Temporal Synchronization among Interacting Individuals in Human-Robot Ensembles and Frog Choruses

Takeshi MIZUMOTO

Abstract

This thesis aims at establishing a model of temporal synchronization among interacting individuals. Temporal synchronization occurs in many interactions with various features such as the size, the place, the stationary state, and the participants. To investigate a wide range of interactions, we analyze two cases of interaction that have different features: *human-robot ensembles* and *frog choruses*. The features of the human-robot ensembles are of small size (a few co-players), indoor interaction, *in-phase synchronization* in the stationary state, and humans as the participants. Here, in-phase synchronization means that the onset timings of all co-players are the same. The features of the frog choruses are of large size (dozens of frogs), outdoor interaction, *anti-phase synchronization* in the stationary state, and frogs as the participants. Here, anti-phase synchronization means that the call timings of two frogs alternate. This is because the chorus is for mating, i.e., the male frogs call to attract female frogs. Therefore, call overlap should be minimized so that females can localize males.

For the human-robot ensemble, we develop a temporal synchronization model of multiperson ensemble and a human-robot ensemble system using the model. The requirement with the human-robot ensemble is threefold: (1) development of the music playing robot, (2) onset timing prediction of co-players, and (3) estimation of the leadership in an ensemble. To satisfy the first requirement, we develop a Thereminist Robot system that controls a robot to play the theremin. To satisfy the second and third requirements, we construct a state-space model by combining two components: the coupled oscillator model to represent the onset timing interaction and the leadership estimation to represent the dynamics of leadership. We design a novel quantification of the leadership named *leaderness* and develop its estimation method.

For the frog choruses, we develop a sound imaging system and analyze the spatio-temporal structure of the chorus of Japanese tree frogs (*Hyla japonica*). The chorus analysis is especially difficult for small nocturnal animals, e.g., frogs, because dozens of conspecific individuals call at night at the same site. Therefore, human researchers have difficulty on measuring the time and location of each call. Also, the microphone array processing methods show insufficient performance. We solve this problem by developing a sound imaging system that consists of two components: a sound-to-light converting device named *Firefly* and an off-the-shelf video camera.

This thesis consists of eight chapters. In Chapter 1, we show the motivation of the thesis and the technical problems of the two case studies.

In Chapter 2, we review the literature related to the two cases. The reviewed areas include the expression of music robots, human robot ensembles, and the acoustic communication of animals.

In Chapter 3, we develop the Robot Thereminist system. The theremin is suitable for robot playing because it requires no physical contact. The problem is its sensitivity to the surrounding environment. We develop a theremin's pitch model and its model parameter tracking method by using the unscented Kalman filter. Experimental results demonstrate that our method outperforms the extended Kalman filter and block-wise update method.

In Chapter 4, we construct a two-person ensemble model by using the coupled oscillator model. We assign two oscillators to a human and a robot and update the model from the human's onset timings. Experimental results show that our method reduces the onset timing error by 46% on average compared with the conventional method, that uses the extrapolation of the last tempo.

In Chapter 5, we construct a multiperson ensemble model by extending the two-person ensemble model. To realize the coupling strength estimation, we design the *leaderness* and its iterative estimation method. In the experiment, we analyze the multiperson tapping of humans as a simple ensemble. The results show that the leaderness successfully captures the dynamics of the leadership and that the prediction performance is equivalent to that of humans.

In Chapter 6, we develop the *Firefly* and video analysis method to extract the time and location of each call. The device characteristics measurement shows their feasibility for the field experiments. The simulation reveals that the Fireflies should be placed six times denser than the target animals that call simultaneously. From the indoor and field experiments, we observe anti-phase and 1:2 anti-phase synchronizations that validate the coupled oscillator model of the choruses.

In Chapter 7, observations, general discussion, and the remaining work of the thesis are described.

Finally, Chapter 8 concludes the thesis.

Acknowledgments

This work was accomplished at Okuno Laboratory, Graduate School of Informatics, Kyoto University. I express my gratitude to all people who helped me.

Primarily, I am deeply grateful to my adviser Prof. Hiroshi G. Okuno for his supervision. He gave me a lot of new perspectives to my research. He reviewed all my papers and presentations intensively, and gave hundreds of valuable comments. His wide knowledge is not limited to research. He also told me a lot about literature and arts. Fortunately, I had a lot of experience to work at the outside of the lab because he gave many opportunities.

I am grateful to Prof. Tetsuya Ogata at Waseda University. He always gave me valuable advice in many viewpoints from robotics to cognitive science. Discussion with him is not only valuable but also pleasant very much. I am grateful to Dr. Kazunori Komatani at Nagoya University. His logical and minute advice essentially improved my research. I express my gratitude of the thesis committee members, Prof. Tatsuya Kawahara, Prof. Hideaki Sakai, for their valuable comments on my thesis.

I am grateful of people at Honda Research Institute Japan. Mr. Hiroshi Tsujino supervised me on my internship. Even after that, he gave numbers of opportunities to join fascinating projects. Prof. Kazuhiro Nakadai also invited me a lot of projects. I experienced a wide variety of things including development, documentation, giving a lecture, and working at Willow Garage. I am grateful Mr. Keisuke Nakamura for his highly precise advice based on his deep knowledge about control theory. His positive attitude encouraged me to enjoy the work. I thank

Mr. Takami Yoshida at Tokyo Institute of Technology. Since he and I started PhD course at the same time, the discussion with him was helpful and enjoyable.

My thesis cannot be accomplished without two important research partners. With Ms. Angelica Lim at Kyoto University, I studied the human-robot ensembles. She gave bright inspirations and encouraging words through discussions. She is my teacher of English; she kindly revised my English expressions both in writing and speaking. Her outstanding presentation skill is always my role model. With Dr. Ikkyu Aihara at RIKEN, I pursued the quite exciting project of frog choruses. Thanks to him, I had an opportunity of going to islands, paddy fields, and forests for the field work. His passion and sincerity on research is my role model. I also thank Mr. Yoshiaki Bando and Mr. Hiromitsu Awano at Kyoto University, for their enthusiastic help on developing a measurement system and fieldwork.

I deeply thank Okuno Lab. members, who are good colleagues and friends. Dr. Ryu Takeda told me fundamental skills as a researcher; logically thinking, validating the claim, and so forth. Mr. Takuma Otsuka always gave me insightful advice and helped my fieldwork. I thank Dr. Shunichi Yamamoto, Dr. Katsutoshi Itoyama, and Dr. Shun Nishide for their valuable advice. The discussions with Mr. Masaki Katsumaru, Mr. Tatsuhiko Itohara, Mr. Kohei Nagira, Mr. Yasuharu Hirasawa, and the other students improved my research and knowledge. I thank excellent secretaries, Ms. Miki Nishii and Ms. Hiromi Okazaki.

I express my gratitude to members at LAAS-CNRS, France. They kindly helped in many things when I stayed there for five weeks. Especially, I deeply thank Prof. Patrick Danes for his help on entire aspect of my stay from residential to technical problems. I thank Mr. Jérôme Manhès for his technical support, Mr. Sovannara Hak and Mr. Thomas Moulard for their kind support and friendship.

I am grateful to the Japan Society for the Promotion and Science (JSPS) for their financially support as a Fellowship for Young Scientists DC2.

Last but not least, I am truly grateful to my parents, Kunihiko Mizumoto and Yukimi Mizumoto for their support of my long student life.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation

Interaction pursues synchronization. From several hundred years ago, synchronization has been found in many interactions among both living and non-living things [1–3]. For example, the pendulum clock, which is the phenomena first observed by Christiaan Huygens in the 17th century, and coupled Belousov-Zhabotinsky chemical reactions [4] are the synchronizations of the interactions among non-living things. For the interactions of living things, synchronization is also found in many areas, e.g., the frog choruses [5] and light timings of fireflies [6] in biology, the cardiac cells [7] and circadian rhythm synchronization with the environment [8] in physiology, and the timing of speaking and body movements [9] and nodding during conversation [10] in human interaction. In addition, synchronization has been implemented in robotics. For example, in human-robot interactions, Braezeal developed an interaction robot that can synchronize its speaking and head motion timings with humans for natural conversation [11].

The final goal of this thesis is to construct a model of the synchronization among interacting individuals. We focus on the synchronization in the interactions of intellectual individuals, such as humans and animals. This is because they are challenging targets; they have complex internal processes, which have not been mathematically modeled precisely. Through model construction, we analyze the

Table 1.1: Interaction factors of the cases

Factor	Case 1 Human-robot ensembles	Case 2 Frog choruses
Number of participants	a few	dozens
Location	indoor	field
Stationary state	in-phase	anti-phase
Participant	humans	animals (*H. japonica*)

dynamics and structure of synchronization and develop an artificial participant that can synchronize with others.

Synchronization occurs in a wide range of interactions with various factors, e.g., the number of participants, the location of the interaction, the stationary state of the synchronization, and the participants. Possible factors and interaction examples are as follows. The number of participants is typically low for human conversations and dozens for animal choruses. The location of the interaction is indoors, i.e., no environmental noise, for human music ensembles, and is outdoors for wildlife animal communications. The stationary state of the synchronization is *in-phase synchronization*, i.e., the event timings are the same time, for human-robot ensembles, and is the *anti-phase synchronization*, i.e., the event occurs alternately, for the mating calls in frog choruses. The participants are humans or animals.

We select two cases having different factors: (1) human-robot ensembles and (2) the choruses of Japanese tree frogs (*Hyla japonica*). The factors of these cases are summarized in Table 1.1. Here, we discuss the stationary state. The stationary state of human-robot ensembles is in-phase synchronization because the timings of the co-players should be the same in order to synchronize the ensemble. In contrast, that of the frog choruses is anti-phase synchronization. This is because the chorus is the mating call produced by male frogs, i.e., the call is to attract female frogs for reproduction. Therefore, they try to call at *different* times in order to be localized by the females.

The purpose of the first case, the human-robot ensembles, is twofold: to develop a co-player robot that can synchronize with other players and to analyze the human-human ensemble. Realizing the co-player robot has two advantages. First, we can realize entertainment that can be participated in. Music playing robots have been actively developed such as a flutist robot [12], however, we are just an audience in their performances. In contrast, with the co-player robots, we can *join* their performances actively. Second, the robot can facilitate non-linguistic communication among people with cultural and generational barriers. Since an ensemble is a non-linguistic interaction, such people can interact together. The assumptions of the target ensemble are as follows. First, we assume that the number of participants is low in order to simplify the problem. Note that the constructed model has theoretically no limitation on the number of participants. Second, we assume a score-based ensemble; in other words, all co-players have their own score. This is because we concentrate on the timing synchronization by ignoring artistic aspects such as improvisation. Third, we assume that no conductors exist in the ensemble because our purpose is to model self-organized synchronization without any external force.

The purpose of the second case, the frog choruses, is to reveal the spatio-temporal structure of their synchronization. Since acoustic communication is a common medium for animals, synchronization has been found in many species, such as frogs [5] and crickets [13]. Measuring the spatial and temporal information, i.e., when and where they call, is extremely difficult because they call with many conspecific individuals at night. Although a mathematical model of the *H. japonica* chorus was proposed [14], it was validated only in indoor experiments. To solve this problem, we develop a novel measurement device named *Firefly* and a sound imaging system that uses multiple Fireflies. By using a measurement result, we validate the model in the field. Since the sound imaging system is designed as a general-purpose visualization system of the spatio-temporal structure of choruses, it can be used for other species, such as crickets and other kinds of frogs.

The key model in the thesis is *a coupled oscillator model.* The model consists of two components: (1) several oscillators that generate events sequentially and (2) coupling terms that affect event timings. The oscillator generates an event when its phase becomes a multiplier of $2\pi n$ (n is a natural number). This model is suitable for our purpose because we can design the oscillator to represent the interaction participant and the coupling term to represent the mutual influence during the interaction. For the first case, we represent the co-players as the oscillators. For the second case, we represent the frogs as the oscillators.

1.2 Technical Problems and Solutions

This section summarizes the technical problems for each case. We solve three for the human-robot ensemble as described in problems 1-1, 1-2, and 1-3 and two for frog choruses as described in problems 2-1 and 2-2.

1.2.1 Problem 1-1: Dynamic Characteristics of Instrument

A music playing robot needs to track the dynamically changing characteristics of music instruments because a music instrument's internal state changes over time. For example, the temperature of a brass, the wetness of a woodwind, or the tension of a guitar changes during play, which causes the pitch to shift. In this thesis, we use an electronic instrument, the *theremin*, for the robot's instrument. Although the theremin has advantages in that it is suitable for robots, e.g., no physical contact is required, this problem is crucial for the theremin due to its sensitivity to the surrounding environment.

We solve this problem with two steps. First, we construct a theremin parametric model by observing the theremin's sound. The model represents the relationship between the arm position and the theremin's pitch. Second, we develop a method for iteratively updating the model parameters. Since the model is nonlinear, we solved it by using an unscented Kalman filter, a nonlinear version of Kalman filters.

1.2.2 Problem 1-2: Onset Timing Prediction of Co-players

A co-player robot needs to predict the other co-players' onset timings. The reason is that they need to play an onset at the same time with others to achieve in-phase synchronization. The problem is that the robot needs to move its body *before* the onset observation because the robot's motion speed is finite. A naive solution is to extrapolate the last inter-onset interval (IOI) [15]. This method has a large error because it assumes that the robot's playing has no affect on a human. However, the robot and the human are mutually dependent, i.e., the robot's playing typically affects humans.

We solve this problem by constructing an ensemble model that incorporates this effect by using a coupled oscillator model. Consider an ensemble consisting of one human and one robot. We model the whole ensemble with the coupled oscillator model by assigning two oscillators to each participant. The phases of the oscillators are updated by using (1) their speed and (2) the difference from the phase of another oscillator. This models a situation in which the human and the robot have their own tempos and try to synchronize with each other.

1.2.3 Problem 1-3: Leadership Estimation

Leadership estimation is an essential function for co-player robots, especially in a multiperson ensemble. This is because the robot needs to know whom to follow or when to lead. In conventional human-robot ensemble studies, this problem is avoided by assuming that the leader is assigned in advance or only one leader exists at the same time. However, in realistic human-human ensembles, the problem is unavoidable because multiple leaders can exist and change over time.

Our solution is twofold: designing a quantified leadership named *leaderness* and constructing a state space model by using the coupled oscillator model and the leaderness. The design policy of the leaderness is that the co-player who plays the new tempo and keeps it constant is the leader. This is because the leaders are necessary only when the tempo changes in order to lead the co-players to the new stable tempo. In the model, the leaderness is used as the coupling strengths of the oscillators.

1.2.4 Problem 2-1: Chorus Measurement in Noisy Environments

Many microphone array processing methods show insufficient performance in the field because many noise sources exist, such as the sound of wind blowing, the calls of other species, and the calls of target species from other places. When we tried to separate the call of the target frog in a paddy field with an independent component analysis [16], we failed because the signal involved too many noise sources to separate.

We solve the problem by using the sound-to-light converting device named *Firefly*, which has an insensitive microphone and a LED. By distributing dozens of them in the field, we can measure the call timings and locations from the light pattern. This solution is less affected by the noise than the microphone recording because each device captures only nearby sound.

1.2.5 Problem 2-2: Large Area of Interest

The area of interest is large because our target is the frogs' habitat. For example, a paddy field where Japanese tree frogs call is typically larger than 20 × 20 m. To cover such a large area, we need to distribute many microphones. This causes three difficulties: (1) many microphones, long cables, and heavy recording equipment are required, (2) the multichannel recording equipment is expensive because synchronization among microphones is required, and (3) measuring the microphone array configuration is time consuming because a field is rough and dark, unlike the indoor environment.

We solve this problem by using light instead of sound itself. We use the Fireflies to convert the sound into light and an off-the-shelf video camera that captures their lighting pattern. Since the speed of light is enough high to ignore that the time difference between the devices blinking and the video capturing, the inter-device synchronization is not necessary. Therefore, we can avoid the difficulties because (1) the system has no cables, (2) no synchronization equipment is required, and (3) the configuration can be calculated from the light pattern in the video.

1.3 Organization

The organization of the thesis is shown in Figure 1.1. In Chapters 3, 4 and 5, the case study of the human-robot ensemble is described. In Chapter 6, the case study of the frog choruses is described.

In Chapter 2, we review the literature to clarify the unsolved problems. The review includes the following three areas: music robots, human-robot ensembles, and the acoustic communication of animals.

In Chapter 3, we develop a solo-playing function to play an instrument correctly. We describe a robot control system for theremin playing named the *Robot Thereminist* system. First, we briefly introduce the theremin and a parametric model of its pitch characteristics. Then, we develop a parameter update method for keeping the pitch correct in a dynamic environment. In the experiment, we evaluate the precision of pitch control by using both a simulation and a real robot.

In Chapter 4, we build an ensemble model of two-persons by using a coupled oscillator model. First, we introduce two building blocks: (1) a real-time beat tracking system for extracting the beat timings and tempos as the input to the model and (2) a Kuramoto model, which is a widely used coupled oscillator model. Then, we construct an ensemble model for a two-person ensemble. In the evaluation, we compare our method to the extrapolation-based method by using a metronome and a human.

In Chapter 5, we construct a state space model of a multiperson ensemble. The building blocks are twofold: (1) the extended version of the two-person model designed in Chapter 4 for a multiperson ensemble and (2) the *leaderness*, which indicates how a co-player leads the ensemble. We evaluate the prediction accuracy of the model and analyze the dynamics of the leaderness by using an experiment with humans.

In Chapter 6, we develop a sound imaging system for visualizing a spatio-temporal structure of animal choruses. We describe the *Firefly* and data analysis method. We evaluate the system in two ways. First, we measure the device characteristics including sound-pressure-level to light-intensity characteristics and directional sensitivity. Second, we validate the temporal and spatial resolution by using indoor experiments and a simulation. Finally, we investigate the spatio-temporal structure of the Japanese tree frog choruses in the field.

We discuss the contributions of the thesis, general aspects of the case studies, and the future work in Chapter 7. Finally, we conclude this thesis in Chapter 8.

