Effects of audio latency in a disc jockey interface

Laurent S. R. Simon, Arthur Vimond, Emmanuel Vincent

Proceedings of Meetings on Acoustics

Volume 19, 2013 http://acousticalsociety.org/

ICA 2013 Montreal
Montreal, Canada
2 - 7 June 2013

Musical Acoustics
Session 2pMU: Musical Preference, Perception, and Processing

2pMU10. Effects of audio latency in a disc jockey interface

Laurent S. Simon*, Arthur Vimond and Emmanuel Vincent

***Corresponding author's address: INRIA, Centre Inria Rennes - Bretagne Atlantique, Rennes Cedex, F-35042, Ille et Vilaine, France, laurent.s.simon@inria.fr**

This study presents an evaluation of the disturbance caused by audio latency in a DJing task. An experiment was conducted, during which subjects were asked to synchronise one song to a reference song using a common DJ interface. Synchronisation was performed by adjusting the speed of one of the songs to that of the reference song and time-aligning both songs. Latency was introduced between the interface and the audio output, varying between 0ms and 550 ms. The average synchronisation time was estimated as a function of subjects, Beat-Per-Minute difference between the songs and latency. Results showed that for trained DJs, synchronisation time increased significantly above 130ms of audio latency, whereas for naive subjects, latency had no influence on the synchronisation time.

Published by the Acoustical Society of America through the American Institute of Physics

 [DOI: 10.1121/1.4800187]
Received 22 Jan 2013; published 2 Jun 2013

A Disc Jockey (DJ) interface can be composed of multiple systems, such as vinyl players or MIDI DJ controllers, a mixing desk, various effects and a computer with several editing softwares. They can be either analogue or digital. When several digital interfaces are used one after another, the latencies of each of these systems are added together. The efficiency of some digital audio effects such as audio source separation algorithms or music instrument synthesis directly depends of the audio latency. It is therefore necessary, in order to optimise these audio effects, to study the influence of latency on a typical DJ task. In this paper, we introduce some of the tools DJs frequently use and some of the tasks a DJ might face. We then present an experiment designed to study the influence of the latency on the difficulty of a DJ task and its results. Finally, we discuss the results in comparison to other studies on latency.

Disk Jockey Setups and Latency Studies

A DJ is in charge of playing recordings one after each other, mixing them together and adding effects, which can vary from a sudden stop to equalisations, samples, “scratchs”. A DJ’s setup is usually composed of

- two vinyl players or MIDI DJ controllers simulating the playback of a vinyl. They let the DJ adjust the playback speed of each song (given in Beat Per Minute, or BPM) by using a pitch slider, manually slowing down the disc or using a jog wheel. Each controller has play / stop / pause buttons,
- a two-channel mixing desk that lets the user control the general level of each song,
- a patch of effects or a computer. The computer can be used to play the audio when the vinyl players are only MIDI controllers.

We started questioning the issue of latency in DJ applications in the context of the I3Dmusic project. This project aims to develop an audio source separation and remixing solution for DJs [1] [2], that is conceived as a tool that can take place between the DJ’s controllers and the sound reproduction system. Latency is a critical problem in real-time audio source separation, where waiting for sufficient information before separating the sources increases the quality of the separation and may reduce the heavy computational cost. In [1], we used buffer frames of 2048 samples, which would cause a latency of 47 ms if computation was instantaneous and separation could be performed once per new buffer. Further work showed us that a better separation could be achieved by updating the model less frequently, thus increasing latency.

How large a latency can be without disturbing the work of a DJ therefore became an important parameter of the I3Dmusic project. It is also an important parameter for every computationally intense DJ processings. Latency in a DJ interface adds a delay between the actions of the DJ and the audible consequences of that action. In that way, it is an audio-haptic latency problem.

An informal discussion with some DJs took part before an experiment was designed. We came to the hypothesis that latency was mostly critical for synchronisation of multiple songs as opposed to other DJ tasks. Synchronisation requires the DJ to adjust the speed of one song to that of a second song and to make sure that the beats of both songs are played together.

