EFFECT OF LATENCY ON PLAYING ACCURACY OF TWO GESTURE CONTROLLED CONTINUOUS SOUND INSTRUMENTS WITHOUT TACTILE FEEDBACK

Teemu Mäki-Patola and Perttu Hämäläinen

Telecommunications Software and Multimedia Laboratory
Helsinki University of Technology, Finland
tmakipat@tml.hut.fi pjhamala@tml.hut.fi

ABSTRACT

The paper reports results from an experimental study quantifying how latency affects the playing accuracy of two continuous sound instruments. 11 subjects played a conventional Theremin and a virtual reality Theremin. Both instruments provided the user only audio feedback. The subjects performed two tasks under different instrument latencies. They attempted to match the pitch of the instrument to a sample pitch and they played along a short sample melody and a metronome. Both the sample sound and the instrument's sound were recorded on different channels of a sound file. Later the pitch of the sounds was extracted and user performance analyzed. The results show that the time required to match a given pitch degrades about five times the introduced latency suggesting that the feedback latency cumulates over the whole task. Errors while playing along a sample melody increased 80% by average on the highest latency of 240ms. Latencies until 120ms increased the errors only slightly.

1. INTRODUCTION

Currently, physical sound modeling is an active research area. Real-time sound production makes it possible to alter any parameter of the sound model while playing. This creates a need for controllers whose input flexibility matches the complexity of the sound model. Virtual reality input technology, such as data gloves and location/orientation trackers with gesture analyses, is one option that offers several degrees of freedom. We are currently experimenting with virtual reality interfaces for the control of physical sound models in an EU funded project called ALMA [1].

Physical sound models are often computationally heavy, which introduces some latency. A virtual reality system also always introduces latency. Latency is a key issue also in networked co-operative playing. Thus, it is of importance to know how much latency can be allowed for different control paradigms.

An article by Paradiso [2] and a book edited by Wanderley and Battier [3] offer a good overview of existing electronic interfaces and controllers. Many have been created, especially during the last few decades. However, only few virtual reality interfaces for sound control exist [4], [5]. They have been interactive sound environments or interactive filters rather than standalone instruments. The interfaces and alternative controllers have been reported mostly as case studies.

There seems to be a lack of quantitative comparisons of the suitability of different interfaces for controlling sound. The importance of parameter mapping has only lately been considered [6], [7]. A comparison of three input devices for timbre space navigation exists [8]. The preliminary parameter mapping observations offer some suggestions to what direction to move on the issue. It would be beneficial to have similar guidelines based on properties of available input technology and its suitability for different kinds of sound control.

Earlier research suggests that tactile feedback improves playing accuracy of an instrument [9]. However, tactile feedback is currently difficult to elegantly integrate into virtual reality interfaces. Thus, if we want to use virtual reality for controlling sound it is of interest to know latency tolerance for cases where the subject does not obtain tactile feedback while playing an instrument.

Several studies have shown that latency degrades user performance in virtual reality [10], [11]. The degradation is gradual and depends of the task. The mentioned studies concentrated on reaching and target acquisition tasks. Feedback was visual and minimum latencies as high as the maximum latency in our test. Similar results by Watson et al. [12] show that latency slows down and reduces placement accuracy when the task requires feedback. They also studied the effect of variations in latency [13] concluding that only variations with standard deviation above 82ms affect performance in a grasping and placement task.

A classical experiment conducted by Michotte and reported by Card, Moran and Newell [14] shows that users perceive two events as connected by immediate causality if the delay between the events is less than 50ms. Dahl and Bresin [15] suggest that over 55ms of latency degrades use of a percussion instrument without tactile feedback while playing along with a metronome. Again the degradation was gradual. Only four professional musicians were tested with a baton instrument. The latency was increased in small steps while playing. Two subjects were tested also with tactile feedback (MIDI drum), concluding that the standard deviation of the flutter of consequent hits increased with increasing delay. However, again the change is slow. The standard deviation is no larger at 50ms than at zero. After 50ms it seems to rise gradually. The amount of subjects and samples in the test was too small for strong conclusions. The study verified also a hypothesis that the performer attempts to compensate the delay by matching sound to sound when he has to synchronize with other audio sources. Finney has shown that delay in auditory response caused large errors in performance of pianists [16]. There was no degradation if the performer did not receive auditory feedback. Discrepancy with sound and tactile feedback seems to be the main source of instrumental problems.

