring, concrete walls, paneled ceiling, and some soft furnishings. The the test area covered 2 × 3 m of this room. The loudspeakers were placed on the floor, simplifying the setup by taking elevation out of the equation. A 0.5-m grid was marked on the floor and a video camera installed on the ceiling, which was approximately 2.5 m high. A setup having eight loudspeakers with the same spacing could cover an area of 6 × 3 m or 2 × 9 m. These were smaller distances than in setup A, which covered an area of 6 × 6 m with only four loudspeakers and accordingly increased the quality of the measurements. Further, the movement of the device to be tracked was automated by attaching it to a mechanical motorized vehicle. An electric toy locomotive (see Figure 9) was used for this, fitted with an AKG CK 55 L omnidirectional microphone. The track followed the line $y = 1$, with LS1 at $x = 0, y = 2$; LS2 at $x = 2, y = 3$; and LS3 at $x = 3, y = 0$. Elevation was negligible, as the loudspeakers were on the same level as the tracked device.

The distance measurements were recorded at a known rate in a log file for further analysis in MATLAB. Accuracy can be evaluated by comparing these measurements with predicted values based on the assumption that the electric locomotive runs at a constant speed on a straight track. An animation of the movement based on the measured data, in which the distance measurements are expressed as radii of circles with origin at the center of each loudspeaker, can be directly compared to a video recording of the experiment. Videos, data files, and the MATLAB scripts are available from https://doi.org/10.5281/zenodo.5607528. Figure 10 shows the merged stills of the animation and the video at 00h:00m:31s.

Setup B2: Performance Area Section

In a setup nearly identical to B1, B2 demonstrated the advantages of the multiprocessor architecture of the current implementation al-Qt over the single processor version Audio1 by testing it on a faster moving object (see Figure 11). A toy car and track of type Hot Wheels was fitted with an Audio Technica ATM350 cardioid microphone (see Figure 12). The car was accelerated by hand, so it decelerated during its journey. The experiments were recorded on video, available with the corresponding data from https://zenodo.org/record/5604446. The difference between setups B1 and B2 is that four physical

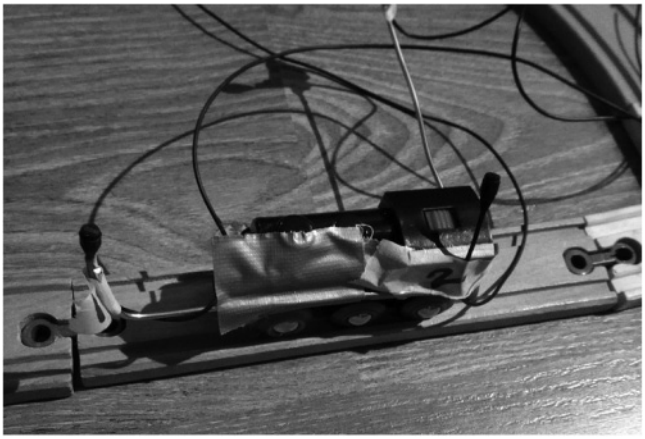


Figure 9. The "device of constant speed" in the picture, had three omnidirectional microphones attached, which in principle would also allow for 3-D tracking on one loudspeaker (multiple-microphone approach). For the experiments in setup B1, only recordings of one microphone were necessary (single-microphone, multiple-speaker approach). Photo by author.

loudspeakers were set up, rather than three as in B1. Furthermore, B2 is in a slightly less reverberant room with a slanted wood-panel ceiling and wooden floor boards, resulting in fewer early reflections.

In this setup, al-Qt ran on a MacBookAir (M1 2020), under macOS 11.5.2.

Results and Discussion

We now discuss the results of the experiments. First we analyze quantitative results and compare them to expected values from earlier works for each experimental setup. This is followed by a discussion of implications for low-latency applications. Finally, we look at qualitative results providing an evaluation of the implementation in the context of artistic practice.