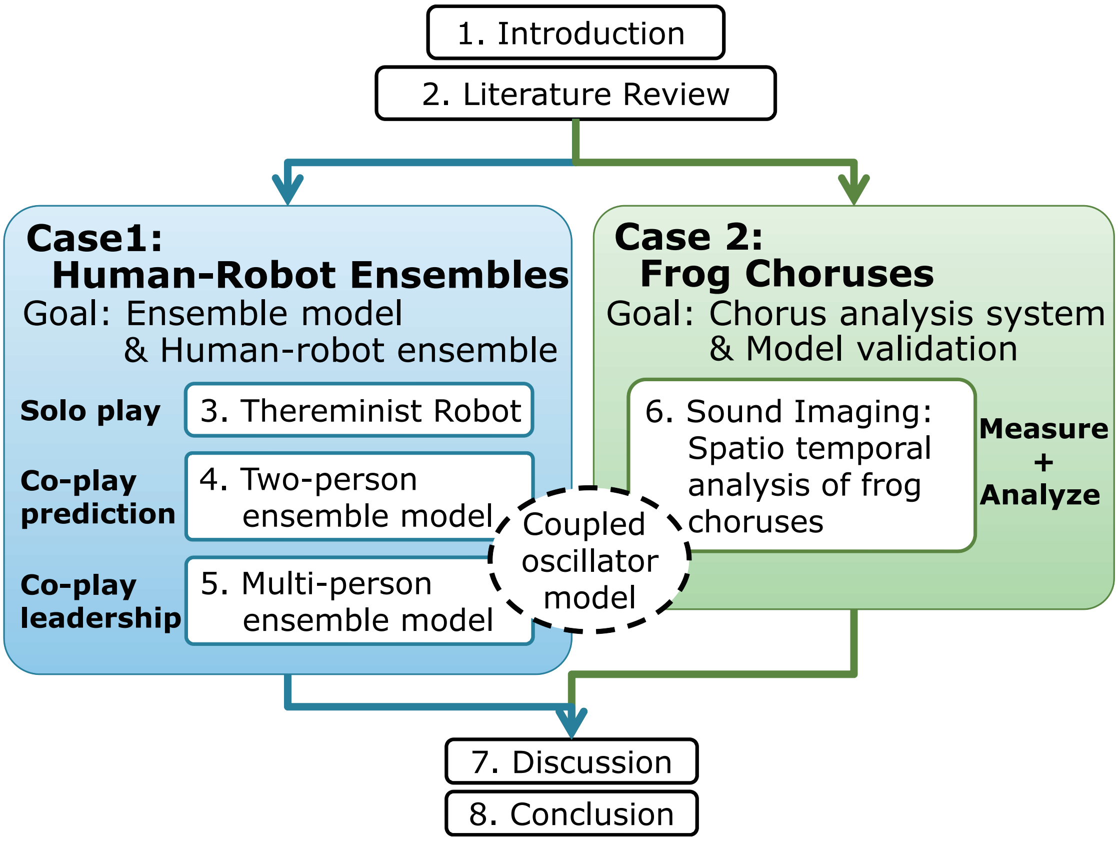


Figure 1.1: Organization of the Thesis

Chapter 2

Literature Review

This chapter reviews the literature related to two cases. Section 2.1 summarizes the music robot studies from three music expressions and the relevant thereminist robots. Section 2.2 reviews the ensemble from psychological studies to human-robot ensemble systems. Section 2.3 describes studies of animal acoustic communication and state-of-the-art communication investigation methods in the field.

2.1 Music Robot

2.1.1 Music Expressions of Robots

We review three main categories of the music expression studies of robots: instrument playing, dancing, and singing. We select the instrument playing because of two reasons; it involves body movement, and we can clearly evaluate the performance comparing with the given score.

Instrument Playing

Many studies reported robots that play the instrument. The advantage of this expression is twofold; (1) playing the instrument requires the body movement of the robot. Therefore, this expression is suitable for robots. (2) We can define a quantitative criterion, for example, a pitch error, compared with the score. The common problem of conventional studies is that the control systems and the robot hardware are indivisibly united. Therefore, porting these systems to another robot requires a lot of effort.

Recent instrument-playing music robots are studied from two motivations: *to develop more sophisticated music robots* and *to develop an ensemble between humans and robots*. The former aims to realize a solo-playing robot that can play like humans, and the latter aims to realize a co-playing robot that can play in synchronization with humans. This section reviews the literature for the former, and the section 2.2 reviews the literature for the latter.

Many instrument-playing music robots have no humanlike shape. Singer summarized some works of automatic music playing machines [17] developed in League of Electronic Musical Urban Robots (LEMUR) by artists and engineers. Kaneko *et al.* developed a mechanical system to play a trombone [18]. Studies of humanlike music playing robots have started from a keyboardist robot named WABOT-2 [19]. Similar to WABOT-2, many studies aim to develop a humanoid robot that plays the instrument. Solis *et al.* developed robots that play the flute [12] and the saxophone [20] that play the given score at the given tempo sophisticatedly. They developed an artificial lip and an artificial lung to imitate the human breathing during the play. Toyota developed the Partner Robot that plays the trumpet and violin [21]. Shibuya *et al.* developed a violin playing robot [22]. Weinberg *et al.* developed a drum playing robot Haile [23], and a marimba playing robot Simon [24]. Kotosaka *et al.* developed a drum-playing robot that changes its tempo according to a human's drumming [25].

Dancing

Dancing is the most popular in the three music expression music of robots because its minimum implementation is the easiest. The "dancing" can be achieved by just moving arms, whereas instrument-playing requires to play a desired pitch at the desired timing at the minimum. Therefore, the typical robot demonstrations with music are dancing whose motion is pre-programmed, and the corresponding music is played simultaneously by humans.

In contrast, some researchers have developed sophisticated dancing robots. Nakaoka *et al.* developed a control method to imitate a human dancing motion [26]. It extracts the key poses from the human dancing motion measured by a motion capture system. They evaluated through the robot motion analysis based on a Laban movement analysis [27]. Yoshii *et al.* and Murata *et al.* developed a function of keep-stepping with an external music on a humanoid robot [28, 29]. They used the beat tracking method [30] for the tempo extraction, and the foot pressure sensor for step timing adjustment. Kosuge *et al.* developed a dancer robot Party Ballroom Dance Robot (PBDR) that can dance with a human by predicting the human's intention from physical interaction [31]. Michalowski *et al.* developed a robot named Keepon which has four degrees of freedom and dances with music [32].

Singing

This expression has the freest range because no physical limitations exist. Although the simplest way is just to play a song from a loudspeaker on the robot, this does not take advantage of the embodiment of the robot.

Some studies reported a singing robot that incorporates embodiment. Mizumoto *et al.* developed a beat counting by listening to music [33]. Murata *et al.* developed a singer robot by adjusting its singing speed with the external music [34]. Kajita *et al.* realized a singing motion generator for controlling the singing face of HRP-4C [35].

2.1.2 Theremin Playing Robots

Theremin is an electronic instrument that can be played without touching it. Its pitch and volume are determined by the distance from the player's arms and the antennae. (Refer to section 3.2 for details.) Using the theremin's feature of the proximity control, some studies used the theremin as a proximity sensor for an interactive system [36] [37]. The problem is that the relationship between the theremin's sound and the player's arm positions is strongly affected by the surrounding environment, e.g., temperature, the player's position, and the number of people near the theremin.

Alford *et al.* created the first theremin playing robot [38]. Its control system uses a look-up table that provides a corresponding arm position for a given note. The table is calibrated by moving the robot's arm to find the best arm position for each note in advance. Therefore, the table must be re-calibrated when the environment changes. This disadvantage makes it difficult to use this method for human-robot ensembles, which has dynamically changing environment. Van der Hurst developed the theremin playing robot using feedback control [39]. The robot controls its arm by comparing the observed and desired pitches. Although this method achieves a precise pitch control, it takes a long time to converge to the desired pitch. Therefore, the method is unsuitable for melody playing.

Mizumoto *et al.* developed a parametric model of the theremin's pitch [40]. The system works in two phases: calibration and performance. In calibration, the robot records the theremin's pitch at some robot arm positions. The key point is that the robot positions have no restrictions. This is because the arm positions and pitches are used the learning data for parameter estimation, unlike [38]. According to [40], the pitch control accuracy saturates when the number of arm positions is twelve. Wu *et al* proposed a pitch model update method [41]. They assumed that the theremin's pitch increase is linear in log-scale, and developed an update method using a linear regression. However, because of the model's simplicity, the pitch control accuracy is less than the parametric model [40].

2.2 Human Robot Ensemble

Not only robot systems, psychological studies of human's music activities are important because to realize a natural interaction for humans. We review the psychological studies of an ensemble first and review the relevant human robot ensemble studies.

2.2.1 Psychological Studies of Ensemble

An ensemble is a multimodal interaction. In addition to audio information, the co-players also interact using visual information such as eye contacts [42] and gestures [43]. Some psychological studies also pointed out the use of visual information in an ensemble. For example, Thompson *et al.* reported that a singer's face influences the audience's judgment of the singer's emotion [44]. Ritchie *et al.* reported a pianist's playing motion has a correlation of the score [45]. Therefore, the robot which has embodiment is an essential component for an ensemble.

Rhythm recognition has been actively studied. A typical task for the experiments is called *tapping task*. In the task, a participant taps a key by listening to the stimulus such as a metronome sound or a sequence of tones. Note that the participant follows the stimulus, i.e., the stimulus is the leader and the participant is the follower. Many psychological models have been proposed; Haken *et al.* constructed a model of a human's bimanual coordination in response to a metronome sound, called *Haken-Kelso-Bunz model* [46]. Large and Jones proposed a model of a human's recognition model of a sequence of beeps [47]. In the experiment, they used a stimulus of a beep sequence whose inter-onset intervals (IOI) is manipulated.

An oscillator model is broadly used to build a model including the models introduced in the above paragraph. The reason is that the oscillation is suitable to represent succeeding events occurred in a constant interval. Many neurophysiological studies suggest the existence of a neural clock, i.e., a timing representation using pulses or oscillators, in our brain (see [48, 49] for detailed reviews). An oscillator is also used for music processing, such as onset prediction [50], robot drumming [51], and beat tracking [52].

Note that these studies investigated a human's *internal* mental process in response to an external signal, not the *interaction* itself. Some studies tackled the modeling of interaction in music. Braasch *et al.* built a model of a free jazz improvisation using a cybernetic model [53]. Pecenka *et al.* studied a duo ensemble of pianists. They found a correlation between a pianist's capability to imagine the music from a score and a sensorimotor synchronization [54]. Keller proposed a cognitive model of human in ensemble [55]. The detail of this model is described in section 5.2 along with the relationship to our multiperson ensemble model.

2.2.2 Ensemble Systems with Humans

We start from human-computer ensembles that have a longer history than human-robot ensembles. Dannenberg's real-time accompaniment system [56] is the first work on human-computer ensembles. We have categorized these studies into two types: (1) a human leads an ensemble and a computer follows it, and (2) humans and computers play an equal role. They started from the first type; Raphael proposed a probabilistic approach [57] and Simon *et al.* proposed a method of code generation based on a hidden-Markov model [58]. In the second type, Goto *et al.* proposed a jazz system whose participants play the instrument by interacting with one another [59]. Hamanaka *et al.* implemented a jazz system of three guitar players that learns the participant's playing style [60]. In these systems, the participants play the same role.

We will now describe the studies on human-robot ensembles. They are categorized into two types: score-based and improvisational. In score-based ensembles, the robot plays a given score with humans. Petersen *et al.* presented an ensemble system with a robotic flutist and a human saxophonist using a score [61]. The robot and the human play melodies alternately, instead of playing simultaneously. As our goal is to achieve synchronization, we need participants to play their instruments at the same time. Otsuka *et al.* developed an ensemble system with the Robot Thereminist and a human drummer [15]. The robot changes its playing speed according to the intervals of the beat in the human's playing. They ignored the perspective of prediction, which is essential for synchronized playing. The prediction of other participants is essential because a robot needs to generate playing motions *on time*. Lim *et al.* developed a synchronized playing system that combines gesture recognition of a flutist with a beat tracking method [62]. Itohara *et al.* developed an audio-visual beat tracking method specialized for a guitarist and an ensemble system using the method [63].

In improvisational ensembles, a robot plays a melody or a rhythm which is not prepared in advance. Weinberg *et al.* proposed an ensemble system with two humans and two robots: robotic drum and marimba players and human drum and keyboard players [64]. They achieved a simultaneous and improvisational performance with multiple-humans and multiple-robots. The robots play the instruments similar melodies or rhythms to the human's playing by transforming them stochastically. In other words, the robots play the similar melody or rhythm to those of humans. Therefore, their approach is insufficient to realize the ensemble in which each player plays different melodies.

2.3 Animal Acoustic Communication

Spatio-temporal structure is an important clue to understand the acoustic communication of calling animals [5, 65]. Such acoustic communication has been well studied for various kinds of animals including frogs [66–70], crickets [65, 71], and bats [72–75]. Animals use the acoustic communications for various purposes, for example, mating, territory maintenance, and localizing preys. Especially for the frogs, the main purposes are twofold; the advertisement calls are for mating, and the aggressive calls are for keeping territories.

Many investigation methods have been proposed for this purpose. The most typical method is manpower. However, human observers are incapable of distinguishing calls in a chorus because it has more than ten frogs calling three to five times per second. According to [76], humans have difficulty to distinguish more than three simultaneous talks. Attaching a logger to the animal is an emerging method to record their activity precisely. This method is used, for example, diving of hawksbill sea turtles (*Eretmochelys imbricata*) [77], tracking of a wild American Crow [78], and Vocalight [79], which is a visualization device for acoustic communication of dolphins. The problem is that the logger affects the animal's behavior especially for small animals. Note that the snout-vent length of our target animal, *H. japonica*, is about 3 - 4 cm.

Microphone array processing is a promising method. For example, sound source localization using arrival-time differences has been used for various animals; for example, bullfrogs (*Rana catesbeiana*) [80], Gulf Coast toads (*Bufo valliceps*), Northern cricket frogs (*Acris crepitans*) [81], marine animals [82], Lek-breeding reed frogs (*Hyperolius marmoratus*) [83], Red-Winged Blackbirds (*Agelaius phoeniceus*) [84], Bowhead whales (*Balaena mysticetus*) [85]. The drawback of the method is that it assumes that only one call exists at the same time. This assumption does not hold for dense choruses such as *H. japonica*. Schwarts estimated frog locations by comparing sound power captured by microphones with a threshold [86]. Because the method selects the microphone closest to the frog, the spatial resolution is limited by the number of microphones. Recently, multiple microphone arrays are used to determine animal locations to investigate social communication of animals [87]. They used a global positioning system to localize each microphone array unit, and cross-correlation methods to localize the sound. The problem of these microphone array approaches is that they assume the temporal sparseness, i.e., the calls must not be overlapped. This assumption does not hold for densely calling animals. Although advanced microphone array processing methods can localize multiple sounds, they have been developed for indoor use. Therefore, the performance is severely degraded for outdoor recordings. For example, blind source separation using independent component analysis [16,88] separates the sound mixture without any prior information but the observed signal. A widely used sound source localization method, MUSIC (MUltiple SIgnal Classification) [89], and a geometrically constrained higher-order decorrelation-based source separation method (GHDSS) [90] are incorporated in the open-source software, HARK [91].

2.4 Summary

In this chapter, we reviewed three areas relevant to the case studies of the thesis; music robots and human robot ensembles are relevant to the human-robot ensemble, and the animal acoustic communication is relevant to the frog choruses analysis.

The unsolved problems are the following: for the music robots, the robot needs to adapt to the dynamic environment to keep playing correctly. For the human robot ensembles, we need to (1) construct a model of the whole ensemble not only the individual behavior, and (2) construct a model of the leader in the ensemble. For the acoustic communication investigation, we need a robust visualization method of the spatio-temporal behavior in their habitat.

Chapter 3

Robot Thereminist

Solo-playing robot is a fundamental component of the human-robot ensemble. In this chapter, we develop a music robot that plays the theremin named *Robot Thereminist*. The key points are the theremin's pitch model and the iterative update of the model parameters.

3.1 Introduction

Robots that play instruments with humans in ensembles are expected to facilitate an intuitive and natural human-robot interaction. Since a music ensemble requires no common linguistic knowledge, it can overcome cultural barriers such as language or generation. From the point of view of entertainment robotics, such robots will provide a participable entertainment in which people can *join* the performance. This crucially differs from existing studies of solo music-playing robots [19,40,92,93] because, in these performances, people are the passive audience, not participants.

The theremin is an electronic instrument that is played without any physical contacts (Figure 3.1). The player can control the theremin's pitch and volume only by moving the both arms; the theremin's pitch increases when the right arm approaches the right vertical antenna. The theremin's volume decreases when the left arm approaches the left horizontal antenna. The detailed description of the theremin is in Section 3.2.

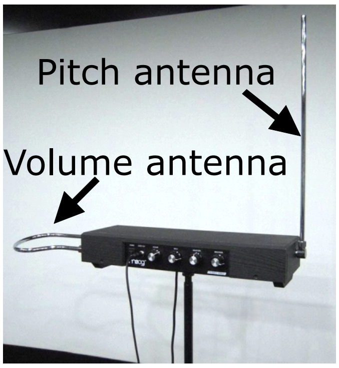


Figure 3.1: Picture of the theremin

The advantage of the theremin is twofold; one is that no physical contacts are required to play. Unlike other instruments, we can play the theremin without touching it. Therefore, the only mechanical requirement for the theremin playing robot is to have two arms. Since this is obviously satisfied by almost all robots, the system can be ported to many robots. In fact, we have ported the Robot Thereminist system on multiple robots [40]. The other is that the theremin generates the continuous pitch. Like a trombone or a violin, the theremin's pitch continuously changes by the arm position. For example, we can play a microtone, glissando, or portamento by the theremin. Therefore, the players can play a rich expression than using the instruments having discrete pitches.

The problem on developing a Robot Thereminist is the theremin's sensitivity to the environment. The surrounding environment strongly affects the theremin's sound, e.g., the temperature of the room, the player's position or the presence of the audience. In other words, even if the player keeps the arm at exactly the same position, the theremin's pitch and volume will change depending on the environment. This is an essential problem especially in ensemble because if a human co-player is close to the theremin, the human's motion interferes to the robot's theremin. Figure 3.2 shows an example; the robot's theremin is affected by the existence of the co-player.