Perception of latency has been studied in uni-modal conditions [3] [4] [5] . However, research in multi-modal conditions is limited [6] [7] [8]. Obu [3] studied more particularly the perception of latency in distributed concert conditions. He showed that latency was perceived when it was higher than 50 ms but was considered to be annoying only above 150 ms. The musicians having to play along with other musicians located in a different place, this was an audio only latency study. Younkin and Corriveau [6] studied lip synchronisation, an audio-visual problem, and

showed that latency became noticeable above 185 ms. In [7] and [8], the Just Noticeable Difference (JND) of cross-modal audio-haptic asynchrony was studied. It showed that the JND varied between 24 ms and 80 ms, depending on the experimental conditions. However, no research studied how difficult latency could make a task in an audio-haptic context.

We therefore designed an experiment to evaluate the influence of latency on the difficulty of synchronising two songs.

Experimental Setup

In a night club, a DJ plays a song to the audience while trying to synchronise the next song with the current using his headphones. In this experiment, the task therefore consists in synchronising a song (labelled A) to another song (labelled B).

Dependent Variable

For this experiment, we rejected the use of direct evaluation: if we were to use direct evaluations, listeners might perceive directly the latency and be influenced by the latency to rate the difficulty of the task. Higher values of latency were always noticeable, hence subjects who would have noticed the latency only after a few stimuli might have suddenly changed their synchronisation strategy. We decided to mention in the instructions to the subjects that latency was one of the variables being tested.

We hypothesised the time taken to synchronise the two songs to be a good measure of how difficult the task is: if the task is difficult, it should take more time to achieve.

The task given to the subjects was “synchronise song A to song B so that their beats are played together consistently”. As in real world conditions, it was left to the subjects to decide when the two songs were synchronised. Adding an algorithm to verify the synchronisation of the two songs might make any trained DJ work in conditions he is not trained for.

Interface

Subjects could start playing song B whenever they wanted using a two-channel USB mixing desk but had no other control over it. A single MIDI DJ player controller was used in this experiment. It was linked to a computer via USB and controlled the playback of song A via MAX/MSP, a visual programming software designed for audio processing and music creation. Using this controller, the subject could start the song, stop it, pause it, adjust the speed of the playback via the pitch slider or by turning the DJ controller jog wheel in one direction or in the other. Starting the playback of song A for the first time triggered the chronometer. The time counter would stop when the user would adjust the mixing desk to play only song A and would not change the settings for 5 seconds.

Independent Variables

In this experiment, song B was defined as the reference song. In order to avoid having a too large number of variables, song B was choosen to be the same song during the whole experiment, *Flat Beat* by Mr Oizo. Its speed was fixed to 126 Beats Per Minute (BPM), a reference speed for DJs, according to the informal discussion with DJs before the design of the experiment.

Song A alternated between 5 different songs in a pre-defined order. They were all similar electronic songs chosen after pre-screening, and included a strong beat. They all started with a strong rhythmic and began on the first beat. These five songs gave similar synchronisation time

during preliminary tests of the interface for all latencies and differences of BPM. The alternation of songs prevented any memory effect.

Indeed, if any version of the songs A had been played directly at 126 BPM, the task of the subjects would have only consisted in starting the song at the correct time. During the tests, song A was therefore played at a speed uniformly randomised in one of four different BPM intervals: [118 122], [122 126], [126 130], [130 134].

Five different latency values were tested at each speed interval: 0 ms, which was used as a reference, 130 ms and 170 ms (to each side of the latency limit that caused annoyance, according to [3]), 300 ms and 550 ms. These two latter values were chosen in order not to correspond to any simple ratio of a beat (476 ms at 126 BPM).

A stimulus was a combination of a latency and a difference of BPM. All combinations of latency and difference of BPM were tested for each subject, leading to a total of 20 stimuli per test. The order in which those combinations were presented to subjects was randomised by the MAX/MSP patch. During the test, one of the five songs A was associated to each of the stimuli in a pre-defined order, as explained above.

Pannel of Subjects, Training and Screening

A total of 17 subjects, male, aged between 20 and 30 years old, took part to the test. 8 of them were trained to the task of DJing and 9 of them had had little or no previous training. Each test began with a 10 min training session. During the training session, subjects had so synchronise each of the songs, without any latency, and could stop the experiment at any time to ask questions.