Experiments in Controlled Environment: Numerical Evaluation

The data discussed in the following are available from https://zenodo.org/record/5604446. The experiments were recorded on video, available from the same URL. From the experiments conducted in setup B1 using the single-processor Audio1 version of ALPS, two sets of data are discussed, both sets generated by the Brio electric toy locomotive. One data set was recorded while processing audio for four loudspeakers, the other for eight loudspeakers. To ensure that the window length was sufficient for the whole range of loudspeakers, the signals were pulsed as if all loudspeakers were present, for example, LS1 received the first pulse, and LS8 the eighth. In B2, which looks at the improvements in the update rate and the latency of the al-Qt version of ALPS, two data sets are discussed. The first was recorded with four loudspeakers and a single microphone, the second set (marked "rerun" in the data set) with four loudspeakers and four microphones.

Experiments in B1

As the comparison between root mean square error (RMSE), mean average error (MAE), and median absolute deviation (MAD) in Table 2 shows, the system can be considerably improved by eradicating outliers, which are proportionally over-represented in RMSE, but have less weight in MAD. The difference between the eight- and four-loudspeaker sets, particularly the measurements for LS7 in the eight-loudspeaker set, is notable. Admittedly, it highlights a flaw in how the data were recorded: A measurement that was not updated at the end of a measured interval repeats the previous (and hence incorrect) reading until a new one is available. This adds error: In the eight-loudspeaker data set (https://doi.org/10.5281/zenodo.5607528), it can be seen that for nearly a third of the duration, no signal was recorded on LS7, repeatedly registering a distance of 0 m. This misrepresents the system's capability. A caveat here to future adopters of the current implementation: Better results would be achieved by filtering all distances that exceed a maximum distance setting of 2.5 m and by ignoring distances of 0 m distance, treating them as, for instance, the floating-point value NaN (not a number). In fact, looking at sequences of measurements in the data file, it is evident that where a signal does exist, there is little difference in error between the two sets. Consequently, performance could be further improved by extrapolating missing readings from direction and speed. The latency visible in Table 2 affects update rate only; it has no influence on the accuracy of the measurements (42.5 msec for four

Figure 10. Graph superimposed on video still of experimental setup B1, showing the tracked device at $Y = 1\ m$, $X = 2\ m$. *Loudspeaker icons indicate Cartesian positions* (x, y) *on a plane with the tracked device* ($Z = 0$) *of the Genelec 1029A loudspeakers at (0,2) for LS1; (3,2) for LS 3 (LS7 in dataset 2); (3,0) for LS4 (LS8 in dataset 2). In the animation video, the distance reading for LS1 is shown in white; that for LS3 (LS7) is dashed white; and LS4 (LS8) is in black. Photo by author.*

Figure 11. The layout of experimental setup B2 shows a toy car track made up of sections 0.3 m in length in a grid with markers every 0.5 m. Photo by author.

