IMPLEMENTATION OF INTERACTIVE TOOLS FOR INVESTIGATING FUNDAMENTAL FREQUENCY RESPONSE OF VOICED SOUNDS TO AUDITORY STIMULATION

Hideki Kawahara*, Toshie Matsui†, Kohei Yatabe‡, Ken-Ichi Sakakibara§, Minoru Tsuzaki¶, Masanori Morise**, and Toshio Irino*

* Wakayama University, Wakayama, Japan E-mail: {kawahara,irino}@wakayama-u.ac.jp
† Toyohashi University of Technology, Aichi, Japan E-mail: tmatsui@cs.tut.ac.jp
‡ Waseda University, Tokyo, Japan E-mail: k.yatabe@asagi.waseda.jp
§ Health Science University of Hokkaido, Japan E-mail: kis@hoku-iryo-u.ac.jp
¶ Kyoto City University of Arts, Kyoto Japan E-mail: minoru.tsuzaki@kcua.ac.jp
** Meiji University, Tokyo, Japan E-mail: mmorise@meiji.ac.jp

***Abstract*—We introduced a measurement procedure for the involuntary response of voice fundamental-frequency to frequency modulated auditory stimulation. This involuntary response plays an essential role in voice fundamental frequency control while less investigated due to technical difficulties. This article introduces an interactive and real-time tool for investigating this response and supporting tools adopting our new measurement method. The method enables simultaneous measurement of multiple system properties based on a novel set of extended time-stretched pulses combined with orthogonalization. We made MATLAB implementation of these tools available as an open-source repository. This article also provides the detailed measurement procedure using the interactive tool followed by offline measurement tools for conducting subjective experiments and statistical analyses. It also provides technical descriptions of constituent signal processing subsystems as appendices. This application serves as an example for adopting our method to biological system analysis.**

I. Introduction

Without feedback regulation, the fundamental frequency (f_o[1]) of sustained vowels cannot keep constant value [2]. Auditory feedback of speakers' voices plays an essential role in this regulation [3]. This feedback regulation consists of voluntary, intentional control, and involuntary automatic control functions. We recently found that the voice f_o systematically responds to auditory stimulation presented while voicing a sustained vowel keeping pitch constant. Based on this finding, we proposed a procedure [4] for investigating this involuntary control behavior by adopting a new simultaneous measurement method using extended time-stretched pulses [5], [6].

The purpose of this article is complementary to our article, which focused on theoretical aspects of the method for investigating the response to auditory stimulation. This article presents detailed procedures of typical experiments, and settings for other researchers can replicate our tests that measure involuntary voice f_o response to auditory stimulation. We believe our procedure provides an objective method for investigating human pitch perception mechanisms, especially for the fine temporal structure of the auditory stimulation. This involuntary response also provides a non-invasive method to insert perturbations into the voice production process.

[1]We use symbol f_o to represent the fundamental frequency adopting the discussion in the forum article [1].

II. Background

This section overlaps significantly with the background section of our article [4] accepted for Interspeech2021. We need this overlap to introduce our motivation and the experiment's goal, which our designed tool conducts.

Without proper regulation, we are not able to keep the fundamental frequency of the voice (for example, sustained vowels) constant [2]. Auditory feedback plays an essential role in this regulation [3], [7], [8]. Vibrato, which makes singing voice attractive, also involves auditory feedback in production [9], [10]. Despite decades of research on voice fundamental frequency control mechanisms, it still is a hot topic [11]–[14]. Note that the target of the regulation is not the f_o value. The target is the perceived pitch and is a psychological attribute, [15]. For periodic signals, f_o value is the perceived pitch's physical correlate. In other words, we can observe the perceptual attribute, pitch, directly using the f_o value of the produced voice. The regulation of voice pitch consists of voluntary and involuntary control [16], [17]. The shifted pitch paradigm [18] used in these studies has difficulty investigating this involuntary response.

The first author proposed to use a pseudo-random signal [19] to perturb the f_o of the fed-back voice. It enabled to make the test signal unpredictable and to derive the impulse response of the auditory-to-voice f_o chain [20], [21]. This unpredictability enabled the measurement of involuntary response to pitch perturbation. However, it was difficult for others to replicate the test because it required a complex combination of hardware and software tools. The procedure also consisted of several drawbacks due to available technology in the 1990s. For example, we measured the response to pitch perturbation using the maximum length sequence (MLS) [19]. Selection of MLS

among other TSP signals [22]–[26] was inevitable to make the test signal unpredictable. However, MLS has difficulty in measuring systems with non-linearity [24], [25]. Conventional pitch extractors are the other source of the problem. They introduced non-linear and unpredictable distortions in the extracted f_o trajectories.