Less than 10ms latencies are often suggested for a music controller [17], [18] as professional piano players might already notice that much. However, tolerable latency is dependent on type of music, type of instrument sound [19] and presence or absence of tactile feedback. We have earlier found that the just noticeable difference (JND) for latency for a continuous sound gesture controlled instrument without tactile feedback is about 30ms [20]. For an extreme perspective let us remind that latencies as high as several hundred milliseconds are not rare for church organs and yet they can be played when also practiced with the same latency.

2. USER TEST

The goal of our user test was to quantify how latency degrades the control accuracy of the tested gesture instruments. The test consisted of two parts on two similar instruments. In the first part the subjects heard a permutation of 16 consecutive sine wave notes on their left earphone. On their right earphone they heard the Theremin or virtual reality Theremin instrument they controlled. Each sample note played for five seconds during which the subject attempted to produce the same tone with the instrument. On the second part the subjects were played a short song (see Figure 1) seven times accompanied with a metronome. The subjects tried to play along the song as well as they could.

Each subject did both tests on five different latencies on the original Theremin and on three different latencies on the virtual reality Theremin. The latencies tested were 0ms (no latency), 30ms, 60ms, 120ms and 240ms on the original Theremin and 60ms, 120ms and 240ms on the virtual reality Theremin. The responsiveness of our virtual reality system was measured to be 60ms with a standard deviation of 8ms. Thus, only the last three latencies could be tested on this instrument.

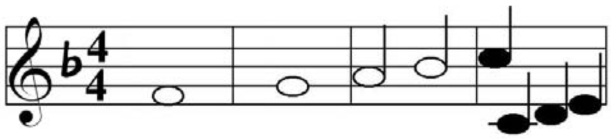

Figure 1: *Simple example melody used in the play-along test. The melody was played in a tempo of 120BPM. It is progressive, simple and uses all note lengths common for Theremin music.*

Each individual pitch-matching test lasted for 80 seconds. The play-along tests were 56 seconds long each. Thus, the original Theremin part of the test took 11 minutes 20 seconds plus the practice and one minute breaks between the tests. The virtual reality Theremin part took 6 minutes and 48 seconds plus the practice and breaks. The whole test took about 40 minutes per subject.

The pitch matching permutation was randomised before the tests and was the same in each test for all subjects. The results were thus easily comparable to each other. None of subjects noticed that the notes came in the same order when asked after the test.

2.1. Subjects

The test subjects were 11 students and researchers from the Helsinki University of Technology. All of the subjects had at least six years of musical instrument practice; seven had more than 10 years of practice with several instruments. Four of the subjects currently practice more than five hours per week, five subjects for 1 to 3 hours and the remaining two less than an hour or not at all. Six of the subjects were 23 to 28 years of age, the rest were 30 to 50 years of age. Only one subject was female. One subject was left-handed. Nine of the subjects had previously participated in a test using the Theremin instrument.

2.2. Test equipment

The Theremin's [21] output was routed to a Boss GX-700 Guitar Effects processor. Using the effects processor the instrument's sound could be delayed for a specified amount of milliseconds. The effects processor was preprogrammed with patches that had only a delay effect active and no direct sound. We tested the effects processor itself to produce less than 1ms of delay when the delay was deactivated. The output of the effects processor was routed to the right earphone of the test subject and to the laptop used for recording the session data (see Figure 2). Another computer was used for generating the sample pitches and melodies.