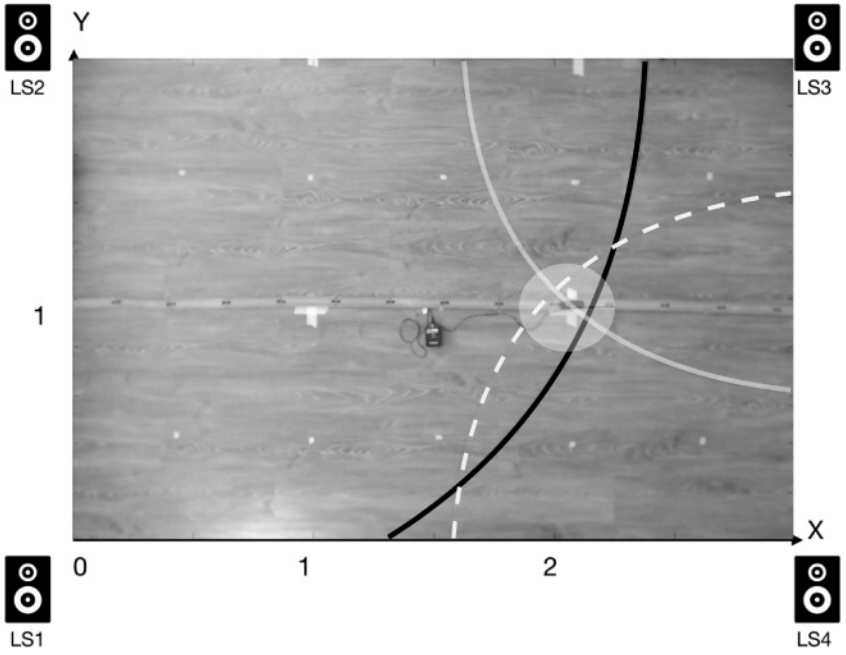


Figure 10.

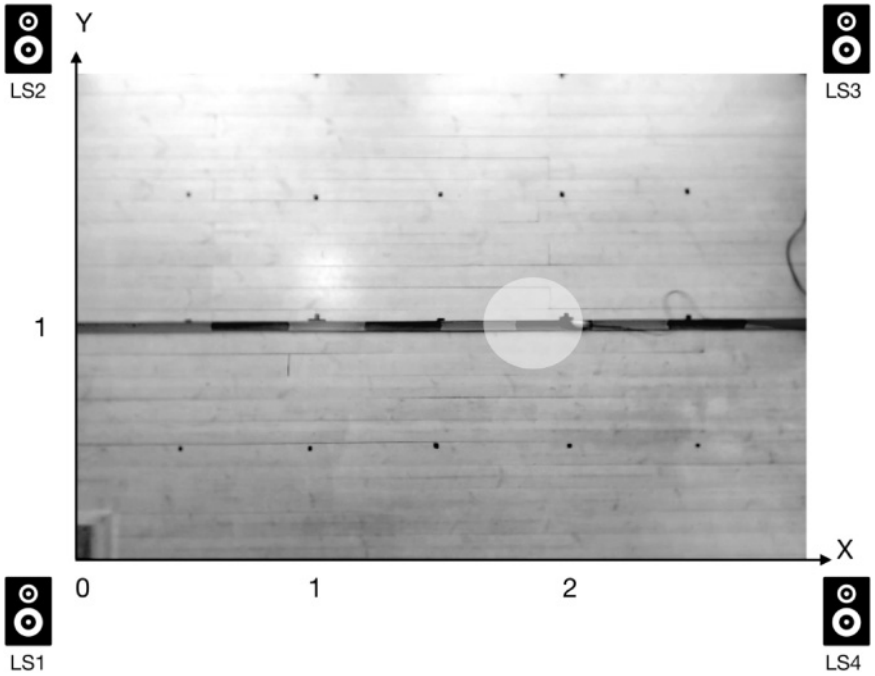


Figure 11.

Figure 12. The "faster moving device," a toy car with a microphone taped to it, was launched by hand and travelled at approximately 1.5 m/sec. Photo by author.