We succeeded in making test signals which are unpredictable and do not have MLS's difficulty. Our new system analysis method uses a new extended TSP called CAPRICEP (Cascaded All-Pass filters with RandomIzed CEnter frequencies and Phase Polarities) [6]. We used CAPRICEP and developed an auditory-to-speech chain analysis system by adopting its simultaneous measurement of linear, non-linear, and random responses [5]. We developed an instantaneous frequency-based f_o analysis method instead of using conventional pitch extractors and removed the distortions [4] mentioned above. Combining these analysis methods and substantially advanced computational power removed all the difficulties in measuring the auditory-to-speech chain response. Consequently, they resulted in an easy-to-use tool for conducting experiments [4].

We speculate that this involuntary response measurement introduces a new strategy in pitch perception research. Psychoacoustics and physiology are two prominent approachs to pitch perception [15], [27]–[29]. The tool provides access to the internal representation of "pitch" skipping cognitive and perceptual processes involved in psychoacoustic experiments. The experiments using resolved and unresolved harmonic components and with and without fundamental component is an essential step. The objective response analysis using test stimuli having different fine temporal structures, such as sine, cosine, and alternating phase setting of harmonic components [30] is a promising direction. Including the low peak-factor signals [31] in this experiment is an informative extension.

Our article [4] focused on theoretical aspects of the involuntary response measurement method. The descriptions of the use and procedure of testing such responses were not enough for the readers to replicate the results. We provide missing details of the tool and the test procedure in this article. The tool is open-sourced and available from the first author's GitHub repository [32]. We hope the combination of these makes the involuntary response measurement accessible to researchers.

III. Experiment Descriptions and Procedures

The goal of the experiment is to measure the involuntary f_o response to the auditory stimulation with f_o modulation. The modulation is random to make it unpredictable for measuring the involuntary response. We prepared different types of test signals for investigating pitch perception mechanisms.

Figure 1 shows a schematic diagram of the experiment. The subjects' task is to produce a sustained vowel keeping pitch constant while listening to a test sound. The presentation of the test sound has two modes, and one mode uses a headphone the other mode uses a loudspeaker. We developed an interactive and real-time tool for experimenting.

A. Subjects' task

We prepared two tasks, the same pitch task and a different pitch task. 1) Same pitch task: We instructed the participant

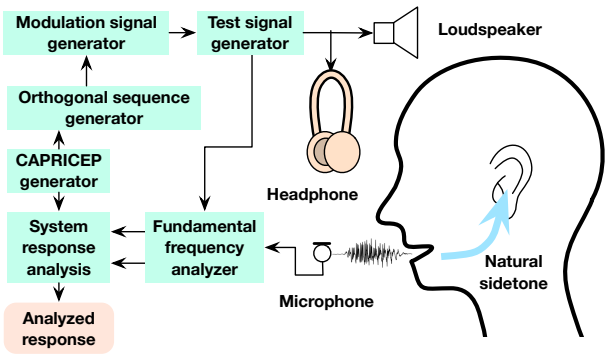

Fig. 1. Schematic diagram of experiments.

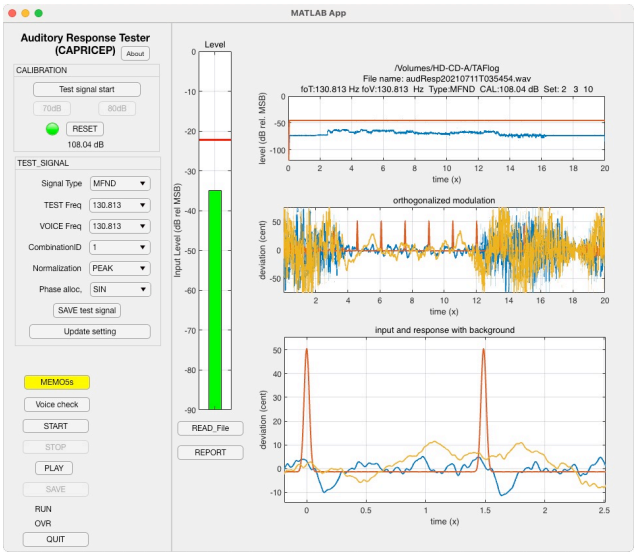

Fig. 2. GUI of the interactive test tool. The left panel is for control, and the right panel displays the analysis results and inspects saved test records.

"Please start voicing about one second after the test tone starts. Please use vowel /X/ keeping pitch constant. Please use the same pitch of the test sound." 2) Different pitch task: We asked the participant to practice voicing using a target sound with a different pitch than the test sound. Then we instructed the participant "Please start voicing about one second after the test sound starts. Please use vowel /X/ keeping pitch constant. Please use the pitch you practiced. It is different from the test sound."

B. Test signal and target signal

Figure 2 shows the GUI of the tool which controls the experiment and analyses the result. The left panel is for control, and the right panel is for displaying the analysis results.