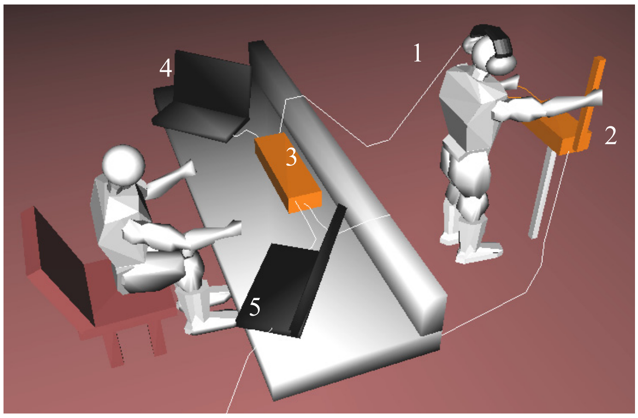

Figure 2: *Test setting for Theremin. The subject (1) hears sample sounds from his left earphone and the Theremin instrument's (2) sound from his right earphone. The instrument's sound is delayed using a guitar effects processor (3). Both sounds are inputted also to a laptop computer (4) that records both sounds on different channels of a sound file for later pitch extraction. Another computer is used for producing the sample sounds (5) and also to produce a metronome sound on a speaker (not visible in figure).*

The virtual reality version of the Theremin used our laboratory's Cave-like virtual reality room, EVE. EVE uses an Ascension Technologies MotionStar magnetic tracker to track the location and orientation of the subject's data gloves. The tracker reads the location and orientation of the sensors at a rate of 100Hz. We wanted the subject to receive only audio feedback. Thus, EVE's stereo goggles were not used in the test and the visualization of the virtual reality Theremin was switched off. The EVE system uses 5DT data gloves to measure the flexure of each finger.

The test setting for the virtual reality Theremin was similar to the test setting on the original Theremin. However, the delay was implemented on our audio program Mustajuuri [22] and not on the effects processor. Mustajuuri is a plugin based DSP program and serves as the sound control system of the EVE. It can be controlled over a local net from an SGI Onyx computer running the virtual reality application.

The sample sound and the instrument sound were outputted on different channels and routed similarly than in the test using the original Theremin. The Mustajuuri application was used in the other computer for procucing all sample sounds in both Theremin and virtual reality Theremin tests. The virtual reality Theremin's sound was produced in the same instance of Mustajuuri.

In the virtual reality Theremin the height of the subject's right hand controlled pitch. Closing the hand controlled volume. The pitch scale represented a vertical keyboard but was continuous.

The original Theremin's sound is controlled through interaction with its two antennas. The distance from the player's left hand from a horizontal loop shaped antenna controls the volume, the closer the hand to the volume antenna the smaller the amplitude of the sound. The distance of the user's hand from a vertical antenna defines the pitch of the sound, the closer the hand the higher the frequency. The user is part of the instrument's sensitive capasitance circuit.

2.3. Test procedure

The order of the test instruments was randomly chosen for each subject. On both instruments, the subject did first all pitch-matching tests and after that all play-along tests. The order of the latencies was randomized. Latencies were changed only between individual tests.

2.3.1. Original Theremin test

In the beginning of the half of the user test that used the original Theremin, the subject was first introduced to the instrument by letting him play it for couple minutes. Then the test procedure was explained and one 80 seconds long pitch matching was practiced without any latency in the instrument's response. If the subject wanted to practice one more time it was allowed. Some subjects did. During the test the subject matched the 16 sound samples on all five latencies keeping a short break after each individual test.

After the pitch-matching test the sample melody was introduced to the subject and he was told to try to play along the melody as accurately as he could. Playing along the melody was practiced without latency for one or two test periods depending on when the subject felt ready for the test. After this the subject attempted to play along the melody on all five latencies. Each play-along test repeated the melody seven times on every latency setting. A one-minute break was kept after each individual test with different latency setting.

2.3.2. Virtual reality Theremin test

The virtual reality Theremin test was similar to the test using the original Theremin. Again the pitch-matching test was first. The user was allowed to play the instrument for few minutes and then practice one or two 80 second long pitch-matching tests with the minimum system latency of 60ms. After this the pitch-matching test began as with the original Theremin followed by the melody test. However, only the mentioned three latencies were tested.