To solve this problem, we construct a parametric model of the theremin and

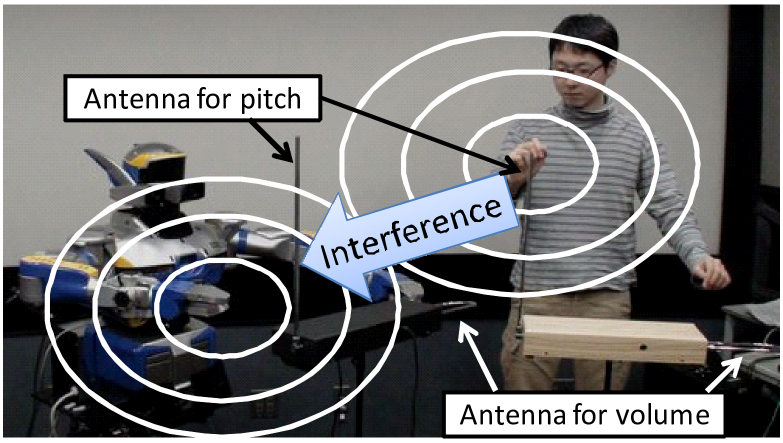


Figure 3.2: Dynamic environment in ensemble

build a Robot Thereminist system of two phases; *calibration* and *performance*. First, we describe the system that estimates the model parameters in the calibration phase, and then plays in the performance phase, assuming that the environment is stable after the calibration phase. Second, we describe the parameter update method during the performance phase because the environment dynamically changes in the ensemble by the co-player's movement. The robot needs to adapt the changing environment to keep playing correctly because the ensemble will become out of harmony without adaptation. Note that this phenomenon is not limited to the theremin, e.g., the temperature of the instrument, the wetness of a wooden instrument, or the tension of a guitar, also causes the pitch shift. The conventional music-playing robots mentioned above assume that such an environmental effect is ignorable, but environmental changes are especially crucial in theremin playing. We empirically found that the environmental capacitance actually changes even if the instrument is left alone in a room.

We present an adaptive Robot Thereminist system that can update the model parameters iteratively. We build a state space model of the model parameters and solve it using an unscented Kalman filter (UKF). The UKF has three advantages [94] compared to the extended Kalman filter (EKF), which is broadly used for nonlinear tracking [95]:

1. The UKF requires no Jacobian of state update and observation functions. Therefore, it works even for the indifferentiable functions.

2. The UKF can track nonlinear systems precisely. This is because the UKF approximates the hidden state and covariance by the second order of Taylor expansion whereas the EKF approximates by the first order .

3. The UKF works with a low number of samples compared to the probabilistic methods such as particle filters [96]. This is because the UKF uses the sample points named *sigma points*. The number of the sigma points is defined by the dimension of the state, and the samples are selected deterministically.

3.2 Theremin Overview

The theremin is one of the oldest electronic musical instruments developed by Léon Theremin [97]. It is a monophonic instrument with continuous musical scale. Therefore, the theremin's production of sound is more similar to that of trombones or violins than pianos or flutes. As shown in Figure 3.1, the theremin has two antennae: vertical one for pitch control and horizontal one for volume control, respectively. The player can control the volume and the pitch only by changing the proximity of the arms and the antennae without touching it. These features are caused by the theremin's mechanism.

The theremin produces sounds using the *beat*, which is a physical phenomenon when two waveforms having slightly different frequencies are multiplied. We firstly describe the pitch control. The theremin has two oscillating circuits for the pitch control. Let f_1 and f_2 be the oscillating frequencies of the oscillators and $f_1 > f_2$. When we multiply these outputs, the result has two frequency components, $f_1 + f_2$ and $f_1 - f_2$, which is the beat. By filtering the signal with a low pass filter with the cut-off frequency of $f_1 - f_2 < c < f_1 + f_2$, we obtain the signal having one component, $f_1 - f_2$. This is the pitch of the theremin. The volume control has the similar mechanism because two oscillators are multiplied. Only difference is that the filtered signal is given to a frequency-to-voltage controller for the gain control of the output signal.

How can we change the beat frequency? The key component is the *virtual capacitors*. We can assume that the capacitance of the oscillating circuit's capacitor is controlled the player's arm position. Since the frequency (f_1) is determined by the values of the parts, the player can change the output frequency by moving the arm. The reason why the player can control the capacitance is as follows: A capacitor consists of three components: two conductive materials and a non-conductive material put between them[1]. The capacitance is mainly determined by two factors: (1) the distance between two conductive materials and (2) the substance of the non-conductive material. In the virtual capacitor, the antenna and the player's arm are the conductive materials, and the air between them is the non-conductive material. Therefore, when the player changes the arm position, the distance of the capacitor changes, then, the pitch and volume of the theremin change.

This concept of the virtual capacitor also explains why the theremin's pitch characteristic is sensitive to the environment. When the temperature of the room or the presence of the audience changes, the substance of the non-conductive material changes corresponding to (2). Therefore, the theremin is sensitive to the surrounding environment.

3.3 Static Pitch Control for Robot Thereminist

First, we describe a parametric model of a theremin's pitch and a static pitch control method. This is the basics of the adaptive pitch control method.

[1] Typically the conductive materials are the metals and the non-conductive material is the ceramic.

3.3.1 Parametric Pitch Model

Let p be the theremin's pitch [Hz], $\boldsymbol{\theta} = (\theta_0, \theta_1, \theta_2, \theta_3)$ be the model parameters, and $x_p \in [0, 1]$ be the abstracted robot's arm position. Here, $x_p = 1$ and $x_p = 0$ mean the closest and farthest arm position to the antenna, respectively. We assume that the robot's arm moves along a line or a curve, and the trajectory monotonically gets closer to the antenna.

The theremin's pitch model M_p represents the theremin's pitch p at given arm position x_p. The formulation is:

$$p = M_p(x_p; \boldsymbol{\theta}) \quad = \quad \frac{\theta_2}{(\theta_0 - x_p)^{\theta_1}} + \theta_3 \tag{3.1}$$

The model is constructed based on the experimental observations; the theremin's pitch increases monotonically and nonlinearly. Because the pitch increase speed depends on the environments, we used these four parameters to represent the variety.

The inverse model, M_p^{-1}, i.e., the function of the arm position that achieves to play the given pitch, is important for playing a score. This can be derived analytically from M_p:

$$x_p = M_p^{-1}(p; \boldsymbol{\theta}) \quad = \quad \theta_0 - \left(\frac{\theta_2}{p - \theta_3}\right)^{1/\theta_1} \tag{3.2}$$

3.3.2 Static Pitch Control

The static pitch control method consists of the two phases: *calibration* and *performance*. In the calibration phase, the model parameter $\hat{\boldsymbol{\theta}}$ is estimated using the $L + 1$ pairs of the arm positions $x_p^{(i)}$ and the observed theremin's pitch $p^{(i)}$ at the position $(i = 0, ..., L)$:

$$x_p^{(i)} \quad = \quad i/L \tag{3.3}$$

$$p^{(i)} \quad = \quad M_p(x_p^{(i)}; \boldsymbol{\theta}_T) \tag{3.4}$$

where $\boldsymbol{\theta}_T$ is the unknown true parameter. Note that Eq. (3.3) denotes the assumption that the arm positions intervals are the same for calibration. Then, the model parameter $\hat{\boldsymbol{\theta}}$ is estimated by minimizing the squared error using the Levenberg-Marquardt method [98]:

$$\hat{\boldsymbol{\theta}} = \underset{\boldsymbol{\theta}}{\operatorname{argmax}} \sum_{i=0}^{L} |p^{(i)} - M_p(x_p^{(i)}; \boldsymbol{\theta})|^2 \tag{3.5}$$

In the performance phase, the trajectory of the robot's arm position $\hat{x}_p(t)$ is calculated from the estimated parameter $\hat{\boldsymbol{\theta}}$ and the given musical score defined as the desired pitch trajectory $q(t)$:

$$\hat{x}_p(t) = M_p^{-1}(q(t); \hat{\boldsymbol{\theta}}) \tag{3.6}$$

The conventional methods [38, 40] assume a stable environment, i.e., the true model parameter $\boldsymbol{\theta}_T$ does not change after the calibration phase. In an actual situation, however, this assumption does not hold because the environment is dynamically affected by (1) gradual environmental changes, such as room temperature or a theremin's internal state, and (2) the co-player's instrument playing motion.

3.4 Adaptive Pitch Control

We present an adaptive pitch control method using a UKF [94] for accurate theremin play under a dynamic environment. In this section, we describe our method as follows: (1) the problem statement, (2) a review of the UKF, and (3) its application to our problem including designs of the state-update and observation functions. Fig. 3.3 depicts the overview of the Robot Thereminist System with the adaptive pitch control.

3.4.1 Problem Statement

The adaptive pitch control problem is summarized below.

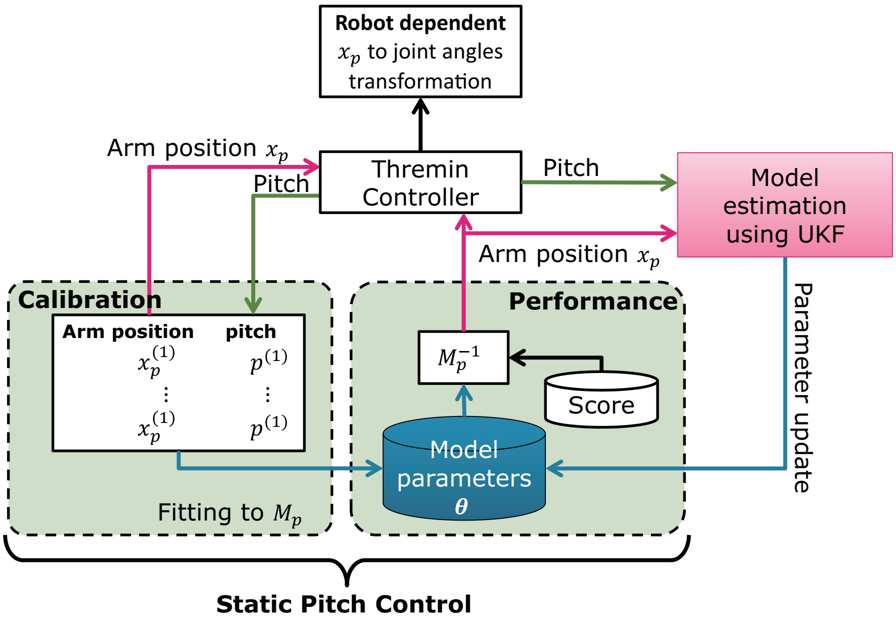


Figure 3.3: Overview of Robot Thereminist System

Assumption

The true parameter $\boldsymbol{\theta}_T(t)$ performs a random walk.

The initial parameter of pitch model $\boldsymbol{\theta}(0)$ is estimated in advance.

Inputs

Desired pitch trajectory $q(t-1)$

Estimated parameter $\hat{\boldsymbol{\theta}}(t-1)$

Observed pitch $p(t-1)$

Outputs

Estimated parameter $\hat{\boldsymbol{\theta}}(t)$

Pitch-control arm position $\hat{x}_p(t) = M_p^{-1}(q(t); \hat{\boldsymbol{\theta}}(t))$

This is a tracking problem of $\boldsymbol{\theta}_T(t)$ to minimize the squared error of $q(t)$ and $p(t)$. The robot estimates the parameter $\hat{\boldsymbol{\theta}}(t)$ from three values: a given musical score $q(t-1)$, an observed theremin's pitch $p(t)$, and the estimated model parameter $\hat{\boldsymbol{\theta}}(t-1)$. Using $\hat{\boldsymbol{\theta}}(t-1)$, the robot moves its arm to $\hat{x}_p(t)$. Next, the robot observes the pitch $p(t)$ that is generated from the theremin with the unknown model parameter $\boldsymbol{\theta}_T(t)$. Using $p(t)$, the robot estimates the parameter $\hat{\boldsymbol{\theta}}(t+1)$.

3.4.2 Unscented Kalman Filter

Overview

The UKF is an iterative estimation method of a nonlinear system. It is based on the unscented transform (UT), or the sigma-point transform, which calculates the mean and covariance of a symmetric probability distribution after propagating through any nonlinearity. They are calculated by the weighted summation of deterministically selected sample points called *sigma points*. The number of sigma points is $2D + 1$, where D is the dimension of the distribution.

The UKF differs from both the EKF and the particle filter. The EKF approximates the nonlinearity with the first-order approximation using the Jacobians of the state-update and observation function [95]. When the system's nonlinearity is strong, the EKF makes serious estimation errors. Here, the strength of the nonlinearity is defined as the summation of the Taylor expansion terms higher than the second order. In contrast, the UKF approximates more accurately because it considers the Taylor expansion terms higher than the second order. The particle filter, a kind of Monte Carlo filters, estimates the hidden state using hundreds of samples, which are drawn from a proposal distribution stochastically [96]. The UKF uses much smaller number of samples determined automatically from the dimension of the state. The samples are also deterministically selected from the mean and covariance before the nonlinear propagation.

Unscented Transform (UT)

The UT is a method that estimates the mean and covariance after propagating a probability distribution through a nonlinear function using those before the propagation. The most accurate solution for this problem the Monte Carlo approximation: draw many samples from a prior distribution, propagate them through the nonlinear function, and calculate the mean and covariance from them. However, this method has a high computation cost because it requires many samples.

The UT can achieve the similar performance with deterministically selected sigma points.

Let $\mathbf{z} = f(\mathbf{x})$ be a given nonlinear transformation. Let $\bar{\mathbf{x}} \in \mathbf{R}^D$ and $\mathbf{\Sigma_x} \in \mathbf{R}^{D\times D}$ be the mean and the covariance matrix of a probability distribution *before* the transformation, and $\bar{\mathbf{z}} \in \mathbf{R}^D$ and $\mathbf{\Sigma_z} \in \mathbf{R}^{D\times D}$ be those of *after* the transformation.

First, given $\bar{\mathbf{x}}$ and $\mathbf{\Sigma_x}$, we calculate the $2D+1$ sigma points, $\chi_0 \cdots \chi_{2D}$:

$$\begin{aligned} \chi_0 &= \bar{\mathbf{x}} && (3.7)\\ \chi_i &= \bar{\mathbf{x}} + (\sqrt{D\mathbf{\Sigma_x}})_i \quad (i = 0 \cdots D) && (3.8)\\ \chi_{i+D} &= \bar{\mathbf{x}} - (\sqrt{D\mathbf{\Sigma_x}})_i \quad (i = 0 \cdots D) && (3.9)\end{aligned}$$

where $(\cdot)_i$ denotes the l th column of a matrix and $\sqrt{\cdot}$ denotes the square root of a matrix; the square root A of M, $M = AA^H$, is calculated using Cholesky decomposition. Next, we calculate the weights, $w_0 \cdots w_{2D}$ for each sigma point:

$$w_i = \begin{cases} \kappa/(D+\kappa) & \text{if } \; i = 0 \\ 1/(2(D+\kappa)) & otherwise \end{cases} \tag{3.10}$$

where κ is the scaling parameter. Hereafter, we define the operator Sigma

$$[\chi_0 \cdots \chi_{2D}, w_0 \cdots w_{2D}] = \mathrm{Sigma}(\bar{\mathbf{x}}, \mathbf{\Sigma_x}) \tag{3.11}$$

as the process that calculates the sigma points and corresponding weights from the mean $\bar{\mathbf{x}}$ and covariance $\mathbf{\Sigma_x}$.

Then, the sigma points are propagated through the function f:

$$Z_i = f(\chi_i) \quad (i = 0 \cdots 2D) \tag{3.12}$$

Finally, $\bar{\mathbf{z}}$ and $\mathbf{\Sigma_z}$ calculated by the weighted summation of the sigma points after propagation:

$$\begin{aligned} \bar{\mathbf{z}} &= \sum_{i=0}^{2D} w_i Z_i && (3.13)\\ \mathbf{\Sigma_z} &= \sum_{i=0}^{2D} w_i (Z_i - \bar{\mathbf{z}})(Z_i - \bar{\mathbf{z}})^T && (3.14)\end{aligned}$$

Note that this procedure is performed only with the given mean and covariance of a prior distribution and the nonlinear function. No Jacobian or probabilistic estimations are required, unlike the EKF or the particle filters.

Kalman Filter using UT

The Kalman filter is expanded to nonlinear state update function f and nonlinear observation function h using UT [94]. The basic equations of the state update and observation equations are:

$$\begin{aligned} \mathbf{x}(t+1) &= f(\mathbf{x}(t), \mathbf{v}(t)) \qquad (3.15) \\ \mathbf{z}(t) &= h(\mathbf{x}(t), \mathbf{w}(t)) \qquad (3.16) \end{aligned}$$

where t denotes time, $\mathbf{x}(t) \in \mathbf{R}^{D_x}$ and $\mathbf{v}(t) \in \mathbf{R}^{D_v}$ denote a state and state-transition noise, and $\mathbf{z}(t) \in \mathbf{R}^{D_z}$ and $\mathbf{w}(t) \in \mathbf{R}^{D_w}$ denote an observation and observation noise, respectively. The means of $\mathbf{v}(t)$ and $\mathbf{w}(t)$ are both zero, and their covariances are $\mathbf{\Sigma_v}$ and $\mathbf{\Sigma_w}$.

First, we define the augmented state and its covariance matrix:

$$\begin{aligned} \mathbf{x}^a &= (\mathbf{x}^T, \mathbf{v}^T, \mathbf{w}^T)^T \qquad (3.17) \\ \mathbf{\Sigma_x}^a &= \begin{pmatrix} \mathbf{\Sigma_x} & 0 & 0 \\ 0 & \mathbf{\Sigma_v} & 0 \\ 0 & 0 & \mathbf{\Sigma_w} \end{pmatrix} \qquad (3.18) \end{aligned}$$

where $\mathbf{x}^a \in \mathbf{R}^D$, $\mathbf{\Sigma_x}^a \in \mathbf{R}^{D\times D}$, $D = D_x + D_v + D_w$. We assume that $\mathbf{x}$, $\mathbf{v}$, and $\mathbf{w}$ are uncorrelated each other. Then, the UKF algorithm is summarized as follows.

1. Calculate the $2D+1$ sigma points from the state $\hat{\mathbf{x}}^a(t-1)$ and covariance $\hat{\mathbf{P}}^a_x(t-1)$ at time $t-1$.

   $$[\chi^a_{0,t-1} \cdots \chi^a_{2D,t-1}, w_0 \cdots w_{2D}] = \text{Sigma}[\hat{\mathbf{x}}^a(t-1), \hat{\mathbf{P}}^a_x(t-1)] \qquad (3.19)$$

   where $\chi^a_{i,t-1}$ denotes the ith sigma point of augmented state $\mathbf{x}(t-1)^a$ at time t-1. The weights w_i have no time subscription because they are time independent.