1) Signal type: The left-center sub-panel controls the test signal settings. The top pulldown menu determines the types of the test signal. We prepared four types of test signals.

1) The first is a frequency modulated sinusoid designated as "SINE" in the GUI. 2) The second is a complex sound (sum of harmonically related frequency-modulated sinusoid) with the fundamental component and following 19 components, "SINES." 3) The third is a complex sound without the fundamental component, in the other word missing-fundamental sound, "MFND." 4) The fourth is a missing-fundamental sound consisting of only higher components, "MFNDH." The following two pulldown menus define the f_o of the test signal and the target sound.

2) Signal attributes: The following "CombinationID" pulldown menu defines the combination of constituent three unit-CAPRICEPs. We selected ten different unit-CAPRICEPs and used twenty sets from 720 possible combinations.

The following "Normalization" pulldown menu defines the level adjustment criteria. We used four criteria. 1) Peak level normalization: This sets the maximum instantaneous amplitude of the signal to 0.8 (1 is full scale), designated as "PEAK." 2) RMS level normalization: This sets the RMS (Root Mean Squared) value of the signal -26 dB to the full scale, "TOTAL_RMS." 3) Component level normalization: This sets the amplitude of the fundamental component to -30 dB to the full scale, "COMPONENT."

The following "Phase alloc." pulldown menu defines the phase relation between components. For multiple component signals, we prepared four options of their phase relations, a sine phase "SIN," a cosine phase "COS," an alternating phase (sine and cosine for every other component) "ALT," and the Schröder phase "SCH."

3) Frequency modulation: We used a mixture of orthogonalized sequences made from extended time-stretched pulses (CAPRICEP: Cascaded All-Pass filters with RandomIzed CEnter frequencies and Phase polarity), processed by a pink noise shaper, for modulating the f_o [4].

4) Definition of settings and saving test signals: The following two buttons provide means to save the generates test signals and updating menu items and test signal details. The "Save test signal" button saves the generated test signal using the wave format with metadata representing test signal attributes. The "Update setting" button updates menu items and the frequency modulation depth of the test signal by reading an experiment condition definition file.

C. Procedure

This section describes a few examples of the procedures consisting of test sessions. It starts with the preparation of the test environment and calibration of the input system.

1) Test environment: We implemented the GUI tool and other analysis tools using MATLAB with Signal Processing Toolbox and Audio Toolbox. We used appdesigner for developing the GUI tool. Table I shows a typical environment for conducting experiments described in this article.

2) Start up procedure: When the tool starts, it asks the location of the storage. Then, it asks to select the audio interface (combined with the driver), which can perform simultaneous input and output[2].

TABLE I
TYPICAL ENVIRONMENT REQUIRED FOR CONDUCTING THE TESTS DESCRIBED IN THIS ARTICLE.

Equipment	Description
Computer system	macOS , Windows 10
Software (Toolbox)	MATLAB Signal Processing and Audio Toolbox
Audio Interface	441000 Hz sampling and 16 bit or more
Microphone	Omni directional or cardioid pattern
Headphone	Circumaural and/or noise cancelling
Loudspeaker	Usable from 100 Hz to 10000 Hz
Sound level meter	IEC Class-2 or better

3) Calibration: For making the acquired voiced sounds, it is better to adopt the recommendations [33], [34]. It is essential to calibrate the sensitivity of the acquisition system to calculate the sound pressure level of the produced voice. The top left sub-panel is for this calibration.

First, set the sensitivity of the audio interface to the highest sensitivity while preventing overloading. Ask the participant to produce the loudest voice for this adjustment. Then, click the "Test signal start" button to start playback the pink noise. Adjust the audio interface's output (and the amplifier) to make the sound pressure level (measured using A-weighting) at the microphone position 70 dB or 80 dB. When the level is appropriate and settled, click the relevant (70 dB or 80 dB) button. The green light turns on, indicating that the system is calibrated. The following text shows the calibration information (necessary gain to convert the acquired value to the sound pressure level).

After calibration, do not change the sensitivity of the input system. When adjustment is unavoidable, use the "Reset" button to re-calibrate.

4) Pre-test vocalization check: Before starting a test, the experimenter may use the "Voice Check" button to let the participant try voicing at the target pitch. The click of the button starts the target sound, which is the same type as the test signal without frequency modulation.

The experimenter uses this pitch monitor to instruct the participant how to adjust voice f_o to the target pitch. Note that participants without some musical training sometimes cannot understand instructions such as "please make the pitch higher" and "please lower the pitch."

5) Test vocalization: After completing the voice check, the test starts by clinking the "Start" button. As mentioned in the task description, the experimenter gives one of the following instructions to the participant.