3. RESULTS

The results indicate that the time taken to match a given pitch with the instrument lengthens about five times the introduced latency. When latency increases 60ms the matching time rises about 300ms. This suggests that the subject uses feedback several times/continuously during the task. First he moves quickly up or down depending on the relative location of the new target pitch. While he moves he uses the sound as feedback of making new estimates, just like an optimization algorithm or a mathematical control system. Latency cumulates over the whole task. The relative increase in matching times from 60ms to 240ms was 45%.

3.1. Valid subjects

Three of the eleven subjects were removed from the results in the final data analyses because they could not hear the differences in the pitches well. These subjects matched the pitches with an error of several halftones. The error was not a consistent amount of notes up or down but altered even during one test recording. The removed subjects were the ones with the least musical practice. Two of the three had not played any instrument for several years.

3.2. Extracting the pitch

We used Matlab for data analyses. The pitch was extracted from the recorded sound files by calculating piecewise Fourier transform for every consecutive 128 samples. The sample rate of the recording was 8000Hz on a 16bit resolution. Every piece was first zero padded to 256 samples resulting in double over sampling. Then the maximum spike (see Figure 3) was searched from each transformed slice with parabolic interpolation and the location of the maximum scaled to a MIDI key number scale.

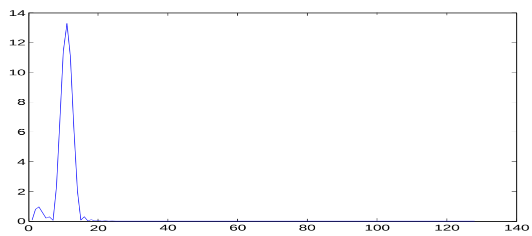

Figure 3: *Frequency spike on a Fourier transformed 128 samples long piece of a test sound file.*

The pitch was extracted from all of the recorded sound files. Figure 4 shows the extracted pitch of one such file. Each subject's data produced eight similar pitch-matching curves and eight play-along melody curves.

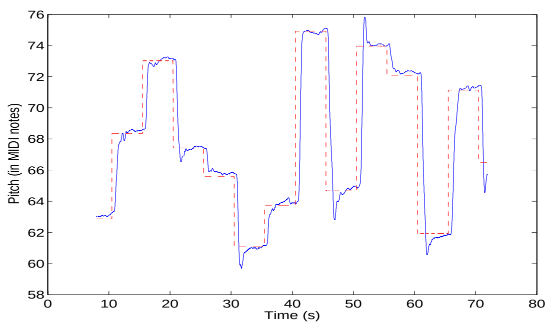

Figure 4: *Pitch extracted from one pitch-matching test. The dashed line is the example pitch. The solid line is the pitch from the subject's instrument. The first eight seconds were cut away from each test to give the subject time to start the test.*

3.3. Effect of latency in the pitch matching task

For the pitch-matching task we wanted to know how much longer it takes to reach the goal pitch when latency is introduced to the instrument's response. We made a Matlab program that searched each sample pitch change from every pitch test file. It then isolated the next five seconds that the sample pitch remained unchanged while the subject was trying to match the instrument's sound with it. We wanted to determine how long each matching took before the goal pitch was reached. Then the 14 matching times for each individual test were averaged to get the subject's average pitch-matching times for each latency setting.