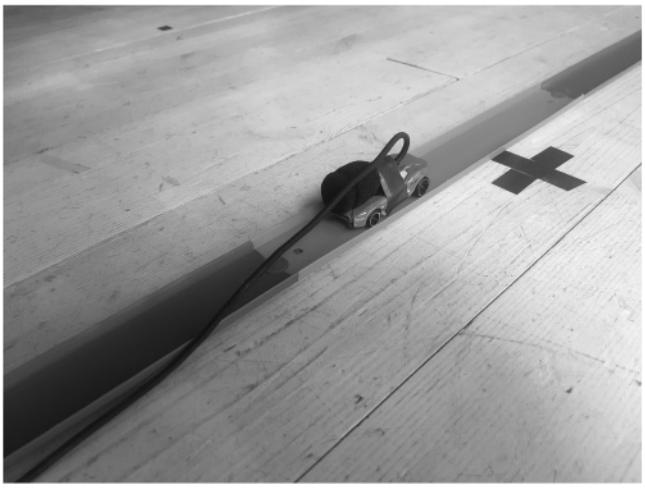


Table 2. Summary of Results from Setup B1

	MAD	*MAE*	*RMSE*	*Frequency (Hz)*	*Latency (msec)*
B1 LS1	2.8	9.7	10.7	4	250
B1 LS2	6.4	15.6	37.1	4	250
B1 LS4	2.5	18.5	56.9	4	250
Mean	3.9	14.6	34.9	4	250
B1 LS1	9.0	11.7	23.6	2	500
B1 LS7	48.7	110.3	163.0	2	500
B1 LS8	19.8	91.1	150.7	2	500
Mean	25.9	71.0	112.5	2	500

Results using ALPS Audio1. LS: loudspeaker number; MAD: mean absolute deviation (this and the next two columns in cm); MAE: mean absolute error; RMSE: root mean square error.

loudspeakers and 85 msec for eight loudspeakers, in relation to the actual computation).

Comparing the data from the experiments in B1 with the requirements set out in an earlier paper (Schlienger 2016c) indicates that "continuous update rate" can only be achieved at walking speed (approx. 1.5 m/sec). Faster speeds will result in a perceptible lag, as the measured update rate for four loudspeakers is 4 Hz, whereas for eight it is only 2 Hz. Running ALPS Audio1 in debug mode showed that the processor we used was not able to compute sufficiently rapidly to provide measurements in anything approaching real time. Yet the required accuracy of ≤0.3 m is met (see Table 1), as is the covered area, due to the scalability of ALPS. Round-trip latency (250 msec), on the other hand, is ten times higher than the requirement of keeping latency under 25 msec.

Experiments in B2

In setup B2, using the newer, al-Qt version of ALPS, no error analysis was undertaken, nor were error corrections applied to the data, as accuracy was less of a focus than latency and update rates. The results are summarized in Table 3. The higher update rates and lower RTL of al-Qt propose some interesting comparisons: Although still 6.6 times slower than world-class sprinters, who clock 100 meters in under ten seconds, the HotWheels-type car covered 1.5 m/sec and ALPS managed to catch between 13 and 17.5 measurements in that timeframe. This places it well within the requirements for update rates (cf. Schlienger 2016c). Latency between 57.1 and 77.5 msec is also four to five times less than for Audio1 in setup B1. For larger-scale applications, such as autopanning, these results improve on values that were already acceptable at a latency of 250 msec and update rate of 4 Hz. It might be worth noting that, from the listener's point of view, it is difficult to define a numerical limit as to what is an acceptable maximum latency. A minimum value could be derived from the speed of sound in air, which makes latency dependent on the observer's position relative to the sound source. Therefore, if the observer is 12 m away, a latency of 35 msec has to be expected. Yet, through the quality of the performers' actions—the "gesturality"—a link can arguably be established between a performative sound and its perception as such by the listener. And even then, perhaps thanks to a sense of syncresis (Chion, Gorbman, and Murch 1994), a causal link may be experienced beyond it being physically possible. Hence, we contend that the less-contestable measure of maximum latency should apply here: the one experienced by performers. If they manage to perform a causal relationship at a particular latency, this can be perceived as such by an observer.

Table 3. Summary of Results from Setup B2

Experiment	*Duration (sec)*	*Sum*	*NaNs*	*RTL (msec)*	*Systemic (msec)*	*Latency (msec)*	*Frequency (Hz)*
Experiments conducted with four loudspeakers and one microphone:							
B2.1	2.3	60	60	35.0	42.5	77.5	13
B2.2	2.0	46	58	35.0	42.5	77.5	13
B2.4	2.7	52	84	35.0	42.5	77.5	13
B2.5	3.5	69	111	35.0	42.5	77.5	13
Experiments conducted with four loudspeakers and four microphones:							
B2 rerun 1	2.7			14.6	42.5	57.1	17.5
B2 rerun 3	2.1			18.25	42.5	60.8	16.4
B2 rerun 5	2.2			21.0	42.5	63.5	15.7
B2 rerun 6	2.1			19.5	42.5	62.0	16.0