1) Same pitch task: We instructed the participant "Please start voicing about one second after the test tone starts. Please use vowel /X/, keeping pitch constant. Please use the same pitch of the test sound." 2) Different pitch task: We asked the participant to practice voicing using a target sound with a different pitch than the test sound. Then we instructed the participant "Please start voicing about one second after the

[2]For macOS, it also asks to select the input audio interface and the output audio interface. The experimenter needs to select the same hardware. This additional procedure is a walk-around for the performance issue of macOS real-time audio processing.

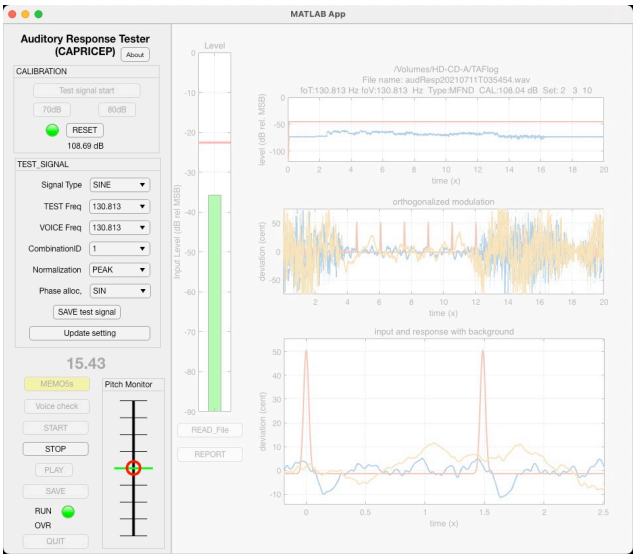

Fig. 3. Snapshot while checking voice pitch control. The pitch indicator appears at the bottom part of the left panel. The green horizontal line indicates the target pitch location. The red circle represents the f_o of the acquired signal in real-time. The interval of the horizontal lines is one semi-tone.

test sound starts. Please use vowel /X/, keeping pitch constant. Please use the pitch you practiced. It is different from the test sound."

We usually use /a/ for the vowel /X/. The test signal duration is 20 seconds. It is challenging to sustain voicing this long. It is usable when the length of voicing exceeds 10 seconds. Please record the distance between the participant's mouth and the microphone to make the acquired data re-usable [33].

While the test signal is on, the timer displays the time from the start, just under the test signal sub-panel. After completion of the test signal, the "PLAY" button and the "SAVE" button go to active. The experimenter can check the recording by clicking the "PLAY" button for glitches and trouble in the recording. If everything is relevant, clicking the "SAVE" button saves the acquired data using a time-stamp-based unique name.

6) Analysis and logging: When the "SAVE" button saves the acquired data, the analysis procedure starts using the saved data. The right panel of the GUI displays the analysis results using three graphs. Note that the experimenter cannot see the analysis results before saving the acquired data. This experiment pipeline design eliminates the chance of cherry-picking in the data acquisition.

The top two graphs are for troubleshooting afterward. The bottom graph shows the response to the auditory stimulation. The horizontal axis represents the time from the maximum stimulation. The vertical axis represents the magnitude of the stimulation and the response. The unit is musical cent. The red line shows the stimulation, and the blue line shows the response. The yellow line shows the random and time-varying response a source of errors in the test. This yellow line is also for troubleshooting afterward.

The tool logs the experimenter's actions with time stamps. The "MEMO5s" button records a voice memo for five seconds and saves the recording using a time-stamp-based unique name. It is an excellent practice to take photos while conducting experiments using smartphones. The timers of smartphones and computers share the synchronized network time using NTP (Network Time Protocol) [35]. The synchronization accuracy is enough to align logged events, voice memos, and smartphone photos.

7) Supporting tools: The GUI tool calls MATLAB functions to generate test and target sounds to analyze the test results. These functions provide offline tools for detailed analyses and generating reports. We developed an acoustic measurement tool [36] by adopting CAPRICEP [6] to the simultaneous multi-attributes measurement method [5]. The tool assesses the recording conditions and sound stimuli presentation referring to the recommendations [33], [34].

IV. Design and Implementation

We designed the tool for assisting the exploratory stage of investigations where flexibility is more important than precision. It is also crucial to record the history of trials with settings and data and avoid retrospective editing. Because this tool encourages trial and error in the exploratory stage, we do not introduce strict security mechanisms. Instead, we time-stamped crucial actions and data. As mentioned before, saving data before displaying analysis results helps prevent (hopefully unintentional) cherry-picking.

Figure 4 shows functional structure of the interactive tool. It shows the relations between GUI parts and functions, data, files, and displays. We placed technical details in appendices because they may disturb readers to acquire appropriate mental models. In this section, we focus on providing a general understanding.