2. Estimate the mean and covariance after state update.

$$\begin{aligned}
\chi_{i,t} &= f(\chi^a_{i,t-1}, \mathbf{0}) && (3.20)\\
\hat{\mathbf{x}}^-(t) &= \sum_{i=0}^{2D} w_i \chi_{i,t} && (3.21)\\
\hat{\mathbf{P}}^-_x(t) &= \sum_{i=0}^{2D} w_i(\chi_{i,t} - \hat{\mathbf{x}}^-_n(t))(\chi_{i,t} - \hat{\mathbf{x}}^-_n(t))^T && (3.22)
\end{aligned}$$

3. Estimate the observation and error covariance:

$$\begin{aligned}
Z_{i,t} &= h(\chi_{i,t}, \mathbf{0}) && (3.23)\\
\hat{\mathbf{z}}(t) &= \sum_{i=0}^{2D} w_i Z_{i,t} && (3.24)\\
\hat{\mathbf{P}}_z(t) &= \sum_{i=0}^{2D} w_i(Z_{i,t} - \hat{\mathbf{z}}(t))(Z_{i,t} - \hat{\mathbf{z}}(t))^T && (3.25)
\end{aligned}$$

4. Calculate the covariance between the state estimation and observation:

$$\hat{\mathbf{P}}_{xz}(t) = \sum_{i=0}^{2D} w_t^{(i)}(\chi_{i,t} - \hat{\mathbf{x}}^-(t))(Z_{i,t} - \hat{\mathbf{z}}(t))^T \tag{3.26}$$

5. Calculate the Kalman gain:

$$\mathbf{K} = \hat{\mathbf{P}}_{xz}(t)\left(\hat{\mathbf{P}}_z(t)\right)^{-1} \tag{3.27}$$

6. Modify the state and covariance estimation using the observation $\mathbf{z}(t)$ at time t:

$$\begin{aligned}
\hat{\mathbf{x}}(t) &= \hat{\mathbf{x}}^-(t) + \mathbf{K}(\mathbf{z}(t) - \hat{\mathbf{z}}(t)) && (3.28)\\
\hat{\mathbf{P}}_x(t) &= \hat{\mathbf{P}}^-_x(t) - \mathbf{K}\hat{\mathbf{P}}_z(t)\mathbf{K}^T && (3.29)
\end{aligned}$$

7. Augment the estimation $\hat{\mathbf{x}}(t)$ and $\hat{\mathbf{P}}_x(t)$:

$$\begin{aligned}
\hat{\mathbf{x}}^a(t) &= (\hat{\mathbf{x}}^{-T}(t), \mathbf{0}^T, \mathbf{0}^T)^T && (3.30)\\
\hat{\mathbf{P}}^a_x(t) &= \begin{pmatrix} \hat{\mathbf{P}}_x(t) & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{\Sigma}_v & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \boldsymbol{\Sigma}_w \end{pmatrix} && (3.31)
\end{aligned}$$

Then, go back to the first step.

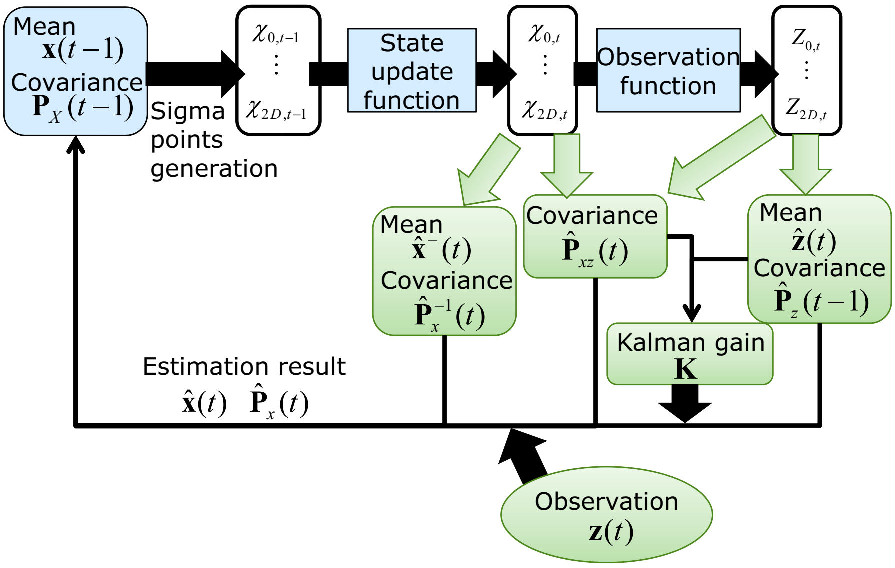


Figure 3.4: Data flow of unscented Kalman filter

Figure 3.4 illustrates the data flow of the UKF. The next state is estimated by calculating the weighted summation of the sigma points after the state update function f. Similarly, the observation is estimated by calculating the weighted summation of the sigma points after the observation function h. Finally, the estimations of the next state are modified using the observations.

3.4.3 Application to Adaptive Pitch Control

Model Design

We design the state update and observation functions for an adaptive pitch control. Let the state be the model parameter $\boldsymbol{\theta} \in \mathbf{R}^4$. The state update function is designed as a random walk according to our assumption discussed in section 3.4.1. The observation function is designed using the pitch model M_p (Eq. (3.1)) because we observe the theremin's pitch.

The state space model is summarized as follows.

$$\begin{aligned}
&\text{State update :}\\
\boldsymbol{\theta}(t) &= f(\boldsymbol{\theta}(t-1), \mathbf{v}(t-1))\\
&= \boldsymbol{\theta}(t-1) + \mathbf{v}(t-1) \qquad (3.32)\\
\theta_0(t) &= \max(\theta_0(t), 1+\epsilon) \qquad (3.33)\\
\theta_1(t) &= \max(\theta_1(t), 0) \qquad (3.34)\\
&\text{Observation :}\\
p(t) &= h(\boldsymbol{\theta}(t), w(t))\\
&= M_p(x_p(t); \boldsymbol{\theta}(t)) + w(t) \qquad (3.35)
\end{aligned}$$

where $\mathbf{v}(t) \in \mathbf{R}^4$ and $w(t) \in \mathbf{R}$ denote the state update noise and the observation noise, respectively. Eqs. (3.33) and (3.34) are additional constraints to ensure that M_p in Eq. (3.1) is real and finite. Although these constraints make indifferentiable points, the state-update function is still valid for the UKF because it requires no Jacobian.

The observation function should be discussed because if we simply substitute Eq. (3.2) with Eq. (3.35), the function is $p(t) = q(t)$, which is not a nonlinear mapping of the hidden state $\boldsymbol{\theta}(t)$; i.e., the observation function does not *observe* the hidden state. This is not true because the parameters used for Eqs. (3.1) and (3.2) are different. Let $\boldsymbol{\theta}_T(t)$ be the true parameter of the theremin. Then, the observation function after substitution is

$$p(t) = M_p(M_p^{-1}(q(t); \hat{\boldsymbol{\theta}}(t)); \boldsymbol{\theta}_T(t)). \qquad (3.36)$$

If the parameter estimation is not perfect, i.e., $\hat{\boldsymbol{\theta}} \neq \boldsymbol{\theta}_T$, M_p and M_p^{-1} are not the inverse functions. Therefore, Eq. (3.35) is a nonlinear mapping until the parameter estimation succeeds. The parameter estimation is unlikely to become perfect because the unknown noise $\mathbf{v}(t)$ fluctuates the true parameter $\boldsymbol{\theta}_T(t)$. The $\mathbf{v}(t)$ corresponds to the physical noise such as the temperature change and the theremin's internal noise.

Physical Constraints

A robot's arm speed is limited because of its physical limitations. Even if the controller commands an optimum arm position, the arm cannot reach the position if it is too far from the current position. We then set the maximum arm speed x_{plim} to incorporate this constraint defined as:

$$x_{plim} > |x_p(t) - x_p(t+1)| \tag{3.37}$$

If the $|x_p(t) - x_p(t+1)|$ exceeds x_{plim}, the next arm position is $x_p(t+1) = x_p(t) + x_{plim}$. Then, we guarantee that the arm speed is limited. In the following experiments, we empirically set $x_{plim} = 0.05$.

3.5 Experiments

We evaluate our method both in simulation and on a real robot. In section 3.5.1, we evaluate the pitch error in a simulated dynamic environment and compare our method with two adaptive pitch control methods, the EKF and a block-wise update. The EKF approximates nonlinear models by the first-order as discussed in 3.4.2. The block-wise update method accumulates the pairs of pitches and arm positions for duration, then, re-estimates the model parameters by a nonlinear optimization method. This method is a simple extension of the static pitch control discussed in section 3.3.2. In 3.5.2, we demonstrate our system implemented on a humanoid robot, HRP-2, that plays the song *Aura Lee*.

3.5.1 Simulation Experiment

Parameterization of Dynamic Environment

Generation of a dynamic environment for simulation should have physical meaning. Therefore, just fluctuating the model parameters $\boldsymbol{\theta}$ is inappropriate because it has no physical meaning. In this experiment, we generate the model parameters using the linear interpolation of actually measured model parameters.

We define a time series of an environment parameter $e \in [0, 1]$. For each value of e, we define a model parameter $\boldsymbol{\theta}$ using the $N + 1$ model parameters, $\boldsymbol{\theta}_0 \cdots \boldsymbol{\theta}_N$ measured in advance. The projection from e to measured model parameters is defined using the linear interpolation of the set of pairs, $(i/N, \boldsymbol{\theta}_i)$ where $i = 0 \cdots N$.

Then, we design a dynamic environment using the time series of $e(t)$. A sinusoidal function with different frequencies ω is used to control the speed of environmental change:

$$e(t) = 0.5 \sin(2\pi\omega) + 0.5 \tag{3.38}$$

with the frequency set to $\omega = 1, ..., 15$. This design represents interference by the co-player's motion. Since the co-player plays the instrument close to the theremin, its motion is like oscillation, i.e., getting closer to and going far away from the robot.

Evaluated Methods and Evaluation Criteria

We evaluated three methods: The UKF, the EKF, and block-wise parameter estimation. The configurations of the compared methods are as follows: For the UKF, the initial values are: $\boldsymbol{\Sigma}_\mathbf{x} = \boldsymbol{\Sigma}_\mathbf{v} = diag(5, 5, 5, 5)$, $\Sigma_w = 10$, $\kappa = 2$. For the EKF, we used the Jacobian of Eq. (3.1), which is derived in [40]. Note that, to ensure the estimated parameters are limited values, we added a constraint $\theta_0 \geq x_p$ by substituting θ_0 with x_p when the constraint does not hold. The noise covariances $\boldsymbol{\Sigma}_\mathbf{v}$ and $\boldsymbol{\Sigma}_\mathbf{w}$ are the same as those of the UKF. For the block-wise parameter estimation, we set the interval of parameter re-estimation to five seconds. In other words, the parameters for arm control in the current five second is estimated from the pairs of arm positions and pitches collected in the past five seconds.

The evaluation criterion is the pitch error c defined by $c = 1200 \log(p/q)$, where c, p, and q denote the pitch error [cent], the observed pitch [Hz], and the desired pitch [Hz], respectively. The 100 [cent] pitch error is equivalent to the half-note error. For each condition, we performed the experiment for 10 times and then averaged the errors.

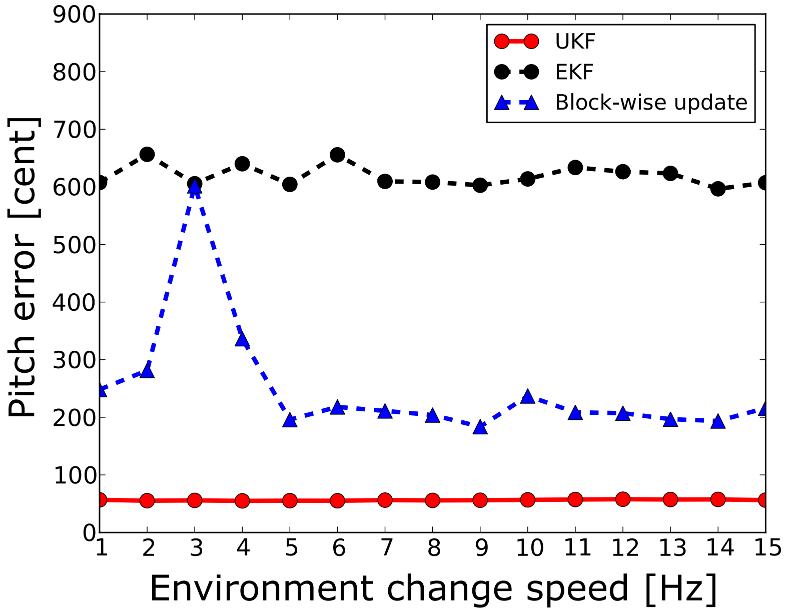


Figure 3.5: Pitch errors in simulated dynamic environments

Results

Fig. 3.5 summarizes the pitch errors for each condition. The vertical axis denotes the pitch error [cent]. The horizontal axis denotes the frequency ω of the environment parameter $e(t)$ in Eq. (3.38), meaning that how fast the environment changes. A dashed blue line shows the result of the block-wise update method, a dashed blue line shows the result of the EKF, and a solid red line shows the result of the UKF (proposed) method.

The UKF method performed the lowest error. This is because the UKF can estimate nonlinear model precisely. The EKF performed the highest error. because the observation model (Eq. (3.1)) has high nonlinearity. In addition, even when we use the constant $e(t)$ as the stable environment, the estimation performance is low because of the error accumulation. The block-wise update performed the middle of the EKF and UKF. It is better than the EKF because it executes a nonlinear optimization method many times. However, since it assumes that the true model parameters change only for every five seconds, its performance is lower than that

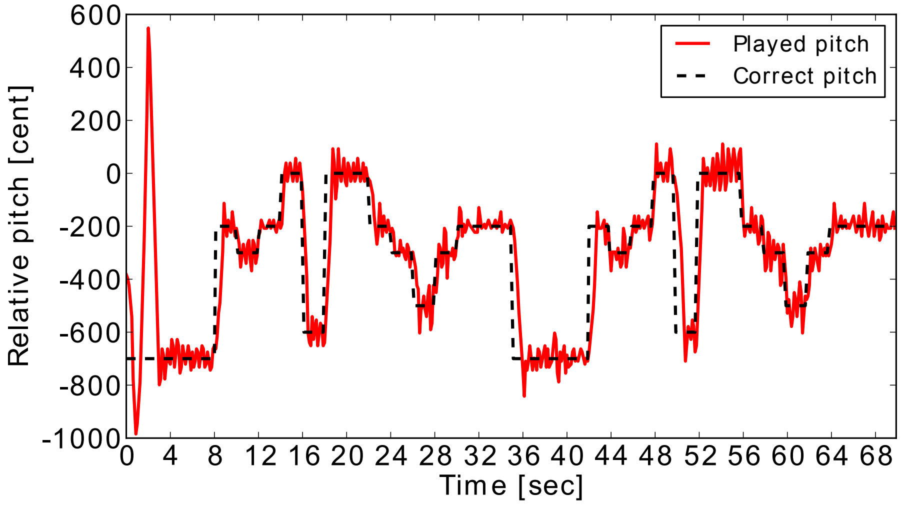


Figure 3.6: Pitch trajectories of the theremin

of the UKF. The pitch error is especially high at $\omega = 3$ in Fig. 3.5. This suggests the sensitivity to the timing of the re-estimation. The worst case for this method is that the model parameters for parameter estimation and playing are completely different. In other words, the environment parameter for parameter estimation is around $e(t) = 1$, whereas that for the theremin playing is around $e(t) = 0$.

3.5.2 Robot Experiment

We implement our method on a humanoid robot, HRP-2, to demonstrate the performance of our system on a real robot. The robot's arm position and the model parameters are updated at 62.5 [msec] interval. We adopt the auto correlation based pitch estimation shown in, e.g., [99]. We used the initial parameter measured in the different configuration to test the adaptation capability. For a musical score, we used an American folk song *Aura Lee*.

Fig. 3.6 shows the pitch trajectory of the theremin played by the robot. The horizontal line denotes the time [sec] and the vertical line denotes the relative pitch [cent], where the cent c is calculated from the pitch h [Hz] by $c = 1200 \log_2(h/220)$. Here, we used the pitch of 220 Hz, the note $A3$, as the 0 [cent]. The black broken

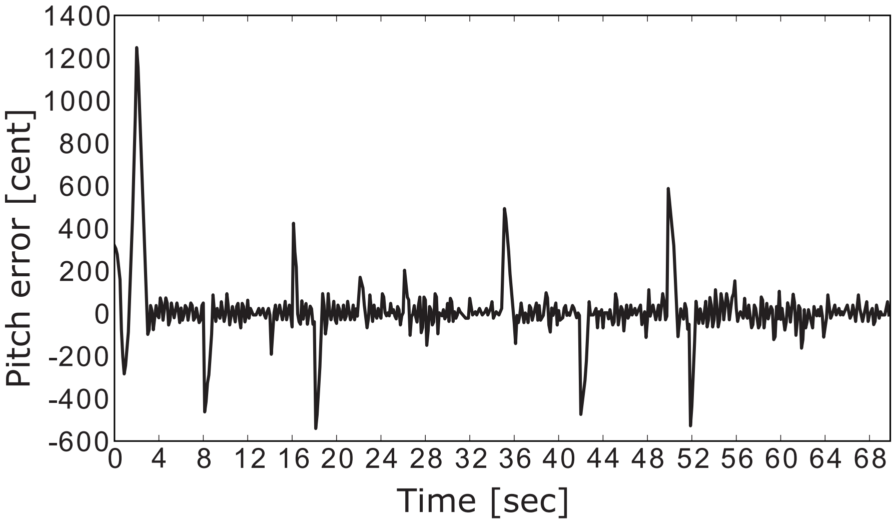


Figure 3.7: Pitch errors of the theremin

line denotes the desired pitch trajectory calculated from the score and the red solid line denotes the played pitch trajectory.

As the pitch converges to the correct pitch in 4 [sec], the robot can play the pitch correctly even if the initial pitch is incorrect. However, there remains fluctuation because of two reasons: (1) the estimated pitch and the arm position have time difference and (2) the target pitch changes frequently before the error converges.

Fig. 3.7 shows the pitch error calculated from Fig. 3.6. The horizontal axis denotes the time and the vertical axis denotes the pitch error [cent]. The mean absolute pitch error was 72.9 [cent], which is less than the half-note. The trajectory has some peaks, e.g., at 8, 16 and 36 seconds, because it takes time to reach the next pitch after the desired pitch changes.

3.6 Summary

We presented an adaptive pitch control method for accurate theremin playing in a dynamic environment for human-robot ensemble. We constructed the theremin's pitch model and its state space model. Utilizing the UKF, we solved the nonlinear model. Experimental results showed that our method achieves more accurate pitch control than both EKF and a block-wise model update method. We also demonstrated the Robot Thereminist system on a real robot. The robot played the given score correctly even if the initial parameter is different from the true one.

Chapter 4

Two-person Ensemble Model using Coupled Oscillators

For the first step to extend the solo-playing robot to the co-playing robot, we develop a timing synchronization method for two-person ensemble. We develop the ensemble model using the coupled oscillator model, and implement on the Robot Thereminist for the two-person ensemble.