As there were 128 matching periods for each subject we needed some automatic measure for deciding when the subject had reached the goal pitch. We made an algorithm that evaluated each matching period to determine the time when the subject had reached the goal pitch. The algorithm moved a 500ms window forward in the five seconds long matching data. Two error measures were defined for the window (equations (1) and (2)).

$$E_1(t) = \frac{1}{N}\left(\sum_{n=1}^{N} abs(y(t-1+n) - x(t-1+n))\right) \quad (1)$$

In equation (1) the error E(t) is defined for a time t. y(t) is the sample pitch and x(t) is the instrument's pitch at time t. The pitches have been scaled to a halftone MIDI key scale. The first error measure defines the average deviation from the target pitch inside the window of length N in semitones.

$$E_2(t) = Max(abs(y(t+n) - x(t+n))), n \in [0, N-1] \quad (2)$$

The second error measure defines the maximum deviation from the target pitch inside the window. We defined that the subject had reached the target pitch when the relationship of equation (3) was satisfied.

$$E_1(t) < 0.5 \wedge E_2(t) < 0.5 \quad (3)$$

Thus, when the average error and the maximum error inside the window, forward from the point t, are less than a quartertone the pitch has been matched. The first t that satisfies equation (3) is marked as the time of reaching the target pitch. If no t satisfies the equation the matching time is marked as the full five seconds.

The average matching time was calculated for every latency setting for each individual. The individual results where then averaged over the test population. The matching times were calculated also with a 800ms window. The two windows gave quite similar results. The final results are the average of the results from analyses on both window lengths. They are presented in Table 1. Figure 5 shows an example of six five second long pitch-matching tasks extracted from the test data.

Latency	0ms	30ms	60ms	120ms	240ms
Original Theremin	2040	2440	2260	2610	3270
Normalized	0.90	1.08	1.00	1.16	1.45
Virtual Theremin	-	-	1930	2260	2780
Normalized	-	-	1.00	1.17	1.44

Table 1: *Average pitch-matching times in milliseconds as a function of instrument latency. The normalized values are expressed as multiples of the matching time on the 60ms latency. The relative increase in matching times is very similar on both instruments.*

As can be seen from Table 1 the results from the last three latencies from both instruments are highly consistent. The matching times grow 350ms and 330ms when the instrument latency rises from 60ms to 120ms. The matching times grow 660ms and 520ms when another 120ms of latency is introduced. Considering all changes after the 60ms latency the matching times grow 5.8, 5.5, 5.5, 4.3, 5.6 and 4.7 times the added latency. After 60ms, the matching time grows 5.2 times the amount of added latency by average. The relative chance in the matching time is 45% from 60ms latency to 240ms latency.

The matching time with 30ms latency is larger than the matching time with 60ms of latency. We state the possible reasons for this in Section 4.

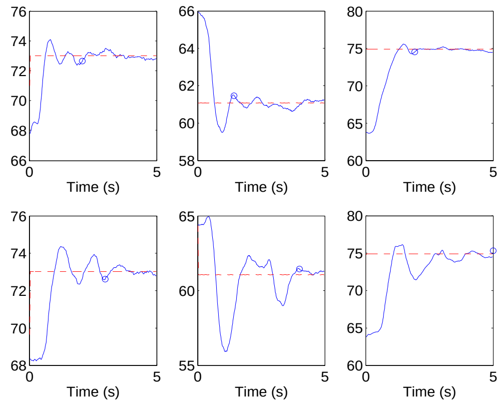

Figure 5: *Three individual pitch-matching curves of the same subject on the original Theremin. Upper row with no latency and the lower row with 240ms latency. It is clearly seen how larger latency causes more fluctuation around the target pitch (dashed line). The small ball on the curve marks the place where our algorithm decided that the target pitch had been reached.*

Table 2 presents an estimate of how much the matching curve fluctuates as a function of latency. The values were calculated by integrating the absolute difference of the sample pitch and the instrument's pitch and dividing it with the length of the integrated area.

Latency	0ms	30ms	60ms	120ms	240ms
Original Theremin	1.26	1.31	1.31	1.52	1.93
Normalized	0.96	1.00	1.00	1.16	1.47
Virtual Theremin	-	-	1.01	1.17	1.29
Normalized	-	-	1.00	1.16	1.28

Table 2: *The average difference in half tones of the subjects' playing and the sample signal on the pitch-matching task. The normalized values are expressed as multiples of the average difference on 60ms latency.*

The matching times of changing the pitch downwards were similar to the matching times of changing the pitch upwards. They differed less than 80ms by average. However, with the maximum latency of 240ms on the normal Theremin the matching times upwards were by average 700ms slower than downwards.