A. Vocal response measurement

Gray rings represent real-time processing loops. The body of the experiment uses the **Response test loop** that is in the bottom center. The "SAVE" button starts this loop and playback data from **Test signal data** and simultaneously acquiring the participant's voice. This loop updates only the time display while the loop is running. This minimum update is to prevent data loss caused by glitches in the real-time processing.

The **Test recording data** holds the acquired data (L-channel: voice, and R-channel: loop-back signal). The **Playback loop** is to check recording errors, if any. The "SAVE" button issues a command to save the acquired data to a uniquely-named file. The command also starts the analysis of the saved file.

The **Response analyzer** is a MATLAB m-function. The experimenter can use the function for offline processing programming.

B. Voice check

The essential secondary loop is the **Voice check loop**. It playback the data in the **Target signal data** and simultaneously acquires the participant's voice. It calculates f_o in real-time and updates the pitch display. It also updates the time display. Please refer to technical details in designing the real-time pitch information display.

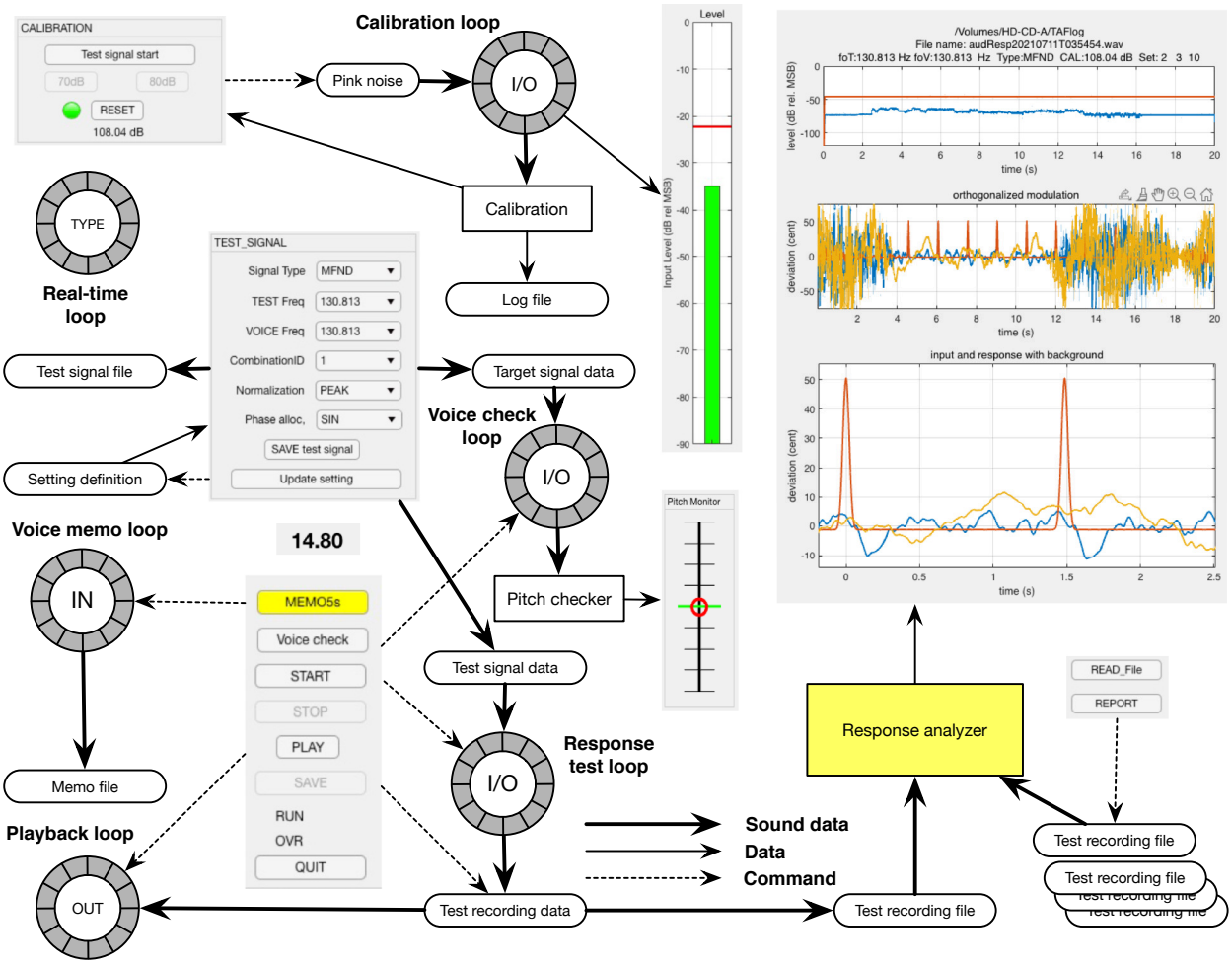

Fig. 4. Functional diagram of the interactive tool.