The training session was then followed by the main test, composed of 20 stimuli to synchronise to the song B, mixing all of the latency conditions and all of the differences of BPM. In order to be able to screen the unreliable subjects, in each test, 4 of the stimuli were chosen randomly and repeated at the end of the test. The time subjects took to complete a whole test varied from one subject to the other, with an average of 50 min per test.

After taking the test, most of the untrained subjects stated having too much trouble with the task and explained that they only felt comfortable with it at the end of the test. It was therefore decided to have all of the untrained subjects take the test once more on another day, to avoid tiredness effect. This second test also included the training session. The data shown in the results for untrained subjects is that of the second session. The trained subjects only took the test once.

Results

Subjects Reliability

The task the subjects were required to perform was multi-modal. It was therefore not possible to use any of the matching tests, detection tests or discrimination tests usually recommended for audio listening tests [9].

Hence for each subject, and each of the repeated stimuli, reliability was estimated as the synchronisation time difference between the repetition and the first occurrence of the stimulus, shown as a percentage. A percentage of 100% indicates that the subject took twice as long to synchronise the repetition as to synchronise the first occurence of the stimulus. A percentage of -50% indicates that the subject took half as long to synchronise the repetition as to synchronise the first occurence of the stimulus. The metric is therefore not symmetrical. However, when the subject is found unreliable, the sign of each of the repetition's reliability is a clue to the reason of its unreliability:

- if all the repetition's reliability values are positive, unreliability might come from tiredness of the subject, as he systematically took longer to synchronise the repetition than the first occurence of the stimulus
- if all the repetition's reliability values are negative, unreliability might come from learning effect, as the subject systematically took less time to synchronise the repetition than he took to synchronise the first occurence of the stimulus
- if the repetition's reliability values do not all have the same sign, the unreliability does not have any clear origin.

All of the reliability values followed the third case. Absolute values of the reliability values were taken and averaged for each subject. Fig. 1 shows the unreliability of each subjects. The reliability threshold was set to 20%. This value may seem large. However, a part of chance is involved in the task the subjects were required to conduct, as larger latency values led to a difficult to predict start of the song. The larger threshold was thought to compensate for this. 8 subjects out of the 17 initial subjects were considered as reliable. 5 of them where trained subjects and 3 of them were untrained subjects.

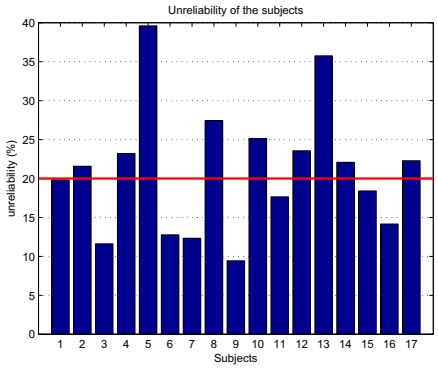

FIGURE 1: Unreliability of the subjects. Subjects 1 to 8 are trained subjects. Subjects 9 to 17 are untrained subjects.

Main Results

Results were analysed with an n-way ANalysis Of VAriance (ANOVA) in MATLAB [10]. It estimated the influence of the latency and difference of BPM on the synchronisation time. For each independent variable a significance (sig.) larger than 0.05 indicates that any observed effect is likely to be an effect of chance, and that the variable therefore has no significant influence on the dependent variable (synchronisation time).

In order to be able to use ANOVAs, data within groups need to be normally distributed. Results were positively skewed. A Lilliefors test was then performed in Matlab on the square root of the results. It showed that each group's results, as well as the results of the totality of subjects, are normally distributed. ANOVA were therefore performed on the square root of the results.

As can be seen in table 1, neither the latency nor the difference of BPM nor the interaction between the two have a significant influence on the results of the reliable subjects (sig. > 0.05). This was highly unexpected, as all of the subjets reported that the task was more difficult when the latency was high. We therefore assumed that the method used to test the reliability of the subjects might have been inadequate.