Differences of six of more half tones took 900ms more time to match by average than differences of less than 3 half tones. Interestingly this did not change as a function of latency but was nearly the same under all latencies. Matching times of all pitch differences shifted almost equally as a function of latency. Maximum differences in the data were 11 notes up and 15 notes down.

3.4. Effect of latency while playing along background music

Figure 6 shows an example performance of a subject playing along the sample melody on two different latencies. Table 3 shows the average pitch error during the playing tests in the whole population. As can be seen the differences between latencies of zero to 120ms are small. We assume that they fit inside the noise of the data. Only the 240ms latency brings forth clear performance degradation.

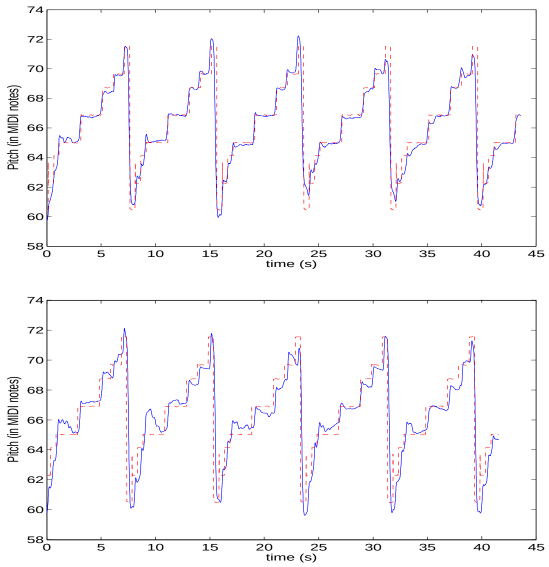

Figure 6: *Example patterns from one subject's play-along tests. The solid line is the pitch of the Theremin instrument, the dashed red line is the pitch of the sample melody. The above graph is with zero latency and the lower graph is the same subject with 240ms latency. The first 16 seconds were cut away from all play-along data to give the subject time to start the test.*

Interviewing the subjects it was found out that they relied mostly on kinaesthetic memory while playing along the music. They quickly learned the tune and the approximate hand locations for its notes and took little advantage of the audio feedback. They compensated the latency well on the range from zero to 120ms. The playing turned clearly more difficult only on the latency of 240ms. This largest latency seemed to be too much to compensate and the small refinements based on the audio feedback started to fluctuate.

Latency	0ms	30ms	60ms	120ms	240ms
Original Theremin play-along	0.69	0.69	0.58	0.65	1.12
Normalized	1.19	1.19	1.00	1.13	1.93
Virtual Theremin play-along	-	-	0.80	0.92	1.32
Normalized	-	-	1.00	1.15	1.64

Table 3: *The average difference of the instrument and the sample signal in half tones on the play-along task. The normalized values are the errors divided by the error at the 60ms latency. The performance degrades 93% on the normal Theremin and 64% on the virtual reality Theremin as the latency rises to 240ms.*

4. DISCUSSION

As can be seen from Table 1 the pitch matching times are smaller on the virtual reality Theremin compared to the original Theremin. Some of this is probably due to the slightly different user interface (height of the hand instead of the distance from a pole). Another reason is that the example sound is exactly similar to the instrument sound on the virtual reality Theremin. The original Theremin used in the test does not produce exact sine wave sound but rather heavily smoothed saw wave. The different color of the sound makes it a bit more difficult to match the pitches.

In the pitch-matching task the virtual reality Theremin seems to be more accurate than the original Theremin. However, the play-along test results are more erroneous on the virtual reality Theremin. This might be because the virtual reality theremin requires longer movements between the melody pitches, as the scale is less dense.