Results using ALPS al-Qt. Duration: duration of experiment in seconds; Sum: sum of measures taken on all loudspeakers during the experiment; NaNs: number of empty readings; RTL: round-trip latency; Systemic: latency due to configuration settings; Latency: RTL and systemic latency combined; Frequency: update rate.

The high quantity of invalid values in the data can partially be attributed not only to the cardioid directivity pattern of the microphone used but also to the decreased reliability of measurements at larger distances. By comparison, in the beginning the loudspeakers located behind the microphone are still within reach; but soon after the car is in motion, only the front-facing loudspeakers are measured.

Owing to an oversight, the experiments with the HotWheels car were initially run with only one microphone. This means that a direct comparison with setup B1, which was run with four loudspeakers and four microphones (plus reference microphone), is problematic. Therefore, it was decided to redo the experiments to clarify what influence the number of inputs has on latency. Puzzlingly, the latency was almost halved in the reruns (see Table 3 for values). It could not be conclusively determined whether this had to do with the way the macOS audio driver handles buffer sizes or with al-Qt itself.

The multiprocessor al-Qt version seemed to make good use of the highly optimized ARM64 system-on-a-chip architecture of the 2021 MacBook Air, so even when running Max on the same processor as al-Qt, none of the eight displayed cores showed significant use. What is more, there is a conspicuous incongruence in the fact that al-Qt performs worse for the simpler task of recording one microphone, but better for recording four. This might be anecdotal evidence that hardware limitations per se can be excluded as a cause.

Implication for Low-Latency Applications

The theoretical possibilities of AL indicate that gesture tracking for gestural control of musical instruments is possible. The idea at the beginning of this project was for a performance-stage-sized application, in which the position of participants and their trajectories through a room would be tracked with the accuracy and update rates given in Table 1. Be that as it may, the research was always also about gestural control of musical instruments, as one of its primary aims was to develop a generally applicable tracking tool for sonic arts. On examination of comparable examples in the literature, al-Qt provides fast and accurate measurements at higher update rates than most. Yet we seem to have reached an impasse: Aside from the fact that the approach of using a pulsed signal, stepping through pairs of inputs and outputs consecutively, limits the upwards

scaleability of the principle (the measurement process takes longer with every additional loudspeaker or microphone), we observed that measurement cycles lower than 42.5 msec do not lead to lower latency or faster update rates. The data suggests that although RTL decreases, measurements are either repeated or left out (compare with data set SetUp_C available from https://doi.org/10.5281/zenodo.5607528). When decreasing duration below 42.5 msec, the de facto update rate does not decrease and remains the same.

Not taking into consideration the systemic latencies in Table 3, since they are significantly lower in applications for gestural control, RTLs between 14.6 and 35 msec at update rates of 15 to 18 Hz are an improvement over previous efforts in the PoC described elsewhere (Schlienger 2016b). These values are within the more generous recommendations quoted in Jack, Stockman, and McPherson (2016). But, as tantalizingly close as this is to the values set by the benchmark, other researchers are encouraged to explore this further: Yes, there are indications that gestural control with latency below 20 msec and update rates over 20 Hz should be possible using AL, but verifying this is unfortunately beyond the scope of this article. Monitoring the CPU activity on the MacBook Air shows that plenty of processing power remains unused, so the issue may well lie elsewhere.

Tests in Situated Use: Qualitative Evaluation

In participatory design, evaluation and development happens concurrently, in which case evaluation by "test subjects" who are not participants in the design project are neither appropriate nor necessary—nor, by this token, is a quantification of qualitative results, which questionnaires applying numerical scaling would arguably constitute! Based on extensive field notes and discussions that form an essential part of the workshop's practice, the following text summarizes the workshop participants' experiences with the Autopan Max patch.