4.1 Introduction

This chapter aims to achieve an ensemble between a human and a robot, especially, a two-person ensemble between a robot thereminist and a human drummer, which is one of the simplest forms of an ensemble. We define an ensemble as *a synchronized performance involving interactions among independent players.* According to our definition, the ensemble consists of three components: (1) a human player, (2) a robot player, and (3) a synchronization method. The technical challenges are the second and third components. For the second one, we have already developed Robot Thereminist in Chapter 3. For the third one, Otsuka *et al.* developed a simple synchronization method [15] using a beat-tracking [34]. The Robot Thereminist is required to predict the human drummer's onset time for synchronization. However, conventional methods only adapt to an interval of a human, ignoring the

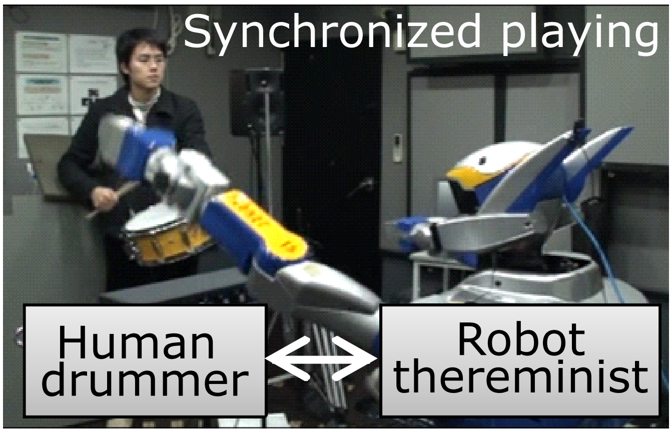


Figure 4.1: Snapshot of our ensemble system

human's adaptation to the robot. This limits the synchronization accuracy.

We present a novel synchronization method using a coupled oscillator model for predicting the human drummer's onset timing. The behavior of coupled oscillator model has been actively studied from theoretical analysis to applications to explain the behaviors of various physical phenomena [3], e.g., frogs' calling behavior [14]. To apply the concepts of the model, we assume that each participant is a self-sustaining oscillator and they adjust their timings using onset timings. Based on this idea, our synchronization method reduces the difference of two participants' onset timings compared with the existing Robot Thereminist, which only adjusts the robot's playing speed using the human's drumming speed. This is because the robot predicts the human's drum hitting considering two-way influence between the human and the robot.

The advantage of the method is twofold: (1) A robot can predict another participant's behavior. This is essential for synchronized motion generation because the robot can start moving its body before the onset time comes. (2) Our model can be applied to various ensemble situations by changing the parameters such as coupling strengths, and the number of participants. This extension is discussed in Chapter 5.

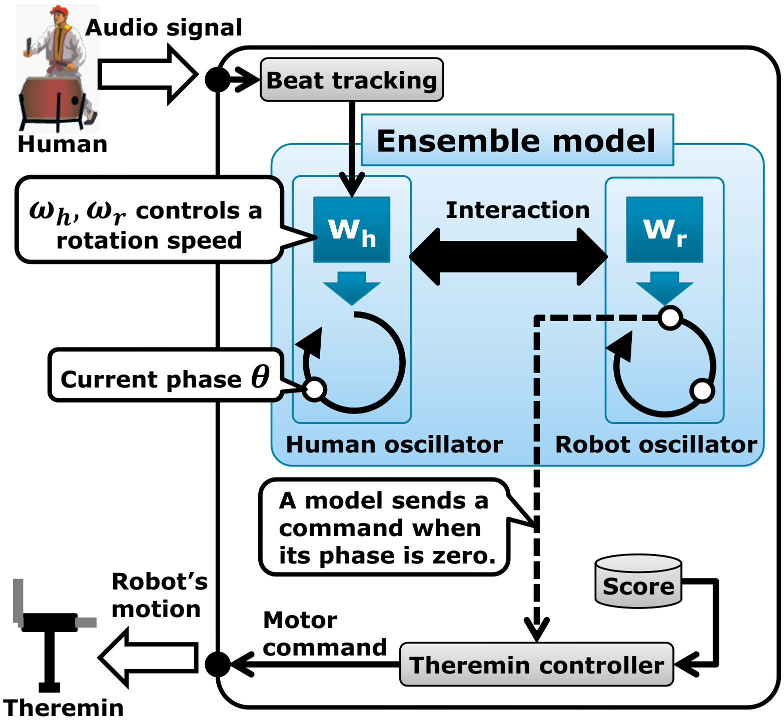


Figure 4.2: Block diagram of duet ensemble system

4.2 Ensemble System with Coupled Oscillators

This section describes the ensemble system for two-person ensemble. First, we present an overview of our system in Section 4.2.1. Then, we explain its two main components: a real time beat tracking method that recognizes a human's drumming speed in Section 4.2.2 and an ensemble model for synchronized performance in Section 4.2.3. Note that another main component, Robot Thereminist system is explained in Chapter 3,

4.2.1 Overview of Two-Person Ensemble System

Figure 4.2 is an overview of the ensemble system consisting of three main modules: (1) a beat-tracking module for estimating the onset of a human's playing, (2) a robot-control module for playing music, and (3) an ensemble model for predicting the human's behavior.

Our system works as follows: it records the sound of a human playing through its own microphone. Then, the beat-tracking module estimates the beat interval in the sound of the human's playing. Our model updates the angular velocity of the human's oscillator model using the estimated interval. This model is used to simulate and predict the human's behavior. The robot waits until the phase of the robot's oscillator becomes zero by updating the human's and the robot's oscillators. When the phase becomes zero, our model commands the theremin controller to play the next musical note from a given score.

4.2.2 Real Time Beat Tracking

We briefly review the beat tracking method developed by Murata *et al.* [34]. The algorithm has three phases: (1) tempo estimation, (2) beat detection, and (3) beat time prediction. The input is a musical signal of the human's performance.

Tempo Estimation

Let $P(t, f)$ be the Mel-scale power spectrogram of the given musical signal where t is the time index and f is the Mel-filter bank bin. Since we use 64 filter banks, we define $f = 0, 1, ..., 63$. Then, we apply Sobel filtering, which is a widely used edge emphasis method in image processing, to $P(t, f)$ to emphasize the onset and obtain $d(t, f)$:

$$\begin{aligned} d(t,f) \;\; = \;\; & -P(t-1,f+1)+P(t+1,f+1) \\ & -2P(t-1,f)+2P(t+1,f) \\ & -P(t-1,f-1)+P(t+1,f-1) \end{aligned} \tag{4.1}$$

where $f = 1, 2, ..., 62$ because the edges $f = 0, 63$ has no values after Sobel filtering. Then, we derive the onset belief $d_{inc}(t, f)$ by the equation:

$$d_{inc}(t,f) = \begin{cases} d(t,f) & \text{if } \; d(t,f) > 0 \\ 0 & \text{otherwise} \end{cases} \tag{4.2}$$

The tempo is defined as the interval between two neighboring beats. To estimate the tempo, we first calculate the normalized cross correlation (NCC), $R(t,i)$:

$$R(t,i) = \frac{\sum_{f=1}^{62}\sum_{k=0}^{W-1} d_{inc}(t-k,f)d_{inc}(t-i-k,f)}{\sqrt{\sum_{f=1}^{62}\sum_{k=0}^{W-1} d_{inc}(t-k,f)^2 \cdot \sum_{f=1}^{62}\sum_{k=0}^{W-1} d_{inc}(t-i-k,f)^2}} \tag{4.3}$$

where W and i denote the window length and the shift offset. To stabilize the tempo estimation, we derive the local peak of $R(t,i)$ defined as

$$R_p(t,i) = \begin{cases} R(t,i) & \text{if } R(t,i-1) < R(t,i) \text{ and } R(t,i+1) < R(t,i) \\ 0 & \text{otherwise} \end{cases} \tag{4.4}$$

Next, we calculate the beat interval $I(t)$, which is an inverse of the musical tempo. For each time t, we determine $I(t)$ using $R_p(t,i)$. Basically, $I(t)$ is chosen as $I(t) = \underset{i}{\operatorname{argmax}} R_p(t,i)$. However, this naive estimation will fluctuate rapidly when a complicated drumming pattern is in the input signal. To prevent the beat interval misestimation, we derive $I(t)$ shown in Eq. (4.5). Let I_1 and I_2 be the first and second peaks in $R_p(t,i)$ when moving i.

$$I(t) = \begin{cases} 2\|I_1 - I_2\| & \text{if } (\|I_{n2} - I_1\| < \delta \text{ or } \|I_{n2} - I_2\| < \delta) \\ 3\|I_1 - I_2\| & \text{if } (\|I_{n3} - I_1\| < \delta \text{ or } \|I_{n3} - I_2\| < \delta) \\ I_1 & \text{otherwise} \end{cases} \tag{4.5}$$

where $I_{n2} = 2\|I_1 - I_2\|$, $I_{n3} = 3\|I_1 - I_2\|$, and δ denotes an error-margin parameter.

Beat interval $I(t)$ is confined to a range between 61 – 120 beats per minute (bpm). This is because this range is suitable for controlling the robot's arm.

Beat Time Detection

The beat time is detected using the onset belief $d_{inc}(t,f)$ and the beat interval $I(t)$. First, we define two kinds of beat reliabilities: the reliability of the neighboring beat and that of the continuous beat. The neighboring beat reliability, $S_n(t,i)$, defined in Eq. (4.6) measures how precisely the adjacent beat lies on the $I(t)$ interval.

$$S_n(t,i) = \begin{cases} \sum_{f=1}^{62} (d_{inc}(t-i,f) + d_{inc}(t-i-I(t),f)) & \text{if } (i \leq I(t)) \\ 0 & \text{if } (i > I(t)) \end{cases} \tag{4.6}$$

The continuous beat reliability $S_c(t,i)$ defined in Eq. (4.7) measures how precisely the sequence of musical beats lies on the estimated beat intervals.

$$S_c(t,i) = \sum_{m=0}^{N_{beats}} S_n(T_p(t,m),i) \tag{4.7}$$

$$T_p(t,m) = \begin{cases} t - I(t) & \text{if } m = 0 \\ T_p(t,m-1) - I(T_p(t,m)) & \text{if } m \geq 1 \end{cases}$$

where $T_p(t,m)$ is the m-th previous beat time at time t, and N_{beats} is the number of beats used to calculate the reliability. These two reliabilities are then integrated into the beat reliability $S(t)$:

$$S(t) = \sum_i S_n(t-i,i) \cdot S_c(t-i,i) \tag{4.8}$$

Finally, the latest beat time, $T(n+1)$, is detected as one of the peaks in $S(t)$ that is the closest to $T(n) + I(t)$, where $T(n)$ is the nth beat time.

Beat Time Prediction

We predict the next beat time T' by extrapolation using the latest beat time $T(n)$ and the current beat interval $I(t)$.

$$T' = \begin{cases} T_{tmp} & \text{if } T_{tmp} \geq \frac{3}{2}I(t) + t \\ T_{tmp} + I(t) & \text{otherwise} \end{cases} \tag{4.9}$$

$$T_{tmp} = T(n) + I(t) + (t - T(n)) - \{(t - T(m)) \mod I(t)\} \tag{4.10}$$

4.2.3 Coupled Oscillator Model for Synchronization

This section describes coupled oscillator and its application for a synchronized ensemble. We explain the oscillator model in Section 4.2.3 and its application to the ensemble system in Section 4.2.3.

The key advantage of the model is that the robot which has the model can know the time when a human hits the drum through the model. Because of this advantage, the robot can reduce the time delay because it can start moving beforehand to generate the rhythm on time.

General Description of Coupled Oscillator Model

A coupled-oscillator model consists of two components: oscillators and their interactions. The oscillator is a self-sustaining system, which keeps working repeatedly by itself. For example, a pendulum clock and a drummer who maintains the same speed can be considered to be oscillators.

We define an oscillator's phase $\phi(t)$ with

$$\phi(t) = (\phi_0 + 2\pi t/T_{osc}) \bmod 2\pi \tag{4.11}$$

where t denotes the time, T_{osc} denotes the period of the oscillator, and ϕ_0 denotes the initial phase. $\phi(t) = 2\pi n$ denotes the same state in an oscillator. The oscillator's dynamics is described by the differential equation of its phase:

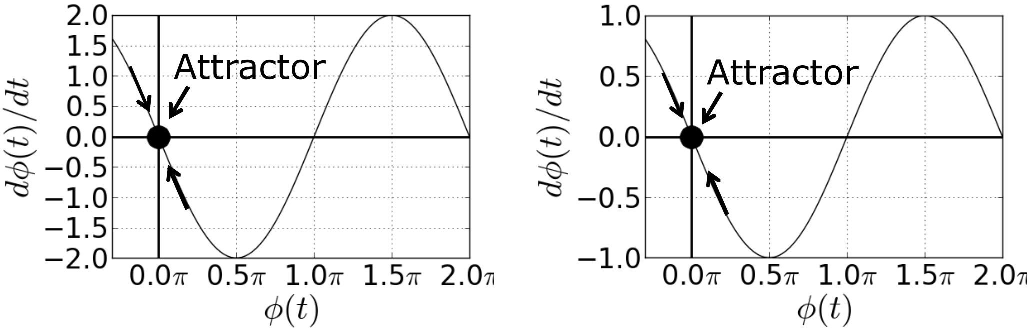
$$\frac{d\phi_1}{dt} = \omega_1 \tag{4.12}$$

where ω_1 denotes an angular frequency of the oscillator. When two oscillators interact, we call them *coupled.* A coupling is represented by adding a 2π-periodic function to Eq. (4.12). We show a coupled two oscillators below:

$$\frac{d\phi_1}{dt} = \omega_1 + K_1 Q(\phi_2 - \phi_1) \tag{4.13}$$

$$\frac{d\phi_2}{dt} = \omega_2 + K_2 Q(\phi_1 - \phi_2) \tag{4.14}$$

where ϕ_1 and ϕ_2 are the phases of the coupled oscillator, the Q is a coupling term which is a 2π-periodic function of the phase difference, K_1 and K_2 are the positive coupling strengths, and ω_1 and ω_2 are the natural frequencies.



(a) $K_1 = K_2 = 1$

(b) $K_1 = 0, K_2 = 1$

Figure 4.3: Attractors in Kuramoto model

We present the Kuramoto model, which is a basic oscillator model [2].

$$\frac{d\phi_1}{dt} = \omega_1 + K_1 \sin(\phi_2 - \phi_1) \tag{4.15}$$

$$\frac{d\phi_2}{dt} = \omega_2 + K_2 \sin(\phi_1 - \phi_2). \tag{4.16}$$

The key feature of this model is that a sinusoidal function is used as a coupling term. We can hence analyze the behavior of these two oscillators. First, we define the phase difference, $\phi = \phi_1 - \phi_2$. Then, the dynamics of ϕ is described as:

$$\frac{d\phi}{dt} = \omega_1 - \omega_2 + K_1 \sin(-\phi) - K_2 \sin(\phi) \tag{4.17}$$

$$= (\omega_1 - \omega_2) - (K_1 + K_2) \sin(\phi) \tag{4.18}$$

Assuming the natural frequencies of two oscillators are the same, we can determine the behavior of them by plotting a graph of Eq. 4.18.

Figure 4.3 shows the behaviors of the two oscillators with two parameters, K_1 and K_2. The vertical axis denotes the differential coefficient of the phase difference and the horizontal axis denotes the phase difference. Figure 4.3 (a) plots the situation when two oscillators are coupled equally ($K_1 = K_2 = 1$). In this situation, two oscillators are synchronized when the phase difference is zero. Figure 4.3 (b) shows that even if only the second oscillator is influenced ($K_1 = 0, K_2 = 1$), the attractor is at the same place.

Application to Ensemble Model

We use two assumptions to apply the oscillator model to an ensemble system. q

1. A participant has an internal oscillator. He generates an onset when his oscillator's phase is zero. For example, a drummer hits a drum when his internal oscillator's phase becomes zero.

2. A participant knows the other's phase when its onset begins. For example, the partner of the drummer knows that the drummer's phase is zero when he hits the drum.

We focus on a two-person ensemble between a human drummer and the Robot Thereminist. We then define the rule for onset timing; for a drum sound, the onset timing is when the drum is hit. For a theremin sound, the onset timings are defined as the time when the pitch is changed. However, calculating the theremin's onset timings from the pitch trajectory is difficult because the trajectory has continuous value unlike a piano. Therefore, we hence rounded the trajectory down to the nearest 100 [cent] to emphasize the onset timings, then, we detect the onset timings.

We use the Kuramoto model in Eqs. 4.15 and 4.16 as the oscillator model. In addition, we add an update rule to reduce the robot's natural frequency to that of a human. This is because a drum usually dominates the rhythm of an ensemble.

Our ensemble model is summarized as follows:

Let ϕ_r and ϕ_h be the robot's and human's phases, K_r and K_h be the robot's and human's coupling strength, and μ be a learning coefficient.
The phase dynamics of two oscillators are:

$$\frac{d\phi_h}{dt} = \omega_h + K_h \sin(\phi_r - \phi_h) \tag{4.19}$$

$$\frac{d\phi_r}{dt} = \omega_r + K_r \sin(\phi_h - \phi_r) \tag{4.20}$$

The update rule of ω_r is:

$$\omega_r \leftarrow \omega_r + \mu(\omega_r - \omega_h) \tag{4.21}$$

4.3 Experiments

We evaluate the accurately of onset prediction with three experiments. The ensemble partner is different for each experiment: (1) a metronome as a completely accurate drummer, (2) a fluctuating metronome that simulates a drummer with fluctuation without entrainment, and (3) a human drummer who has both fluctuation and entrainment. Note that we do not evaluate the performance of beat tracking because it has already been evaluated by Murata *et al.* [34].

In the second experiment, we show a simulation result how much the amount of fluctuation or the coupling strengths affect the onset prediction accuracy. If the amount of fluctuation, i.e., the variance of IOIs, is too large, it is impossible to predict the onsets whereas the prediction should be stabilized if the variance is relatively small. This experiment presents the quantitative evaluation about the prediction error.

We also evaluate a human's onset prediction error to compare our method with human's ability. We asked four subjects who are non-professional drummers to listen to a sequence of one hundred 440Hz pure tones and press a key at the same time by predicting them. We prepared the sequences with four tempos (66, 80, 100, and 112 bpm), and two noise ratios (0 and 10%.) The noise ratio is defined

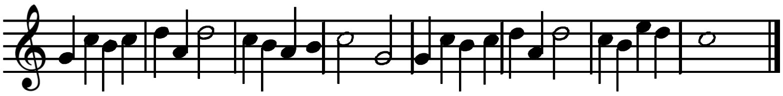


Figure 4.4: Musical score of Aura Lee

in section 4.3.3. The results are compared in sections 4.3.2 and 4.3.3.