As mentioned, the matching times upwards were 700ms slower than downwards with 240ms latency on the original Theremin. This did not happen on the virtual reality Theremin. We suspect that the reason is the nonlinearity of the original Theremin's scale in the high pitch end. A one-centimeter difference in the hand location becomes several tones when close to the pitch antenna. With such exact movements the iteration time is bound to rise when the feedback latency is high. The virtual reality Theremin had a tone-wise linear scale. The same amount of height difference always resulted in the same amount of change in half tones.

Table 1 and Table 2 show interestingly similar behavior for the increase in pitch matching times and average error. It should be noted that these are rather different things. Although that the matching time fluctuates under the 60ms latencies the average error is very similar on those latencies.

There is quite much noise in the data of individual subjects. Some of it is due to learning. Especially in the play-along test the subjects got better and better kinaesthetic memory of the hand poses for the melody. The order of the latencies was randomised but as we had only eight valid subjects it still leaves some noise to the data from learning and fluctuations in the subject's concentration. The matching time with 30ms latency was larger than with 60ms latency. This is likely due to chance. Note that the error measure (Table 2) was the same for both cases.

The function for evaluating the matching times affects the answers a bit. With stricter error limits the lengthening of the matching times is a bit more severe. However, the relative differences of the answer distribution remain similar.

Dahl and Bresin's study [15] with percussion instruments suggested that in the presence of background rhythm the subject attempts to match sound with sound. In the play-along part of our test the same thing was evident. The subjects compensated the latency quite well on all latencies except for the largest latency.

In the virtual reality Theremin the latency was not constant but had a standard deviation of 8ms. This variance was so small that its possible effects are shadowed by other sources of errors such as the noise in the data.

5. CONCLUSIONS AND FUTURE WORK

It was found that starting from the latency of 60ms the time required to match a given pitch lengthens by roughly five times the introduced latency. This suggests that the feedback latency cumulates over the whole task. The matching time differences on latencies below 60ms were within the noise threshold. Errors in following the sample pitch increased 40% on the maximum latency of 240ms as the subject's playing started to fluctuate around the goal value. Starting the matching process over six half tone distances away from the goal pitch required 900ms (47%) more time compared to less than 3 half tone distances. Interestingly this time was not significantly increased by latency.

Errors while playing along a sample melody increased 80% by average on the highest latency of 240ms. Latencies until 120ms increased the errors by less than 20%. The subjects still managed to compensate the 120ms latency but not the maximum latency of 240ms. Interestingly the subjects produced least errors on 60ms latency while playing along a sample melody. In our preliminary studies there were also other anomalies around 60ms latency. It could be that this time constant has some special characteristic in human physiology, but from our part this is still a matter of further research.

As our test was to see the effect of latency on playing accuracy we did not examine our results from the perspective of Fitt's law [23] or Meyer's law [24]. However, it might be interesting to fit the pitch matching results to Mayer's law, as it is a model for target reaching movement that consists of several sub-movements. We could then create a model for each latency and maybe come up with a latency dependent function for the model parameters. Our intention is to analyse the data further by trying to fit a second order control system model to the human pitch-matching behavior.

6. ACKNOWLEDGEMENTS

This research was supported by Pythagoras Graduate School funded by the Finnish Ministry of Education and the Academy of Finland and by an EU IST program (IST-2001-33059). The authors would also like to thank professor Tapio Takala for ideas, help and guidance.

7. REFERENCES

[1] ALMA project home page: (visited April 2004) `http://ftp-dsp.elet.polimi.it/alma/`

[2] J. Paradiso, "Electronic Music Interfaces: New Ways to Play," *IEEE Spectrum,* vol. 34, no. 12, pp. 18–30. Later expanded as an online article (1998) (visited April 2003): `http://web.media.mit.edu/~joep/SpectrumWeb/SpectrumX.html`

[3] M. Wanderley and M. Battier, Eds. *Trends in Gestural Control of Music.* Ircam - Centre Pompidou, 2000.

[4] I. Choi, "A Manifold Interface for Kinesthetic Notation in High-Dimensional Systems," *Trends in Gestural Control of Music*. Ircam - Centre Pompidou, 2000.