C. Calibration

The **Calibration loop** also uses simultaneous playback and recording. It updates the level display that displays the RMS value and the peak value. It also updates the time display. The vertical axis represents the level in terms of the full scale of the input signal.

D. Voice memo

The voice memo uses a real-time input loop, the **Voice memo loop**. It updates the time display because it only records five seconds.

V. Conclusions

We introduced a practical procedure for measuring the involuntary response of voice f_o to auditory stimulation with frequency-modulated f_o. The procedure uses an interactive and real-time tool designed for investigating relevant settings of substantial experiments. The subjects' task is to produce a sustained vowel keeping pitch constant while listening to a test sound. The presentation of the test sound has two modes, and one mode uses a headphone the other mode uses a loudspeaker. Each test session lasts about 20 seconds, and the analysis result shows up in several seconds. We designed and implemented the tool to facilitate the exploratory phase of investigations. This article focuses on providing an appropriate mental model of the tool's behavior to experimenters. We provided technical details in appendices. The tools described in this article and related materials are accessible in the first author's GitHub repository [32].

Acknowledgment

This work was supported by JSPS (Japan Society for the Promotion of Science) Grants-in-Aid for Scientific Research Grant Numbers JP18K00147, JP18K10708, JP19K21618, and JP21H04900. We also thank Liao Jiahui, a master student of Toyohashi University of Technology for comments on defects and suggestions on improvement of the tools.

REFERENCES

[1] I. R. Titze, R. J. Baken, K. W. Bozeman, S. Granqvist, N. Henrich, C. T. Herbst, D. M. Howard, E. J. Hunter, D. Kaelin, R. D. Kent, J. Kreiman, M. Kob, A. Löfqvist, S. McCoy, D. G. Miller, H. Noé, R. C. Scherer, J. R. Smith, B. H. Story, J. G. Švec, S. Ternström, and J. Wolfe, "Toward a consensus on symbolic notation of harmonics, resonances, and formants in vocalization," *J. Acoust. Soc. Am.*, vol. 137, no. 5, pp. 3005–3007, 2015.

[2] I. R. Titze, *Principles of Voice Production*. Englewood Cliffs, New Jersey: Prentice-Hall, Inc., 1994.

[3] J. A. Tourville, K. J. Reilly, and F. H. Guenther, "Neural mechanisms underlying auditory feedback control of speech," *NeuroImage*, vol. 39, no. 3, pp. 1429 – 1443, 2008.

[4] H. Kawahara, T. Matsui, K. Yatabe, K.-I. Sakakibara, M. Tsuzaki, M. Morise, and T. Irino, "Mixture of orthogonal sequences made from extended time-stretched pulses enables measurement of involuntary voice fundamental frequency response to pitch perturbation," *arXiv preprint arXiv:2104.01444*, 2021, (Accepted: Interspeech2021).

[5] H. Kawahara, K. I. Sakakibara, M. Mizumachi, M. Morise, and H. Banno, "Simultaneous measurement of time-invariant linear and nonlinear, and random and extra responses using frequency domain variant of velvet noise," in *Asia-Pac. Signal Inf. Process. Assoc. Annu. Summit Conf. (APSIPA ASC)*, 2020, pp. 174–183.

[6] H. Kawahara and K. Yatabe, "Cascaded all-pass filters with randomized center frequencies and phase polarity for acoustic and speech measurement and data augmentation," *Proc. ICASSP2021*, pp. 306–310, 2021.

[7] J. A. Jones and D. Keough, "Auditory-motor mapping for pitch control in singers and nonsingers," *Exp. Brain Res.*, vol. 190, no. 3, pp. 279–287, 2008.

[8] E. F. Chang, C. A. Niziolek, R. T. Knight, S. S. Nagarajan, and J. F. Houde, "Human cortical sensorimotor network underlying feedback control of vocal pitch," *Proc. Natl. Acad. Sci. U.S.A.*, vol. 110, no. 7, pp. 2653–2658, 2013.

[9] C. Leydon, J. J. Bauer, and C. R. Larson, "The role of auditory feedback in sustaining vocal vibrato," *J. Acoust. Soc. Am.*, vol. 114, no. 3, pp. 1575–1581, 2003.

[10] I. R. Titze, B. Story, M. Smith, and R. Long, "A reflex resonance model of vocal vibrato," *J. Acoust. Soc. Am.*, vol. 111, no. 5, pp. 2272–2282, 2002.

[11] C. R. Larson and D. A. Robin, "Sensory processing: Advances in understanding structure and function of pitch-shifted auditory feedback in voice control," *AIMS Neurosci.*, vol. 3, no. 1, pp. 22–39, 2016.

[12] R. Behroozmand, K. Johari, K. Bridwell, C. Hayden, D. Fahey, and D.-B. Den Ouden, "Modulation of vocal pitch control through high-definition transcranial direct current stimulation of the left ventral motor cortex," *Exp. Brain Res.*, vol. 238, pp. 1525–1535, 2020.