TABLE 1: Results of the n-way ANOVA conducted on the synchronisation time for all the subjects estimated as reliable.

Tests of Within-Subjects Effects

Source	Sum of Squares	df	Mean Square	F	Sig.
Latency	78244.4	4	19561.1	1.64	0.16
BPM difference	44225.4	3	14741.8	1.23	0.30
Latency * BPM difference	65416.3	12	5451.4	0.46	0.93
Error	1674126.8	140	11958		
Total	1862012.9	159			

TABLE 2: Results of the n-way ANOVA conducted on the synchronisation time for trained subjects.

Tests of Within-Subjects Effects

Source	Sum of Squares	df	Mean Square	F	Sig.
Latency	182451.7	4	45612.9	8.5	0
BPM difference	18509.4	3	6169.8	1.15	0.33
Latency * BPM difference	82966.9	12	4913.9	0.92	0.53
Error	751106.1	140	5365		
Total	1011034.2	159			

Separately analysing the significance of the independent variables for trained and untrained subjects gave different results. Table 2 shows that for trained subjects, latency had a significant influence on the synchronisation time (sig. = 0 and F = 8.5) but that the difference of BPM had not (sig. = 0.33) and neither had the interaction between the latency and the difference of BPM (sig. = 0.53). Fig. 2 shows the average synchronisation time for trained subjects for all songs and all differences of BPM as a function of latency. It can be seen that the synchronisation time increases significantly above a latency value of 130 ms.

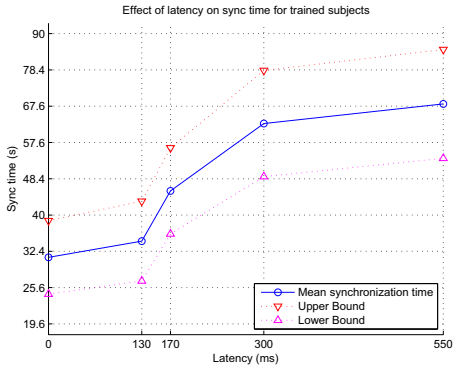

FIGURE 2: Synchronisation time as a function of latency for trained subjects.

The analysis of variance for untrained subjects, who had taken the test twice, shows that none of the independent variables had a significant influence on the synchronisation time, as can be seen in table 3. Fig. 3 confirms this visually, as the confidence intervals of the synchronisation time for all latency values overlap.

TABLE 3: Results of the n-way ANOVA conducted on the synchronisation time for untrained subjects.

Tests of Within-Subjects Effects

Source	Sum of Squares	df	Mean Square	F	Sig.
Latency	15881.7	4	3970.4	0.31	0.87
BPM difference	56299.3	3	18766.4	1.46	0.22
Latency * BPM difference	63353.6	12	5279.5	0.41	0.95
Error	2059985.5	160	12874.9		
Total	2195520.1	179			

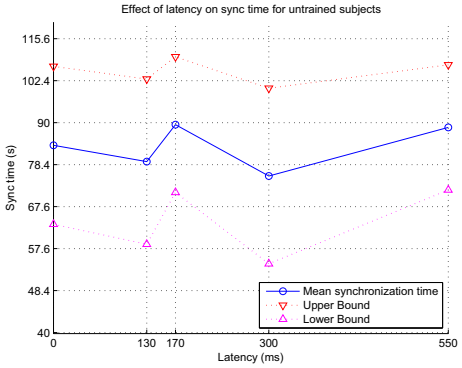

FIGURE 3: Synchronisation time as a function of latency for untrained subjects.

DISCUSSION

The estimation of reliable subjects was biased. The random component of the task means that a larger number of repetitions would have been required. However, this would have made the experiment considerably longer and more tiring.

Another possible explanation for the non-significant results obtained when selecting those subjects which were estimated to be reliable is that a difference of 30 seconds of synchronisation time between the original presentation of a stimulus and its repetition is not penalised as much if the subject takes 2 minutes to synchronise the song as if he takes 30 seconds to do it. This was initially desired, as untrained subjects would always take longer to perform this difficult task than trained subjects.