[5] A. Mulder, *Design of Virtual Three-Dimensional Instruments for Sound Control.* Ph.D. Thesis, Simon Fraser Univ., 1998.

[6] A. Hunt, M. Wanderley, M. Paradis, "The Importance of Parameter Mapping in Electronic Instrument Design," *Proc. of Conf. on New Instruments for Musical Expression (NIME-02)*, 2002.

[7] A. Hunt, M. Wanderley, R. Kirk, "Towards a Model for Instrumental Mapping in Expert Musical Interaction," *Proc. of Int. Computer Music Conf.*, 2000.

[8] R. Vertegaal, B. Eaglestone, "Comparison of Input Devices in an ISEE Direct Timbre Manipulation Task," *Interacting with Computers,* vol. 8, no. 1, pp. 113–130, 1996.

[9] M. S. O'Modhrain, *Playing by Feel: Incorporating Haptic Feedback Into Computer-Based Musical Instruments,* Ph.D. Thesis, University of Stanford, 2000.

[10] C. Ware, R. Balakrishnan, "Reaching for Objects in VR Displays: Lag and Frame Rate," *ACM Trans. on Computer-Human Interaction,* vol. 1, no. 4, pp. 331–356, 1994.

[11] I. MacKenzie, C. Ware, "Lag as a Determinant of Human Performance in Interactive Systems," *Proc. of Conf. on Human Factors in Computing Systems,* pp. 488–493, 1993.

[12] B. Watson, N. Walker, B. Ribarsky, V. Spaulding, "Managing Temporal Detail in Virtual Environments: Relating System Responsiveness to Feedback," *Proc. of ACM CHI Conf. on Human Factors in Computing Systems*, 1999.

[13] B. Watson, N. Walker, B. Ribarsky, V. Spaulding, "The Effects of Variation of System Responsiveness on User Performance in Virtual Environments," *Human Factors,* vol. 40, no. 3, pp. 403–414, 1998.

[14] S. Card, T. Moran, A. Newell, *The Psychology of Human-Computer Interaction.* Lawrence Erlbaum Associates, 1983.

[15] S. Dahl, R. Bresin, "Is the Player More Influenced by the Auditory Than the Tactile Feedback from the Instrument?" *Proc. of Conf. on Digital Audio Effects*, 2001.

[16] S. A. Finney, "Auditory Feedback and Musical Keyboard Performance," *Music Perception,* vol. 15, no. 2, pp. 153–174, 1997.

[17] A. Freed, A. Chaudhary, B. Davila, "Operating Systems Latency Measurement and Analysis for Sound Synthesis and Processing Applications," *Proc. of Int. Computer Music Conf.*, 1997.

[18] J. Wright, E. Brandt, "System-Level MIDI Performance Testing," *Proc. of Int. Computer Music Conf.*, 2001.

[19] A. A. Sawchuk, E. Chew, R. Zimmermann, C. Papadopoulos, C. Kyriakakis, "From Remote Media Immersion to Distributed Immersive Performance," *Proc. of ACM SIGMM Workshop on Experiential Telepresence,* 2003.

[20] T. Mäki-Patola, P. Hämäläinen, "Latency Tolerance for Gesture Controlled Continuous Sound Instrument without Tactile Feedback," *Proc. Int. Computer Music Conf.* 2004.

[21] K. Enkelaar, A website with information about the Theremin used for the test: (Visited 1.3.2004) `http://www.our.net.au/~cytronic/theremin/INDEX.HTM`

[22] Mustajuuri software homepage (visited 14.4.2004) `http://www.tml.hut.fi/~tilmonen/mustajuuri/index.html`

[23] I. S. MacKenzie, "Movement time prediction in human-computer interfaces," *Proc. of Graphics Interface '92,* pp. 140–150, 1992.

[24] D. A. Rosenbaum, *Human Motor Control.* San Diego, CA: Academic Press, 1991.