On the principle that situations rather than solutions should be prototyped, countless sessions of the workshop were dedicated to the scenario of moving sound sources before we progressed to experiments using the Autopan Max patch. This allowed us to observe how the practice changed with the introduction of the technology. We noticed marked differences. It was striking how nonacoustic virtual sound sources suddenly blended in among the other acoustic moving sound sources. If, in improvisations without the system, nonacoustic instruments provided a nondiegetic background or background texture for what was happening with the acoustic sources, now they were perceived as equivalent to the acoustic sources, as musical spatial actants.

In free improvisation with amplified musical instruments, it is fairly common that musicians with unamplified, quieter acoustic instruments feel that they are not heard. For example, in several early sessions of MSI, electric guitars and electronic instruments like laptops were experienced as overpowering by violin players, flute players, or singers. In discussions, the guitar was described as "amplified," a term predominantly used with a negative connotation, for example, a participant felt that the "amplified sound drowned my sound completely" or others thought the "electronic sound is too loud." Yet, in practically identical setups after the introduction of the Autopan system, which localized the amplified instruments, this term was not used in this sense in any of the discussions. From these and similar reactions to experiments with the system, it became evident that the increased localization of the panned sound increased transparency in the overall sound and created more room for the other participants.

The use of the system also allowed for the creation of multiple localized areas of different reverberant acoustics within the same performance space. In earlier sessions, before introducing the system, whenever any amplified instrument applied some form of artificial reverberation, the whole performance space was immediately immersed in that uniform artificial space, which overrode the physical space's acoustic characteristics.

This was problematic for the specifically spatial interactive practice, as the artificial spatiality was forced onto all participants, leaving no choices. For example, participants with unamplified instruments could not create reverberant rooms of their own (if

the artificial reverberation was set to "cathedral," everybody in the room had to be "in a cathedral"). In contrast, with the experiments with the Autopan system, when the reverberation was panned according to the instrument's position, the reverberation was experienced as localized. It was perceived as part of the distinct individualized quality of the sound associated with the spatial source for which it was intended.

Not only individualized, localized sound qualities have artistic validity. There are many spatio-acoustic phenomena used in musical contexts that do not rely on the listener being able to locate sound sources, but consist instead of diffuse sounds (Blesser and Salter 2007). But the workshop participants' experience was that localized qualities were hard to achieve in multiple-loudspeaker setups in improvisations before introduction of the ALPS system: It gave sound sources of both nonacoustic and acoustic origin equally spatial interactive roles.

A quadraphonic recording of an improvisation session is available for the sake of completeness (https://doi.org/10.5281/zenodo.5607027). The essence of a participatory event can only be experienced through participation; good free improvisations do not automatically make good recordings, and, as a piece of music, this would benefit from further editing. Nevertheless, it illustrates the panning trajectories.

Conclusions and Future Work

The Autopan Max patch in conjunction with the ALPS software overcomes shortcomings of standard audio technology for moving sound sources in a multiple-loudspeaker environment. It enhances immersion in spatially interactive musical performances. For the artistic idea from which this project started, ALPS and the Autopan Max system provided a relevant technological solution, at low cost, working on widely available equipment. Still, the necessary responsiveness for gestural control of musical instruments could not be completely achieved. In comparison with comparable examples from the literature, the ALPS system provides a competitive alternative and is indeed a step towards a general kinaesthetic interface.

Many optimizations are possible for the current setup—for example, tracking and filtering that are more advanced. But to achieve distinctly lower overall latency at higher update rates, even an embedded implementation would have to envisage alternatives to the data acquisition other than a pulsed signal. As one possible strategy, concurrent measuring of several output/input pairs should be envisaged.

For this project's initial artistic idea, in which accuracy, latency, and update rate must meet the requirements of spatial hearing, the autopanning system with ALPS now provides a viable and tested solution.