Additionally, for some untrained subjects, the task might have been too difficult. They might therefore have often decided to validate the synchronisation of songs after a given time, even if the songs were not correctly synchronised.

Finally, nothing prevented the subjects to validate the synchronisation if the two songs were not correctly synchronised. This was initially wanted, as in real conditions, nothing prevents a DJ from playing a song that is not synchronised to the current song. However, might also have made the task too difficult for non-DJ subjects, thus decreasing their reliability.

This article does not show the annoyance threshold caused by latency. However, it shows that the task of DJing becomes noticeably more difficult when latency is larger than 130 ms, a conclusion similar to that of [3], even if the conditions of test were different.

Above 300 ms, a synchronisation time threshold begins to appear. The task becomes so difficult that synchronising successfully the two songs mainly relies on chance. This was confirmed by the subjects verbal reports.

After the test, an informal discussion with the trained subjects was held. Its conclusion was that most DJ thought they would be able to work with a latency of up to half a beat (238 ms for a song at 126 BPM). Our results nevertheless shows that at this latency value, the task is much more difficult than when latency is null.

Conclusion

This experiment showed that above 130 ms of latency, synchronising the beats of two songs becomes significantly more difficult. This result is similar to that of uni-modal audio studies concerning distributed music [3]. Above 300 ms of latency, the task becomes less dependent on latency, as this task becomes too difficult. Assessing the reliability of the subjects was a challenge and the solutions tested during this experiment proved to be insufficient. As discussed above, increasing the number of repetitions would be necessary. Using the time difference and not a difference ratio, as well as preventing the subjects from validating the synchronisation when the beats of the two songs are not synchronised, should lead to a better selection.

ACKNOWLEDGMENTS

This work was supported by the EUREKA Eurostars i3DMusic project funded by Oseo.

References

[1] L. S. R. Simon and E. Vincent, “A general framework for online audio source separation”, in *Proc. 2012 International conference on Latent Variable Analysis and Signal Separation* (Tel-Aviv) (2012), URL `http://hal.inria.fr/hal-00655398/en/`.

[2] E. Corteel, L. Rohr, X. Falourd, K.-V. NGuyen, and H. Lissek, “Practical 3 dimensional sound reproduction using wave field synthesis, theory and perceptual validation”, in *Proceedings of the 11th French Congress of Acoustics and 2012 Annual IOA Meeting* (2012).

[3] T. K. Yuka Obu, “M.A.S.: a protocol for a musical session in a sound field where synchronization between musical notes is not guaranteed”, Technical report of IEICE. Multimedia and virtual environment **103**, 15–18 (2003), URL `http://hdl.handle.net/2027/spo.bbp2372.2003.016`.

[4] C. Aymoz and P. Viviani, “Perceptual asynchronies for biological and non-biological visual events”, Vision research **44**, 1547–1563 (2004), PMID: 15126064.

[5] J. C. Craig and B. H. Xu, “Temporal order and tactile patterns”, Perception & psychophysics **47**, 22–34 (1990), PMID: 2300421.

[6] A. Younkin and P. Corriveau, “Determining the amount of audio-video synchronization errors perceptible to the average end-user”, IEEE Transactions on Broadcasting **54**, 623 –627 (2008).

[7] B. D. Adelstein, D. R. Begault, M. R. Anderson, and E. M. Wenzel, “Sensitivity to haptic-audio asynchrony”, in *Proceedings of the 5th international conference on Multimodal interfaces*, ICMI ’03, 73–76 (ACM, New York, NY, USA) (2003), URL `http://doi.acm.org/10.1145/958432.958448`.

[8] A. Nishi, M. Yokoyama, T. Ogata, T. Nozawa, and Y. Miyake, “The effect of voluntary movement on audio-haptic temporal order judgment”, in *2011 IEEE/SICE International Symposium on System Integration (SII)*, 649 –654 (2011).

[9] N. Zacharov and S. Bech, *The Perceptual Audio Evaluation: Theory, Method and Application* (John Wiley & Sons) (2006).

[10] A. Field, *Discovering Statistics Using SPSS*, third edition (SAGE Publications Ltd) (2009).