4.3.1 Configurations

We implemented the Robot Thereminist system and our synchronization method on a humanoid robot, HRP-2. We placed the robot and the theremin, Etherwave Theremin of Moog Music Inc., at the 50 cm distance. ϕ_r, ϕ_h and ω_r are updated at 50 [msec] interval. We use an American folk song *Aura Lee* as a musical score shown in Fig. 4.4. We use four different metronome tempi: 66, 80, 100 and 112 bpm. These tempi cover the possible speeds of the beat tracking. For all experiments on the robot, three trials are conducted for each tempo.

The four parameters of the oscillator model are empirically set as: $K_r = 0.4$, $\omega_h = \omega_r = 2\pi/700$ and, $\mu = 0.01$. We set $K_h = 0$ for the first and second experiments because the metronome is never influenced by the robot. In contrast, we set $K_h = 0.4$ for the third experiment since the human drummer should be influenced by the robot. The value of $K_h = 0.4$ is equal to K_r; the human and the robot are influenced by each other with the same strength. The effect of the coupling parameters K_r and K_h are examined in Section 4.3.3.

We compare our onset prediction method with the extrapolation based method [15] as the baseline. This method directly uses the latest IOI estimate. If no onset correction mechanism is used, the initial timing difference persists through the performance. For example, if the robot starts playing 50 msec behind the human player, this delay is kept until the end of the performance. Only the human can adjust his/her onset timing to that of the robot, but this is not the goal of our targeted co-player robot. Both the robot and the human should mutually adjust their onset timings like a human-human ensemble.

The evaluation criteria are the mean onset error (Eq. (4.22)) and the normalized mean onset error (Eq. (4.23)). The normalized mean onset error is used to evaluate the regardless of the tempo by showing the prediction error in the note-length whose duration changes depending on the tempo. Let $onset_t(i)$ and $onset_d(j)$ be the theremin's ith and the drum's jth onset, respectively. The mean onset error e is defined as the mean of the minimum differences between $onset_t(i)$ and $onset_d(j)$:

$$e = \frac{1}{N} \sum_{j=1}^{N} \min_{i=1,\ldots,M} \left| onset_t(i) - onset_d(j) \right| \tag{4.22}$$

where N denotes the number of drum onsets and M denotes the number of theremin onsets. This criterion is same as [34] and [30] with the exception that they decide whether the onset prediction is "correct" by comparing a threshold for each error. In contrast, we directly use the error itself for detailed evaluation of onset prediction. The normalized mean onset error is defined as the mean onset error e normalized by the ground truth IOI:

$$\text{Normalized error} = \frac{e}{60/T} \tag{4.23}$$

where T is the ground truth tempo in bpm: $60/T$ is the ground truth IOI. The worst normalized error is 0.5 since if the error exceeds the half of the IOI, the onset is assigned to the next onset. Thus, the worst normalized error corresponds to the eighth-note error.

4.3.2 Ensemble with a Metronome

We evaluate the accuracy of the onset prediction using a metronome, which represents the "perfect" drummer, instead of using a real human drummer.

Fig. 4.5 shows the result. The horizontal and vertical axes denote the tempo of the metronome and the normalized mean onset error. The red bars denote the errors of our method, while the black bars denote those of the baseline method. The errors of our method are 0.106, 0.149, 0.158, and 0.156, and those of baseline

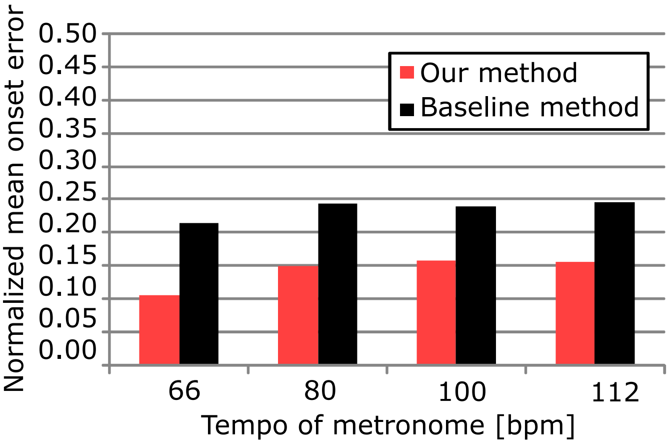


Figure 4.5: Onset errors in an ensemble with a metronome

method are 0.214, 0.244, 0.239, 0.245 for 66, 80, 100, and 112 bpm, respectively. Our method reduces onset errors of the baseline method by 39% on average. Therefore, the robot with our prediction method can play a melody in synchronization with a human more accurately than the baseline method.

The results with the human's onset prediction were 35.9, 39.8, 43.2, and 31.6 msec, with the tempos of 66, 80, 100, and 112 bpm, respectively. The normalized errors were less than 0.06 for every condition. Comparing these results with our method, this shows that there is a room for improvement to achieve the equivalent onset prediction capability of humans.

Next, we evaluate the sensitivity of the coupling strengths K_r and K_h. We set the coupling strengths K_r and K_h to $0, 0.3, 0.6, 1.0$. The tempo of the metronome is fixed to 80 bpm in call cases. The score used in this evaluation is a simple score: a sequence of notes increasing from C4 to D5 and then decreasing from D5 to C4 for 8 times, having 136 notes. This is for evaluation of the onset errors regardless of the note length. The results are summarized in Fig. 4.6. The onset error is significantly high if $K_r = 0$ because the zero coupling strength means no influence; the robot plays the score without prediction nor adaptation. On the other hand, when the K_r is more than zero, the errors are around 40 msec regardless of other

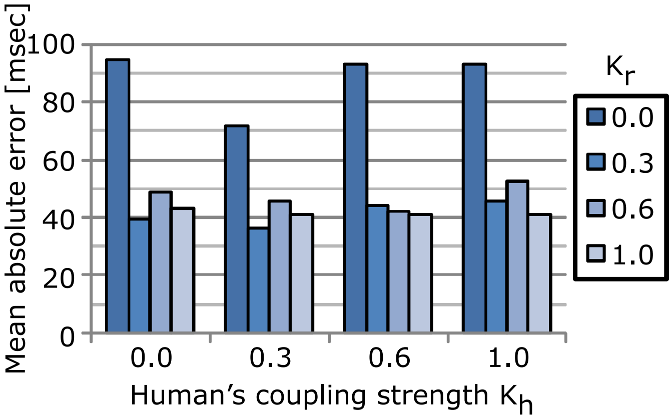


Figure 4.6: Onset errors in different coupling strengths. The horizontal line denotes K_h and the color denotes K_r.

conditions. Therefore, our method is stable to coupling strengths.

Then, we evaluate the convergence time after the tempo change. The score is the simple sequence defined in the previous paragraph. The tempo is switched for five times between 80 bpm and 100 bpm. We used two sets of coupling strengths: $(K_r, K_h) = (0.3, 0.3)$ and $(0.6, 0.6)$. The convergence time is defined as the time when the onset prediction error becomes less than 1000 samples (22.7 msec). The result is as follows: The mean convergence times are 2.31 sec for $(K_r, K_h) = (0.3, 0.3)$ and 2.01 sec for $(K_r, K_h) = (0.6, 0.6)$. The result suggests that a larger coupling strength accelerates the convergence of the phases of the both oscillators. This is because the larger strength means the more effect on the robot's coupling oscillator.

Finally, we evaluate the sensitivity of the learning coefficient μ using three values; 0.01, 0.001, and 0.0001. We fixed $(K_r, K_h) = (0.6, 0.6)$. Since μ is used to adapt to the human's natural frequency, we evaluate with the convergence time. The results are: 2.04 sec for $\mu = 0.01$, 2.03 sec for $\mu = 0.001$, and 2.01 sec for $\mu = 0.0001$. This result suggests that the convergence time to the changing tempo is more dependent on the coupling strengths than on the leaning coefficient.

4.3.3 Ensemble with a Fluctuating Metronome

We evaluate the fluctuating metronome in this experiment to simulate a human's drumming without entrainment. The standard deviation of the tempo fluctuation is 10% of the IOI. We also evaluate the error by changing the coupling strengths K_r, K_h, and the amount of the fluctuation in Section 4.3.3.

The fluctuated IOIs are the samples drawn from the normal distribution with the mean of the ground truth IOI and the standard deviation defined as the noise ratio. For example, if the tempo is 80 bpm, i.e., the IOI is 750 msec, and the noise ratio is 5%, the IOIs are the samples drawn from the normal distribution with the mean of 750 msec and the standard deviation of 750× 0.05 msec. The noise ratio simulates the human's inevitable tempo fluctuations whose amount depends on the drummer's capability of the rhythm keeping.

Synchronization with Fluctuating Metronome on a Robot

Fig. 4.7 shows the bar chart of the results. The horizontal and vertical axes represent the tempo of a metronome and the normalized mean onset error. The results reveal that the improvement with our method is suppressed compared to the first experiment while our method still outperforms the baseline method.

We compare the result to the human prediction error. For the four tempos, 66, 80, 100, and 112 bpm, the errors were as follows: When the noise ratio is 1%, the errors were 59.8, 94.3, 48.9, and 52.2 msec, respectively; when the noise ratio is 10%, the errors were 224.9, 259.7, 192.7, and 158.4 msec, respectively. The normalized mean onset errors on average were 0.235 for human, 0.256 for our method, and 0.278 for the baseline method. The errors of both ours and humans are significantly increased when the IOIs are fluctuated.

The reason of this error increase is that our model assumes that the IOIs changes gradually, which does not hold in the experiment. The result that the human's prediction error also increases for the fluctuating metronome supports that the use of the coupled oscillator model for the human behavior modeling.

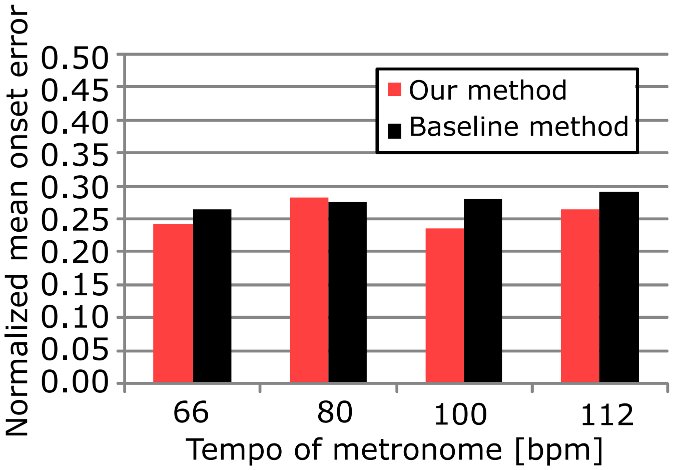


Figure 4.7: Onset errors in an ensemble with a fluctuating metronome

Synchronization with Fluctuating Metronome in Simulation

This section evaluates the onset prediction error using various noise ratios and coupling strengths in simulation. The setup for the simulation is as follows: We set the coupling strengths K_r and K_h to $0, 0.1, ..., 1.0$, the noise ratio to $1\%, 2\%, .., 20\%$, and the tempo to $66, 80, 100, 112$ bpm. For each parameter combination, we evaluated the normalized mean onset error using 1000 beats. In this experiment, we assume that the onsets of the human's performance are extracted exactly, i.e., the beat tracking works perfectly.

The error curve for various noise ratios is illustrated in Fig. 4.8. The vertical and horizontal axes denote the error and the noise ratio, respectively. The red solid line denotes the average of the baseline method of 10% noise ratio calculated from the Fig. 4.7. The plot is obtained by averaging the error for all combinations of K_r, K_h, and the tempos. Although the error increases as the noise ratio increases, it saturates after 15%. The error at the 10% noise ratio is the same situation as the second experiment shown in Fig. 4.7. The point is that, even if the noise ratio is 20%, i.e., the twice more than the second experiment, the error is still less than those of the baseline method shown as the red solid line. This suggests the

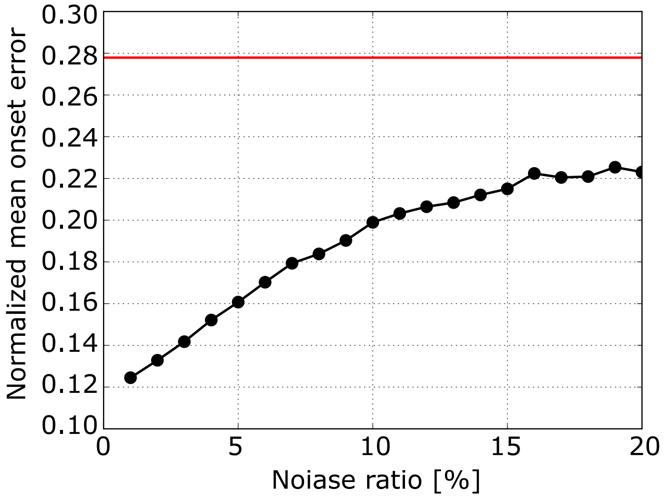
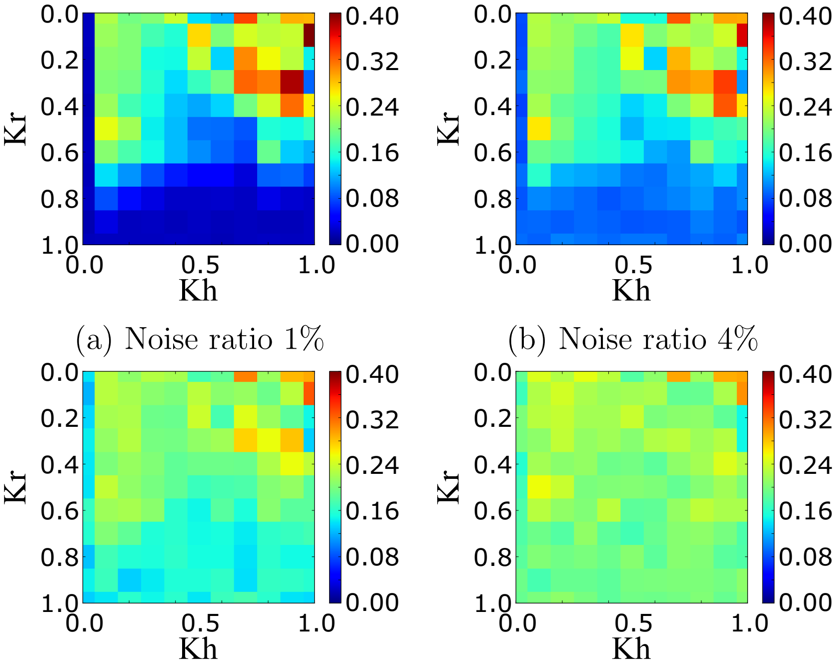


Figure 4.8: The onset error in various fluctuations

robustness of our method against fluctuating beat intervals.

Next, we discuss the effect of the noise ratio by comparing the sub figures in Fig. 4.9. For each figure, the vertical and the horizontal axes denote the robot's and human's strengths, respectively. The color denotes the normalized mean onset error as shown in the color bar. According to the result, the error is highly dependent on K_r and K_h when the noise ratio is low because the error (or color) much differs for different K_r, K_h as shown in Figs. 4.9(a) and (b). In contrast, when the noise ratio is high, the dependency decreases as shown in Figs. 4.9(c) and (d). Thus, the influence of the coupling parameters decreases as the amount of tempo fluctuation increases because the model assumes that the human's drumming is the same interval.

In the Fig. 4.9(a), the error of various coupling strengths is clearly found. If K_h is zero, the error is low for any K_r. This means that if the human do not listen to the robot and keeps its drumming, the robot easily *follows* the drumming. However, since the human listens to the robot's play and be influenced in nature, it is inevitable for K_h to have a positive value. Therefore, in a natural ensemble, no clear distinction of a leader and a follower exists because $K_r > 0$ and $K_h > 0$. In



(a) Noise ratio 1%

(b) Noise ratio 4%

(c) Noise ratio 7%

(d) Noise ratio 10%

Figure 4.9: Normalized mean onset errors in various coupling strengths.

other words, both the human and the robot (implicitly) lead each other. Another point is that, even if the K_h is positive, the best coupling strength K_r exists that minimizes the error. Therefore, if we can tune K_r from the human's play, we can achieve the synchronization in an ensemble with low onset error. This idea will be realized in Chapter 5. On the other hand, the error is high when both K_r and K_h are high. This suggests that the synchronization corrupts if the both participants try to follow the other participant too much. This matches our experience for successful demonstration: The human should keep the tempo and adjust with the robot just slightly.

4.3.4 Experiment 3: Ensemble with a Human

Finally, we evaluate we evaluate the performance of the system using a human. This experiment is carried out with three subjects. They are all non-professional

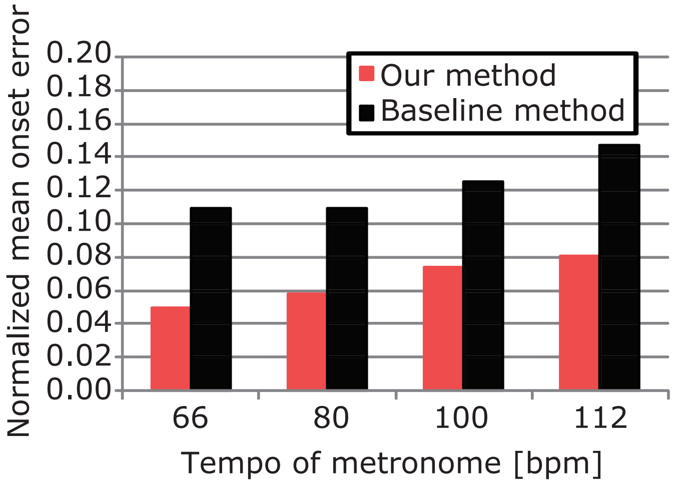


Figure 4.10: Onset errors in an ensemble with a human

Table 4.1: The error statistics in experiment with a human

Initial	Maximum [msec]		Minimum [msec]		Average [msec]	
tempo	Proposed	Baseline	Proposed	Baseline	Proposed	Baseline
66 bpm	156	223	-127	-119	45.38	99.14
80 bpm	218	214	-75	-150	43.36	82.01
100 bpm	223	219	-86	-153	44.55	75.19
112 bpm	224	224	-187	-138	43.20	78.80

drummers, males, and from 22 to 25 years old. The procedure of the experiment is as follows: Prior to each trial, the human listens to a metronome as an initial tempo. Then, he starts drumming according to the sound. After the drumming becomes stable, the robot starts playing and the metronome is stopped in order to ensure hum to interact only with the robot's playing sound.