Acknowledgments

Many thanks to the participants in the workshop on Music, Space, and Interaction; all friends and colleagues at the Center for Music and Technology of the Sibelius Academy at the University of the Arts Helsinki, Helsinki; and Kone Foundation Finland, that has supported this project with a Researcher's Grant.

References

Aguilera, T., et al. 2017. "Broadband Acoustic Local Positioning System for Mobile Devices with Multiple Access Interference Cancellation." *Measurement* 116:483–494.

Andean, J. 2014. "Research Group in Interdisciplinary Improvisation: Goals, Perspectives, and Practice." In A. Arlander, ed. *This and That: Essays on Live Art and Performance Studies*. University of the Arts Helsinki, pp. 174–191.

Bazoge, N., et al. 2019. "Expressive Potentials of Motion Capture in the Vis Insita Musical Performance." In *Proceedings of the International Conference on New Interfaces for Musical Expression*, pp. 266–271. 10.5281/zenodo.3672954

Blesser, B., and L.-R. Salter. 2007. *Spaces Speak, Are You Listening? Experiencing Aural Architecture*. Cambridge, Massachusetts: MIT Press.

Brena, R. F., et al. 2017. "Evolution of Indoor Positioning Technologies: A Survey." *Journal of Sensors*: Art. 2630413. 10.1155/2017/2630413

Chion, M., C. Gorbman, and W. Murch. 1994. *Audio-Vision: Sound on Screen*. New York: Columbia University Press.

Dean, R. T., and G. Paine. 2012. *Gesture and Morphology in Laptop Music Performance*. Oxford: Oxford University Press.

Dobrian, C., and F. Bevilacqua. 2003. "Gestural Control of Music Using the Vicon 8 Motion Capture System." In *Proceedings of the International Conference on New Interfaces for Musical Expression*, pp. 161–163.

Filonenko, V., C. Cullen, and J. Carswell. 2010. "Investigating Ultrasonic Positioning on Mobile Phones." In *Proceedings of the International Conference on Indoor Positioning and Indoor Navigation*. 10.1109/IPIN.2010.5648235

Harter, A., et al. 2002. "The Anatomy of a Context-Aware Application." *Wireless Networks* 8(2–3): 187–197. 10.1023/A:1013767926256

Hightower, J., and G. Borriello. 2001. "Location Systems for Ubiquitous Computing." *Computer* 34(8):57–66. 10.1109/2.940014

Jack, R. H., T. Stockman, and A. McPherson. 2016. "Effect of Latency on Performer Interaction and Subjective Quality Assessment of a Digital Musical Instrument." In *Proceedings of the Audio Mostly Conference*, pp. 116–123. 10.1145/2986416.2986428

Janson, T., C. Schindelhauer, and J. Wendeberg. 2010. "Self-Localization Application for iPhone Using Only Ambient Sound Signals." In *Proceedings of the International Conference on Indoor Positioning and Indoor Navigation*. 10.1109/IPIN.2010.5648094

Lopes, C., et al. 2006. "Localization of Off-the-Shelf Mobile Devices Using Audible Sound: Architectures, Protocols and Performance Assessment." *Mobile Computing and Communications Review* 10(2):38–50. 10.1145/1137975.1137980

Lossius, T., P. Baltazar, and T. de la Hogue. 2009. "DBAP: Distance-Based Amplitude Panning." In *Proceedings of the International Computer Music Conference*, pp. 489–492.

Malham, D. 1998. "Spatial Hearing Mechanisms and Sound Reproduction." Available online at digitalbrainstorming.ch/db_data/eve/ambisonics/text02.pdf. Accessed February 2022.

Mandal, A., et al. 2005. "Beep: 3D Indoor Positioning Using Audible Sound." In *Proceedings of the Consumer Communications and Networking Conference*, pp. 348–353.