[13] E. S. H. Murray and C. E. Stepp, "Relationships between vocal pitch perception and production: A developmental perspective," *Sci. Rep.*, vol. 10, no. 1, pp. 1–10, 2020.

[14] D. Peng, Q. Lin, Y. Chang, J. A. Jones, G. Jia, X. Chen, P. Liu, and H. Liu, "A causal role of the cerebellum in auditory feedback control of vocal production," *Cerebellum*, pp. 1–12, 2021.

[15] B. C. J. Moore, *An introduction to the psychology of hearing*, 6th ed. Brill Academic Pub., 2013.

[16] T. C. Hain, T. A. Burnett, S. Kiran, C. R. Larson, S. Singh, and M. K. Kenney, "Instructing subjects to make a voluntary response reveals the presence of two components to the audio-vocal reflex," *Exp. Brain Res.*, vol. 130, no. 2, pp. 133–141, 2000.

[17] J. M. Zarate, S. Wood, and R. J. Zatorre, "Neural networks involved in voluntary and involuntary vocal pitch regulation in experienced singers," *Neuropsychologia*, vol. 48, no. 2, pp. 607–618, 2010.

[18] T. A. Burnett, J. E. Senner, and C. R. Larson, "Voice F0 responses to pitch-shifted auditory feedback: a preliminary study," *J. Voice*, vol. 11, no. 2, pp. 202–211, 1997.

[19] M. R. Schroeder, "Integrated-impulse method measuring sound decay without using impulses," *J. Acoust. Soc. Am.*, vol. 66, no. 2, pp. 497–500, 1979.

[20] H. Kawahara, "Interactions between speech production and perception under auditory feedback perturbations on fundamental frequencies," *J. Acoust. Soc. Jpn. (E)*, vol. 15, no. 3, pp. 201–202, 1994.

[21] H. Kawahara, H. Kato, and J. C. Williams, "Effects of auditory feedback on F0 trajectory generation," in *Proc. Int. Conf. Spok. Lang. Process. (ICSLP)*, vol. 1, 1996, pp. 287–290.

[22] N. Aoshima, "Computer-generated pulse signal applied for sound measurement," *J. Acoust. Soc. Am.*, vol. 69, no. 5, pp. 1484–1488, 1981.

[23] C. Dunn and M. J. Hawksford, "Distortion immunity of MLS-derived impulse response measurements," *J. Audio Eng. Soc.*, vol. 41, no. 5, pp. 314–335, 1993.

[24] A. Farina, "Simultaneous measurement of impulse response and distortion with a swept-sine technique," in *Audio Eng. Soc. Conv. 108*. AES, 2000.

[25] G.-B. Stan, J.-J. Embrechts, and D. Archambeau, "Comparison of different impulse response measurement techniques," *J. Audio Eng. Soc.*, vol. 50, no. 4, pp. 249–262, 2002.

[26] P. Guidorzi, L. Barbaresi, D. D'Orazio, and M. Garai, "Impulse responses measured with MLS or Swept-Sine signals applied to architectural acoustics: an in-depth analysis of the two methods and some case studies of measurements inside theaters," *Energy Procedia*, vol. 78, pp. 1611–1616, 2015.

[27] R. P. Carlyon and T. M. Shackleton, "Comparing the fundamental frequencies of resolved and unresolved harmonics: Evidence for two pitch mechanisms?" *The Journal of the Acoustical Society of America*, vol. 95, no. 6, pp. 3541–3554, 1994. [Online]. Available: https://doi.org/10.1121/1.409971

[28] A. de Cheveigné, "Pitch perception models," in *Pitch: neural coding and perception*, C. J. Plack, A. J. Oxenham, and R. R. Fay, Eds. Springer, 2005, pp. 169–233.

[29] R. F. Lyon, *Human and machine hearing*. Cambridge University Press, 2017.

[30] R. D. Patterson, "A pulse ribbon model of monaural phase perception," *J. Acoust. Soc. Am.*, vol. 82, no. 5, pp. 1560–1586, 1987.

[31] M. Schroeder, "Synthesis of low-peak-factor signals and binary sequences with low autocorrelation," *IEEE Trans. Inf. Theory*, vol. 16, no. 1, pp. 85–89, 1970.

[32] H. Kawahara, "GitHub repository for speech and hearing research/education tools," 2021, (retrieved 20 July 2021). [Online]. Available: https://github.com/HidekiKawahara

[33] R. R. Patel, S. N. Awan, J. Barkmeier-Kraemer, M. Courey, D. Deliyski, T. Eadie, D. Paul, J. G. Švec, and R. Hillman, "Recommended protocols for instrumental assessment of voice: American speech-language-hearing association expert panel to develop a protocol for instrumental assessment of vocal function," *Am. J. Speech-Lang. Pathol.*, vol. 27, no. 3, pp. 887–905, 2018.