Tab. 4.1 summarizes the maximum, minimum, and average of the mean onset error with our and baseline methods for each tempo in [msec]. The minimum and the maximum errors are equivalent for the proposed and baseline method. In contrast, the average has the significant difference. Fig. 4.10 shows the average of the normalized mean onset error. The vertical axis denotes the normalized error and the horizontal axis denotes the initial tempo. The results reveal that our method reduces the onset error by 46% on average. As the Tab. 4.1 shows, the average error of our method is almost the same for all tempos whereas that of the baseline method is increasing as the initial tempo becomes slower. However, as shown in Fig. 4.10, the normalized estimation error increases as the tempo becomes faster even for our method. This is because the tolerance of the onset prediction error becomes smaller as the tempo becomes faster. Therefore, the more accurate prediction is required for the faster tempo.

The typical errors in human are 35 msec in an orchestra [100], and 10-20 msec for stabilizing the tempo in an ensemble [101]. Compared to these results, our method thus achieves a comparable onset prediction with human in orchestra while the baseline method do not achieve. However, the prediction error is insufficient for tempo stabilization. In fact, from the experience of the experiment, the subjects tend to hit the drum faster during a trial.

4.4 Summary

We presented a two-person ensemble using a coupled oscillator model and a synchronization method using the model for a co-player robot. The model consists of the Kuramoto model an update equation of a natural frequency for tempo adaptation to the human. We implemented the ensemble system between a human drummer and a robot theremin player that predicts the human's drumming time. The experimental results revealed that our system reduce onset error more than the extrapolation based method. The results also revealed that coupling strengths should be tuned for better prediction performance.

As future work, we need to evaluate and discuss our ensemble model more strictly by comparing with the observations of a human-human ensemble since we only evaluated the onset errors of robot-human ensemble in this paper. We have also three research projects planned for the future. First, we should extend our ensemble model, for example, it may be more suitable to use a relaxation oscillator, whose oscillations emit spikes like a drum sound. Second, we need extend our ensemble system into multiple-robots and multiple-humans to evaluate our model's scalability. Third, we need to develop a visual-cue recognition system, e.g., one that can identify gestures. This is important because an ensemble involves multi-modal interactions. When we add visual information to a system, we need to consider how to integrate audio and visual cues.

Chapter 5

Multiperson Ensemble Model using Leadership Estimation

In this chapter, we extend the two-person ensemble model to the multiperson ensemble. The most important problem for multiperson ensemble is to find the leader. We present the quantified leadership metric, *leaderness*, and its estimation method. Then, we construct a state space model of the ensemble and evaluate the model using the human tapping task.

5.1 Introduction

An ensemble is a common activity to most human societies. The number of ensemble co-players can range from two to dozens, e.g., from a duo ensemble to an orchestra. Regardless of the ensemble size, humans can synchronize their playing timings with each other. This chapter aims to realize the robots to play their instruments in synchronization with the co-players in such multiperson ensembles.

The problem of the co-playing robot in a multiperson ensemble is *who to follow and when to lead*. Conventional human-robot ensemble studies avoided the problem by assuming that only one leader exists at the same time [50, 64, 102]. A tapping task, which is used to investigate human's response to rhythms in psychology, uses the assumption; the leader is the stimulus and the follower is a participant.

However, in a multiperson ensemble of humans the problem is inevitable because the leaders have two features:

1. **Multiple leaders exist**: More than two co-players sometimes lead the ensemble together. For example, if the ensemble score has two main melodies, the main-melody players will be the leaders, and the bass part and the drum part in a band will cooperate to lead the ensemble.

2. **Leaders change**: For example, if the current leader makes a mistake, another co-player will be a new leader to keep the rhythm, and when a main melody part in a score changes, the leader will change.

We present an estimation method of both of them: (1) the multiple and time-varying leaders and (2) the tempos and beat timings of all co-players in a multiperson ensemble. We define the ensemble tempos as the aggregation of the co-players' rhythms because the ensemble tempo can be interpreted as the consensus of the individual tempo among the co-players. This interpretation is analogous to the multiple person decision making (MDPM) problems [103]. It models the consensus making process among experts who have their own opinions, e.g., policy making in a political committee. The MDPM involves three steps: the experts give their opinions with preference or confidence, these opinions are aggregated into one opinion, and the experts modify their opinion to get closer to the aggregation. Many researchers have been studied this problem, e.g., aggregating different preference representations [104] and an opinion dynamics analysis [105]. The multiperson ensemble is also a MDPM problem because the rhythm synchronization in the ensemble has the similar steps: the co-players generate a tempo as an opinion, the tempo is aggregated into the ensemble tempo, and the co-players modify their tempo to get close to the ensemble tempo.

The key idea is a *leaderness* that represents the co-player's leadership, i.e., the power to influence the others. Fig. 5.1 illustrates our representation of a multiperson ensemble; each participant has an individual tempo and leaderness,

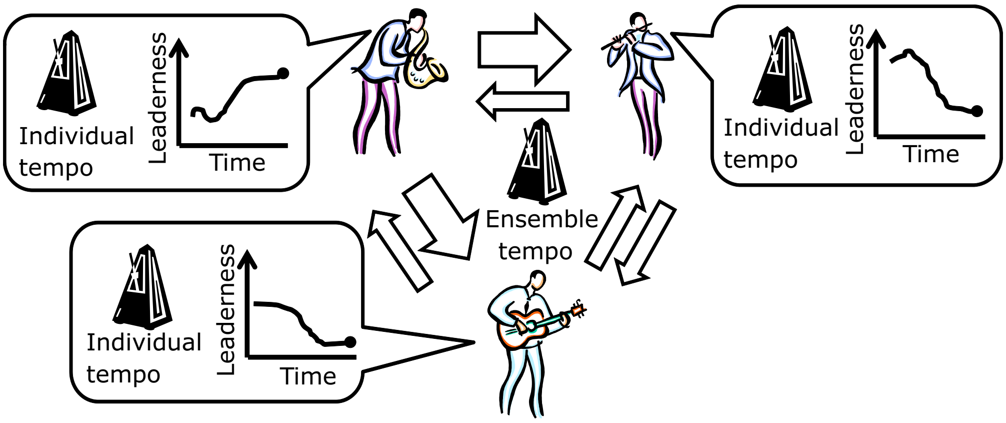


Figure 5.1: Time-varying and continuous leaderness

and the leaderness is time-varying. These tempos are aggregated into the ensemble tempo. We define the leaderness as the product of the *tempo stability* and the *tempo difference* from the ensemble tempo. The tempo stability reflects the reliability of the co-player; this metric works as a filter to select only the co-player who can keep the tempo stable. The tempo difference reflects the co-player's desire to change the tempo; this metric means that a co-player having a strong desire to change the tempo will obtain the high leaderness.

The leaderness distribution depends on a situation; if no co-players have a desire to change, they simply try to match each other. In this case, the leaderness has a uniform distribution because no leaders exist. In contrast, if a co-player tries to change the tempo, his/her leaderness becomes high, i.e., the leaderness has a sparse distribution. Then, the ensemble tempo will approach his/her tempo. From the view point of dynamical systems, the leaderness is the attractor of the tempo because the co-player's tempo converges to the ensemble tempo.

On the basis of this idea, we build a nonlinear state space model consisting of three parts: the leaderness update, the individual tempo update using the leaderness, and a coupled oscillator model for rhythm synchronization. This model has a relationship to psychological models as discussed in section 5.2. We estimate a hidden state from the observed onset timings and inter-onset intervals (IOIs)

Table 5.1: Notations

t	Time
Δt	Time interval
N	Number of co-players
M	Number of past onsets used to calculate stability
i	Index of a co-player ($i \in \{1, ..., N\}$)
$\omega_s(t)$	Ensemble tempo
$\omega_i(t)$	ith tempo
$\theta_i(t)$	ith phase (Onset is produced when $\theta_i(t) = 2n\pi$.)
$l_i(t)$	Leaderness of the ith participant ($\sum_{i=1}^{N} l_i(t) = 1$)
$s_i(t)$	Stability of the ith co-player.
$p_i(t)$	Difference from the ensemble tempo of the ith co-player.
$\mathbf{x}(t)$	State vector, $\mathbf{x}(t) \in \mathbf{R}^{1+N(M+2)}$
$\mathbf{z}(t)$	Observation vector, $\mathbf{z}(t) \in \mathbf{R}^{2N}$)

using on an unscented Kalman filter (UKF) [94].

5.2 Psychological Models of Ensembles

We introduce a psychological model of an ensemble co-player proposed by Keller [55]. The model is relevant to our model as we discuss in section 5.3.5. Briefly, his model consists of cognitive processes: anticipatory auditory imagery, prioritized integrative attention, and timing adaptation.

The anticipatory auditory imagery is a co-player's mental imagery of the ideal music, i.e., a goal image shared among all co-players. The co-player plays the music in synchronization with the other performances anticipated by the imagery. While playing, each co-player modifies the imagery by observing the others' performances. This process is also relevant to motion control in addition to the cognition; the ability to making imagery and an ability to generate a motion in synchronization with the heard sound, i.e., an ability to generate a motion, have a positive correlation [54, 106].

The prioritized integrative attention is a process to select who to pay attention by giving priority. The co-player needs to pay attention to others and him-self/herself because each co-player has to achieve both his/her successful play and synchronization with the others. Obviously, the leaders gather more attention than the followers. The point is that the prioritized integrative attention has a *priority* instead of binary selection. In other words, this is a hybrid of selective attention and non-prioritized attention models: The former selects only one leader and the latter pays attention to all the others equally.

Timing adaptation is a process for adjusting the beat timings and tempos, called a mental timekeeper. The co-players play using the timekeeper and adjust it to compensate onset timing mismatches with others. The mental timekeeper adjustment is twofold: phase and period correction [107]. The phase correction is a timing update that occurs regularly and unconsciously, whereas the period correction is a tempo update that occurs only if the participant notices an obvious tempo change. This correction occurs not only for the beat timings but also for subdivisions of beats [108].

5.3 Multiperson Ensemble Model

This section presents a model of multiperson ensemble for estimating the beat timings, tempos, and leaderness of co-players. First, we define a problem statement. Then, we build a state space model of the ensemble. Finally, we describe the state estimation using the UKF. Table 5.1 summarizes the notations.

5.3.1 Problem Statement and Model Overview

Problem statement

Input: Last onset timings and IOIs of all co-players
Output:
Leaderness of all co-players
Next onset timings and tempos of all co-players
Assumptions:
(1) *Simple ensemble*: All co-players generate onsets at all quarter notes
(2) *Cooperative ensemble*: All co-players try to synchronize with each other

The inputs are obtained using a beat tracking methods such as [34]. The outputs are for prediction; the next onset timings, tempos, and leaderness. The first assumption is for simplicity. We can avoid considering a complex beat structure and focus on the timing prediction. Note that this assumption can be relaxed by representing the beat structure using a coupled oscillator model [109]. The second assumption is typically satisfied because all co-players try to cooperate with each other for success of the ensemble.

We solve this problem by building the state space model of co-players' rhythms. The model overview is shown in Fig. 5.2. The ith co-player has two parameters: an individual tempo $\omega_i(t) > 0$ and a phase $\theta_i(t) > 0$. The ith co-player generates an onset when $\theta_i(t) = 2n\pi (n \in \mathbf{N})$. The ensemble tempo $\omega_s(t)$ is aggregated by averaging $\omega_1(t) \cdots \omega_N(t)$ using $l_i(t) \cdots l_N(t)$ as the weights. The individual tempos are updated so that they converge to $\omega_s(t)$ with a step size of $1 - l_i(t)$. The phase $\theta_i(s)$ is updated using both $\omega_i(t)$ and the phase differences from the others. $l_i(t)$ is updated using $\omega_s(t)$ and the ith co-player's past M tempos, $\omega_i(t) \cdots \omega_i(t-M-1)$.

5.3.2 State Update Model

The state update model consists of three parts; tempo, phase, and leaderness.

Tempo Update

If a co-player has a higher leaderness, his/her tempo has a greater effect on the ensemble tempo. Therefore, we design the ensemble tempo aggregation as the leaderness-weighted average of the individual tempos:

$$\omega_s(t+1) = \sum_{i=1}^{M} l_i(t)\omega_i(t) \tag{5.1}$$

The co-players adjust their tempos so that they converge to the ensemble tempo. The amount of the adjustment is less for leaders and more for followers. Therefore, we use the leaderness as the step size of the individual tempo update:

$$\omega_i(t+1) = \omega_i(t) + (1 - l_i(t))(\omega_s(t) - \omega_i(t)) \tag{5.2}$$

Note that $1 - l_i(t)$ means the rest of the leaderness, in other words, the leaderness owned by the other co-players since the total of them is one.

This calculation is similar the ordered weighted average in a consensus model [103], which is one of a MDPM models. The opinions $\omega_1(t) \cdots \omega_N(t)$ are generated by N co-players with the preference or confidence $l_1(t) \cdots l_N(t)$ they are aggregated into one opinion $\omega_s(t+1)$ in Eq. 5.1, and opinions are updated using the aggregation in Eq. 5.2.

Phase Update

We design the beat generation using a coupled oscillator model as we described in Chapter 4. The coupling oscillator model consists of phases, natural frequencies, and coupling strengths. The natural frequencies and coupling strengths correspond to the tempo and the leaderness, respectively. The phase is updated using its tempo and the phase difference from the other oscillators measured by a 2π periodic function called the coupling function. We use a sinusoidal function as the coupling

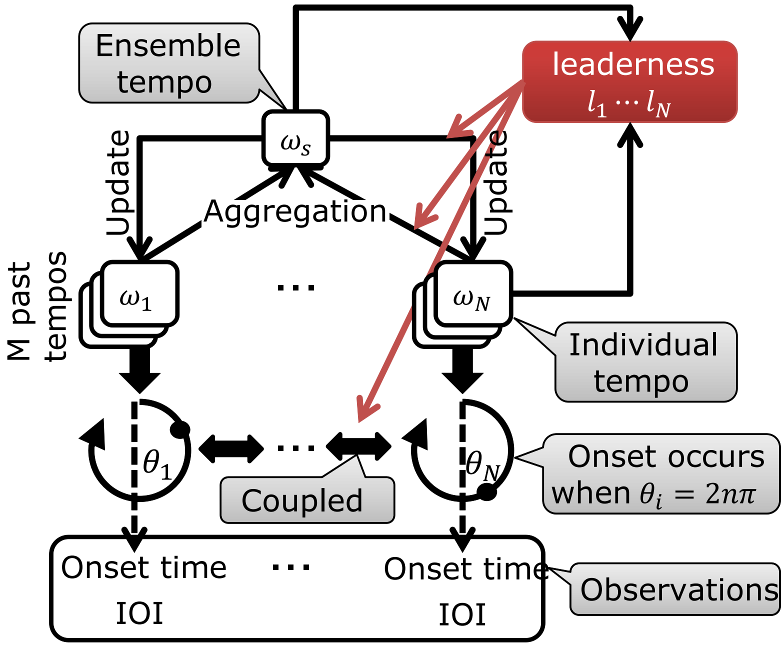


Figure 5.2: Overview of multiperson ensemble model

function, known as Kuramoto model [2]. The phase update is formalized as:

$$\theta_i(t+1) = \theta_i(t) + \omega_i(t)\Delta t + \sum_{j=1}^{N} l_j(t)\sin(\theta_j(t) - \theta_i(t)) \tag{5.3}$$

Leaderness

We design the leaderness as the product of the stability and the difference from the current ensemble tempo. This is because the leader should have both a desire to change the tempo and an ability to keep it. Without the former, the model is incapable of representing the tempo dynamics. Without the latter, the model is incapable of ignoring the perturbations caused by mistake or unreliable players.

The stability, $s_i(t)$, is defined as the exponential of the standard deviation (std) of past M tempos including the current tempo $\omega_i(t)$:

$$s_i(t) = \exp\left(-Std[\omega_i(t), \cdots \omega_i(t-M-1)]\right) \tag{5.4}$$

where $Std[\cdot]$ denotes an operator to calculate the std. Instead of directly using the std, we use an exponential because we need to limit the range of $s_i(t)$ in $[0,1]$. Thus, $s_i(t)$ becomes maximum if the past M tempos are exactly the same and becomes less if the tempo fluctuates more.

The difference from the tempo, $p_i(t)$, is defined as the shifted-sigmoid function of the absolute difference:

$$p_i(t) = \frac{3}{2}\left(\frac{1}{1+\exp\left(-\left|\omega_i(t)-\omega_s(t)\right|\right)} - \frac{1}{2}\right) \tag{5.5}$$

The sigmoid-like function is used to ensure that $p_i(t) \in [0,1]$.

Then, the leaderness is updated as the multiplication of them:

$$l_i(t+1) = s_i(t)p_i(t) \tag{5.6}$$

Note that we need not considering the scales of them because the both values have the same scale. Finally, $l_i(t+1)$ is normalized to satisfy $\sum_{i=1}^{N} l_i(t+1) = 1$.

The leaderness determines the attractor of the individual tempos because they converge to the ensemble tempo (Eq. (5.2)) and the ensemble tempo is calculated (Eq. (5.1)). If the leaderness remains stable, the individual tempos converge to the ensemble rhythm. Therefore, our definition can be paraphrased that the leaders are the co-players who change the tempo attractor.

State Vector

In summary, the state vector $\mathbf{x}(t)$ is the vector with $i + N(M+2)$ dimension:

$$\begin{aligned}\mathbf{x}(t) &= (\omega_s(t), \omega_1(t), ..., \omega_N(t), \\ &\quad \omega_1(t-1), ..., \omega_N(t-M-1), \\ &\quad l_1(t), ..., l_N(t), \theta_1(t), ..., \theta_N(t))^T \end{aligned} \tag{5.7}$$

The breakdown of the dimension is as follows: the ensemble tempo (1 dim., Eq. (5.1)), the individual tempo (N dim., Eq. (5.2)), the individual phase (N dim., Eq. (5.3)), the leaderness (N dim., Eq. (5.6)), and the past $M-1$ tempos ($N(M-1)$ dim.).

5.3.3 Observation Model

In consideration of the problem statement in section 5.3.1, the system can observe the co-players' tempo and onset timings only at their onset timings. Therefore, the observation occurs partially.

We now design the observation model. Let the ith ($i = 1, .., N$) element of $\mathbf{z}(t)$ be $z_i(t)$. The first half elements of $\mathbf{z}$, $z_1(t) \cdots z_N(t)$, are assigned to the tempos and the rest of them, $z_{N+1}(t) \cdots z_{2N}(t)$, are assigned to phases. When an onset from ith co-player is not observed, we substitute $\emptyset$ to z_i and z_{i+N} to indicate no observation. When it is observed, we obtain two kinds of information; the tempo ($z_i(t)$) and the phase ($z_{i+N}(t)$) being zero at the time.