Medvedev, Y. P., A. V. Sorokin, and V. I. Khashchanskiy. 1989. Способ ввода информационных сигналов в цифровое устройство обработки [Method for Information Signal Acquisition Into a Digital Processing Device]. USSR Patent 1,508,275 A1, filed 9 October 1987 and issued 15 September 1989. Available online at yandex.ru/patents/doc/SU1508275A1_19890915. Accessed February 2022.

Mitchell, T., and I. Heap. 2011. "SoundGrasp: A Gestural Interface for the Performance of Live Music." In *Proceedings of the International Conference on New Interfaces for Musical Expression*, pp. 465–468.

Nymoen, K., S. l. A. Skogstad, and A. R. Jensenius. 2011. "SoundSaber: A Motion Capture Instrument." In *Proceedings of the International Conference on New Interfaces for Musical Expression*, pp. 312–315.

Peng, L., and D. Gerhard. 2009. "A Wii-Based Gestural Interface for Computer Conducting Systems." In *Proceedings of the International Conference on New Interfaces for Musical Expression*, pp. 155–156.

Pulkki, V. 1997. "Virtual Sound Source Positioning Using Vector Base Amplitude Panning." *Journal of the Audio Engineering Society* 45(6):456–466.

Robinson, F., et al. 2015. "Gestural Control in Electronic Music Performance: Sound Design Based on the 'Striking' and 'Bowing' Movement Metaphors." In *Proceedings of the Audio Mostly Conference*, Art. 26. 10.1145/2814895.2814901

Salazar, S., and J. Armitage. 2018. "Re-Engaging the Body and Gesture in Musical Live Coding." In *Proceedings of the International Conference on New Interfaces for Musical Expression*, pp. 386–389.

Schlienger, D. 2016a. "Acoustic Localisation for Spatial Reproduction of Moving Sound Source: Application Scenarios and Proof of Concept." In *Proceedings of the International Conference on New Interfaces for Musical Interaction*, pp. 407–412. 10.5281/zenodo.1176116

Schlienger, D. 2016b. "Gestural Control for Musical Interaction Using Acoustic Localisation Techniques." In *Proceedings of the International Conference on Live Interfaces*, pp. 161–167.

Schlienger, D. 2016c. "Requirements on Kinaesthetic Interfaces for Spatially Interactive Sonic Art." In *Proceedings of the Audio Mostly Conference*, pp. 162–169. 10.1145/2986416.2986441

Schlienger, D., and A. Olarte. 2016. "Carte Blanche, Right Now!" In *Proceedings of Art without Borders*, pp. 253–256.

Schlienger, D., and S. Tervo. 2014. "Acoustic Localisation as an Alternative to Positioning Principles in Applications Presented at NIME 2001–2013." In *Proceedings of*

the International Conference on New Interfaces for Musical Expression, pp. 439–442. 10.5281/zenodo.1178933f

Şentürk, S., et al. 2012. "Crossole: A Gestural Interface for Composition, Improvisation and Performance using Kinect." In *Proceedings of the International Conference on New Interfaces for Musical Expression*. 10.5281/zenodo.1178201

Seob Lee, J., and W. S. Yeo. 2011. "Sonicstrument: A Musical Interface with Stereotypical Acoustic Transducers." In *Proceedings of the International Conference on New Interfaces for Musical Expression*, pp. 24–27. 10.5281/zenodo.1181432

Simonsen, J., and T. Robertson. 2013. *Routledge International Handbook of Participatory Design*. Abingdon, UK: Routledge.

Trail, S., et al. 2012. "Non-Invasive Sensing and Gesture Control for Pitched Percussion Hyper-Instruments Using the Kinect." In *Proceedings of the International Conference on New Interfaces for Musical Expression*, pp. 1–4. 10.5281/zenodo.1178435

Zhang, L., et al. 2017. "Acoustic NLOS Identification Using Acoustic Channel Characteristics for Smartphone Indoor Localization." *Sensors* 17:Art. 727.