[34] J. G. Švec and S. Granqvist, "Guidelines for selecting microphones for human voice production research," *Am. J. Speech-Lang. Pathol.*, vol. 19, no. 4, pp. 356–368, 2010.

[35] D. Mills, "Internet time synchronization: the network time protocol," *IEEE Transactions on Communications*, vol. 39, no. 10, pp. 1482–1493, 1991.

[36] H. Kawahara, T. Matsui, K. Yatabe, K.-I. Sakakibara, M. Tsuzaki, M. Morise, and T. Irino, "Interactive and real-time acoustic measurement tools for speech data acquisition and presentation: Application of an extended member of time stretched pulses," *Proc. Interspeech2021*, 2021, (Accepted: Interspeech2021 Show and Tell).

[37] MathWorks, "Real-Time Audio in MATLAB," (Last visit: 2021-June-21) . [Online]. Available: https://jp.mathworks.com/help/audio/gs/real-time-audio-in-matlab.html

[38] ——, "Audio I/O: Buffering, Latency, and Throughput," (Last visit: 2021-June-21) . [Online]. Available: https://jp.mathworks.com/help/audio/gs/audio-io-buffering-latency-and-throughput.html

[39] ——, "Measure Audio Latency," (Last visit: 2021-June-21) . [Online]. Available: https://jp.mathworks.com/help/audio/ug/measure-audio-latency.html

[40] ——, "Develop Apps Using App Designer," (Last visit: 2021-June-21) . [Online]. Available: https://jp.mathworks.com/help/matlab/app-designer.html?lang=en

APPENDIX

MATLAB IMPLEMENTATION

We implemented these tools using MATLAB and three toolboxes, signal processing, dsp, and audio [37]–[39]. The audio toolbox enables stream processing at an audio sampling rate by providing the system object of MATLAB. The system object interfaces the driver of the audio device and the procedure written in MATLAB.

The audio toolbox provides three types of system objects, `deviceReader`, `deviceWriter`, and `playRec` for input

stream (for example, microphone input) processing, output stream (for example, sound output for loudspeaker), and synchronized input and output stream processing, respectively. Initially, we used `playRec` for implementing all real-time procedures in the tools. The toolbox assigns different device names for different system objects, even though the connected hardware is the same.

A. Real-time loop

The synchronized input and output stream processing has the following structure.

```
playRec = audioPlayerRecorder(<Setting>);
while <condition>
    <User procedure-1: prepare audioToDevice>
    audioFromDevice = playRec(audioToDevice);
    <User procedure-2: Use audioFromDevice>
end
```

The first line assigns the device that is capable of synchronized input and output. The while loop is the body of the real-time process. The object reads and writes repeatedly at a constant speed. Therefore, the total processing time of the user procedures needs smaller than the repetition interval. We made all signal generation procedures operate outside this real-time loop and store the generated signals to shared buffers. The user procedure-1 in the loop reads the data for `audioToDevice` from this shared buffer.

This processing structure makes real-time programming very easy. The system object conceals details of real-time processes from MATLAB programmers.

B. Issues with `App Designer`

`App Designer` is an interactive development environment for designing an app layout and programming its behavior [40]. It makes GUI application design easy. However, interactivity support and graph rendering are not compatible with real-time processing because the former introduces many interruptions, and the latter requires significant computing resources.

We disabled interactivity of each graph using the "`disableDefaultInteractivity`" function. We set "`limitrate`" option to the graph rendering function "`drawnow`." This option allows skipping update rendering when the total processing time exceeds the repetition interval of the real-time loop.

C. Issues in ***Response test loop***

We tested the real-time latency and glitches (overrun and underrun) of macOS and Windows 10. We found that simultaneous input and output using `audioPlayerRecorder` causes underrun infrequently. We decided to run `audioDeviceWriter` and `audioDeviceReader` simultaneously and let the audio interface to synchronize the input and output.

D. Real-time pitch monitor

The buffer length used by all real-time loops of the tool is 1024, and the sampling frequency is 44100 Hz. This buffer length is too short for calculating f_{o} of low-pitched male voices. We implemented a ring buffer that has appropriate buffer length and updated the data in the **Voice check loop**. We calculated two short-time Fourier transforms of two time-windowed segments (one sample shifted segment pair) and calculated f_{o} using the instantaneous frequency.

We applied a first-order IIR low-pass filtering to the raw f_{o} value to calculate the value for the pitch monitor display. This low-pass filtering makes the display move smoothly and is appropriate for instructing participants.

E. Calibration

We also implemented a ring buffer for calcularing the RMS value for smoother level display. The **Calibration loop** updates the ring buffer.