Therefore, the observation model is ($i = 1 \cdots N$):

$$z_i(t) = \begin{cases} \omega_i(t) & \text{if } \theta_i(t) = 2n\pi \\ \emptyset & \text{otherwise} \end{cases} \tag{5.8}$$

$$z_{i+N}(t) = \begin{cases} 0 & \text{if } \theta_i(t) = 2n\pi \\ \emptyset & \text{otherwise} \end{cases} \tag{5.9}$$

5.3.4 Unscented Kalman Filter for State Estimation

We estimate the state $\mathbf{x}(t)$ using an UKF [94] because both the state update and observation models are nonlinear. The UKF utilizes an unscented transform; it estimates the mean and covariance after any nonlinear transformation from given mean and covariance before the transformation using deterministically selected samples. The UKF approximates the nonlinear state space model at least second order. Note that although no noise terms are included in sections 5.3.2 and 5.3.3, we consider additive Gaussian noises which are the design parameters of the UKF.

The observation vector can have $\emptyset$ because the observation is obtained partially. In that case, we skip the state estimation update by an observation, i.e., the innovation is assumed to be zero.

The robot control is possible from the estimated state. Let the robot be the jth participant. If the UKF successfully estimate the hidden state, the robot can predict when it should generate an onset from $\omega_j(t)$ and t. The robot can play an instrument based on this prediction.

5.3.5 Relationship to Psychological Model

We discuss the relationship between our model and the three parts of Keller's model introduced in section 5.2.

The anticipatory auditory imagery, which is used for anticipation and is adjusted during the ensemble, corresponds to the individual and ensemble tempos, $\omega_i(t)$ and $\omega_s(t)$. The reason is twofold: (1) $\omega_i(t)$ is used for anticipating the next onset time. The robot can predict the ith co-player's next onset time using $\omega_i(t)$ and the last onset time by extrapolation. (2) The ensemble tempo $\omega_s(t)$ behaves as the shared goal of the co-players because all $\omega_i(t)$s converge to $\omega_s(t)$. As shown in Eq. (5.2), the co-players adjust the individual tempo $\omega_i(t)$ so that it converges to $\omega_s(t)$.

The prioritized integrative attention corresponds to the leaderness $l_i(t)$. A participant with a higher leaderness receives more attention because $l_i(t)$ determines the influence of the ith co-player as shown in Eqs. (5.1), (5.2), and (5.3). In addition, $l_i(t)$ is a generalization of selective and non-prioritized attention models; if we add a constraint on the leaderness to only one co-player's leaderness is one and the others are zero, the leaderness represents the selective attention model. In contrast, if we add a constraint that the leaderness is $l_i(t) = 1/N$, the leaderness represents the non-prioritized attention model.

The timing adaptation corresponds to the coupled oscillator model in Eqs. (5.2) and (5.3). Eq. (5.3) represents the phase correction, and Eq. (5.2) represents the period correction.

5.4 Simulation Experiment

In the simulation experiment, we evaluate three aspects of our model and estimation method: (1) the convergent IOIs for various initial tempos, (2) the convergent IOIs for various numbers of co-players, and (3) the onset estimation error for various numbers of co-players, i.e., the ensemble size. The first two experiments are designed for revealing the model limitation, and the third one is designed for evaluating the robustness against the ensemble size.

5.4.1 Configuration

For all experiments, the time step Δt is 0.05sec, the simulation is performed for 40sec, and the initial leadernesses are the same, $l_i(0) = 1/N$. For the first experiment, we set N to 3, and the initial IOIs $\omega_1(0), \omega_2(0), \omega_3(0)$ to all combinations 1.0, 0.75, 0.6, 0.5sec corresponding to 60, 80, 100, 120bpm, respectively. In total, the number of conditions is $4^3 = 64$. For the second experiment, we set N from 2 to 21. The initial IOI $\omega_i(0)$ is set to $(60+60i/N)$bpm. In other words, the initial IOIs are equally spaced between 0.5 and 1.0sec. For the third experiment, we use the same conditions as the second experiment. The number of past tempos M is set to 10, and the state update and observation noise matrices are the diagonal matrices, $diag(0.05, \cdots, 0.05)$. The evaluation criterion for the third experiment, the onset error, is defined as the mean of the absolute difference between the nearest onset timings. Let $o_A^{(i)}$ and $o_B^{(j)}$ be the ith and jth onset times of the co-players A and B. Let L be the number of A's onsets. Then, the onset error e_{AB} is defined as

$$e_{AB} = \frac{1}{L} \sum_{i=1}^{L} \min_j \left| o_A^{(i)} - o_B^{(j)} \right| \tag{5.10}$$

Finally, the onset errors of all co-player combinations are averaged.

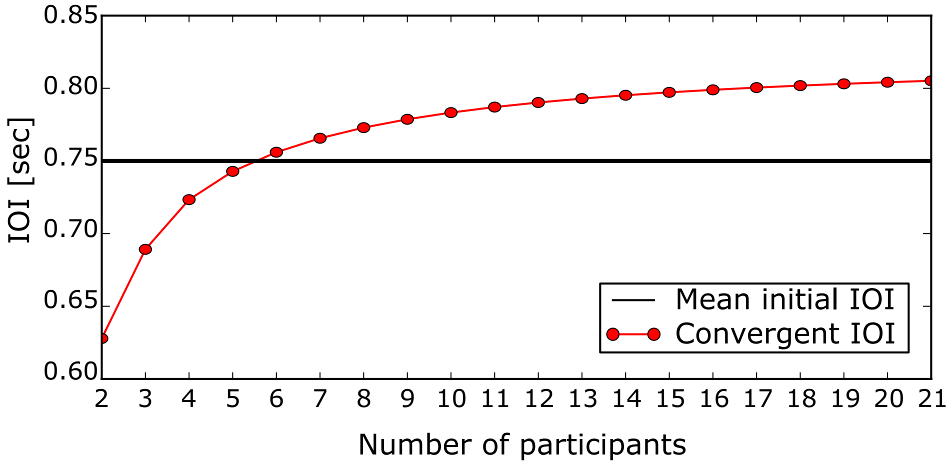


Figure 5.3: IOIs for various numbers of participants: the black and red lines denote the mean of initial IOIs and the convergent IOIs.

5.4.2 Results and Discussion

The result of the first experiment is the convergent IOI of various initial tempos. For all combinations, three IOIs, $\omega_1(t), \omega_2(t), \omega_3(t)$ are converged to the same IOI after less than 500msec. This shows that our state update model is stable. The mean difference between the mean initial IOIs, $\sum_{i=1}^{3} \omega_i(0)/3$, and the convergent IOI is 16msec with the std 25msec. Because the difference is small, the mean of initial IOIs may be used as the predictor of the convergent IOI. This intuition is evaluated by the next experiment.

The result of the second experiment is the convergent IOIs for various ensemble sizes. As shown in Fig. 5.3, the convergent IOI is less than the mean of initial IOI if the ensemble size is small ($N < 5$). In contrast, the difference increases as the size increases. In other words, the convergent tempo becomes slower as the ensemble size becomes larger. The intuitive understanding of the result is that they need to slow down the tempo to synchronize a large ensemble.

The third result is obtained by the absolute onset estimation error. The mean onset error is 120msec, and the std is 36msec. The median of them is 80msec, and the std is 40msec. This result that the mean is larger than the median suggests

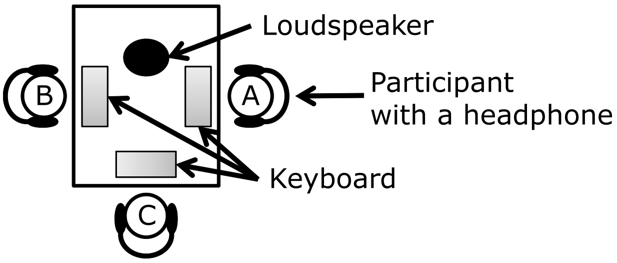


Figure 5.4: Configuration of the multiperson tapping task

that the onset prediction succeeds mainly, but it fails occasionally.

5.5 Multiperson Tapping Experiment

The target of this experiment is humans. We analyze the human's behavior using a multiperson tapping task and evaluate the onset estimation errors.

5.5.1 Configuration

We collected nine participants without any motor or sensory impairment ranging in age from 21 to 38 (8 men and 1 woman). From them, we randomly formed two kinds of groups: four pairs (named A and B) and three triads (named A, B, and C). For each group, we gave a keyboard and a headphone to each participant and asked to sit on a chair. When a participant taps a key of the keyboard, a sound is played through a loudspeaker. The sounds are pure tones of 880, 440, and 220Hz for participants A, B, and C, respectively. The positions of participants and equipment are shown in Fig. 5.4. Note that the position C was empty for the pair experiment.

We gave four instructions: (1) Initial tapping tempo is given through the headphone at the beginning of the trial. When a starting cue is given from the headphone, they start tapping at the tempo. (2) Tap the key and synchronously with others. (3) Close your eyes and only use eyes. (4) When a metronome sound is given through the headphone, follow it and ignore the others. The first instruction

Table 5.2: Stimulus for multiperson tapping experiment

Pair tapping

	Pre-stimulus			Main stimulus				
A	60	sil	cue	sil	sil	80	sil	cue
B	80	sil	cue	sil	50	sil	sil	cue
Duration [sec]	25	5	0.1	25	25	25	25	0.1

Triad tapping

	Pre-stimulus			Main stimulus					
A	50	sil	cue	sil	80	sil	sil	sil	cue
B	60	sil	cue	sil	sil	50	sil	sil	cue
C	80	sil	cue	sil	sil	sil	80	sil	cue
Duration [sec]	25	5	0.1	25	25	25	25	25	0.1

is to force the timing correction through interaction. The second instruction is to satisfy the cooperative ensemble assumption. The third instruction is to avoid the audio visual integration since our model ignores visual information. The fourth instruction is to control the dynamic change of the leaders.

The stimuli are summarized in Tab. 5.2. The top table is for pairs, and the bottom table is for triads. The column starting A, B, and C are for the stimulus for participants A, B, and C, respectively. The values in the duration row denote the duration of parts. The number denotes the metronome tempo in bpm given through a headphone, the *cue* denotes the 0.1sec, 880Hz pure tone, and the *sil* denotes the silence. The total durations are 115.2sec and 140.2sec for pairs and triads, respectively. The cue in pre-stimulus means the starting cue and that in main stimulus means the ending cue; the participants start tapping with the starting cue and ends with the ending cue. The stimuli consist of two parts: pre-stimulus and main stimulus. The former is before the tapping starts, and the latter is during the taping. We generate the stimulus in advance and gave to the participants using a multichannel audio player, Roland UA-101. We expect that a participant given a metronome obtains a high leaderness. For each pair or triad, we asked to do four trials, one for a practice and three for experiments.

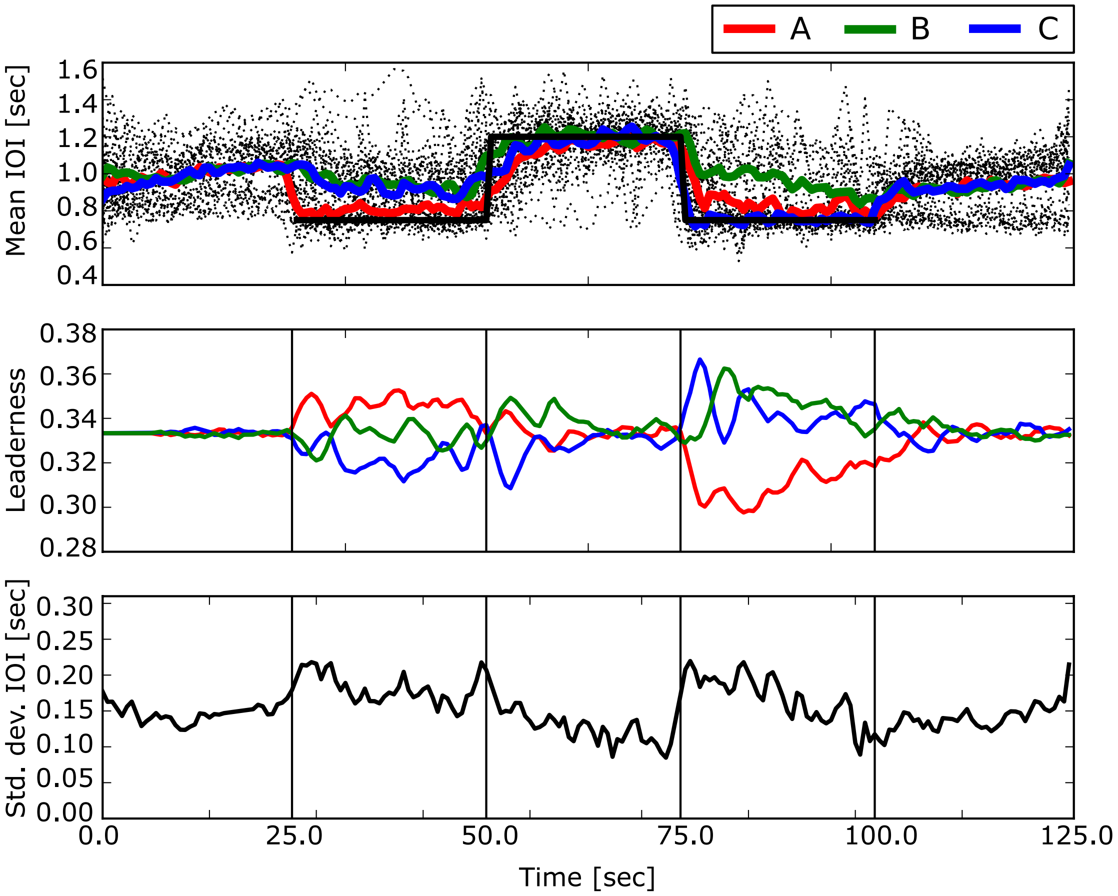


Figure 5.5: Multiperson tapping result of triads

5.5.2 Results and Discussion

Figs. 5.5 and 5.6 show the result of triad tapping and pair tapping, respectively. The figures have the same structure; the top figure shows the IOI trajectories. The black solid line denotes the given tempo. The red, green, blue lines denote the mean trajectories of A, B, and C, respectively. The dotted black lines denote the IOI trajectories of all trials. The middle figure shows the leaderness of all participants. The leaderness is smoothed by a moving average with window size 5 to reveal the trend. The color assignment is the same as the top. The bottom figure shows the std of all trajectories. The vertical axis of all figures denotes the time,

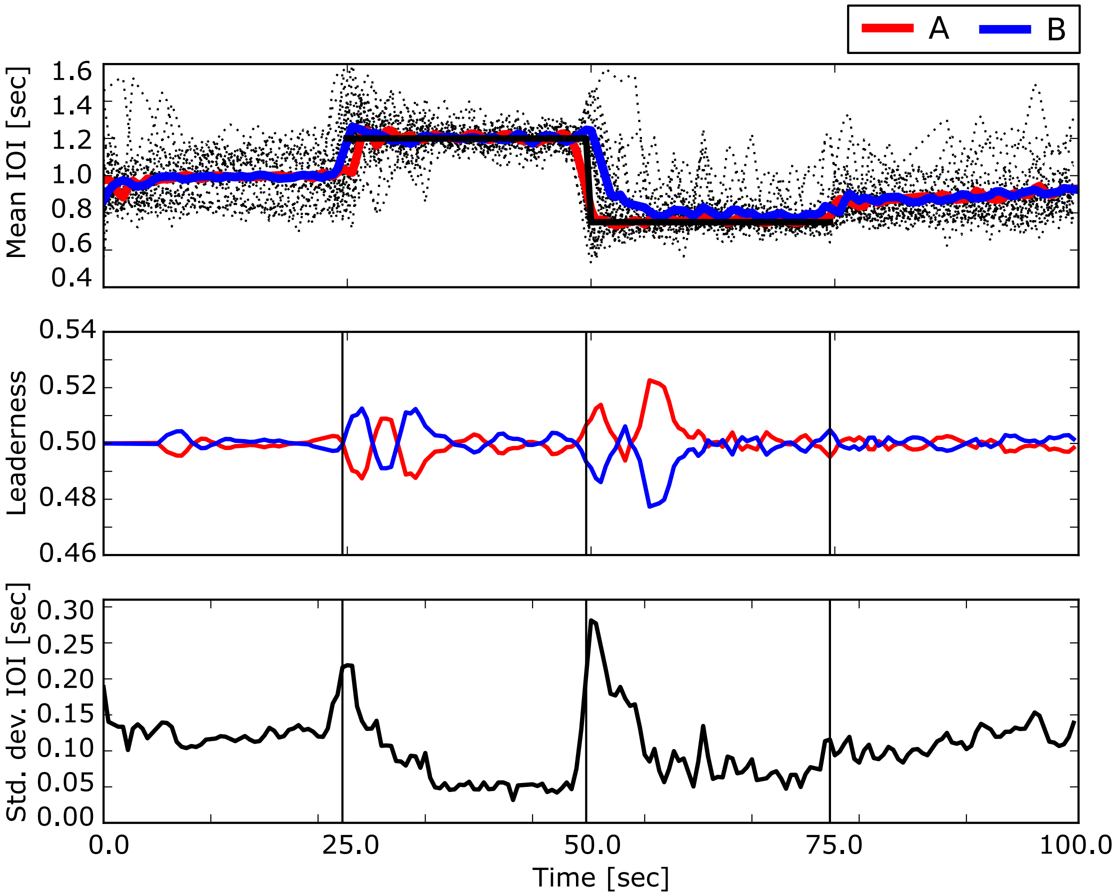


Figure 5.6: Multiperson tapping result of pairs

and the horizontal axis of the top, middle, and bottom denote IOI, leaderness, and std of IOI, respectively. Fig. 5.5 is divided into five parts on the basis of the given tempo: silence, 0.75sec (80bpm) to A, 1.2sec (50bpm) to B, 0.75sec (80bpm) to C, and silence. Similarly, Fig. 5.6 is divided into four parts: silence, 1.2sec (50bpm) to B, 0.75sec (80bpm) to A, and silence. For implementation, the initial values of leadernesses are equal, and start updating after observing M onset because the stability calculation requires M samples.

IOI trajectories

We discuss the top of the figures, the IOI trajectories. We can confirm that the participants achieved the task well from the mean trajectories of the participants (colored solid lines) and the target IOI (black solid lines) of both figures; this is because when the target IOI is given to a participant, his/her IOI quickly approaches the target and then the others approach later.

From the first and the last part of all trials (dotted lines) of both figures, we can confirm that the convergent IOI depends on the initial tempo as we hypothesized in the simulation experiment. This is because the dotted lines are widely spread in the first part, whereas they spread narrower in the last one. This difference is caused by the initial tempos; they are different for the first part, whereas they are similar for the last; this is because the IOIs are largely different in the beginning, whereas they are similar after the interaction in the past interaction.

As shown in the last part in Fig. 5.5, one triad keeps the previous IOI (around 0.7 sec), whereas those of the others are increasing. This suggests that the former group has higher capability of tempo keeping than other groups. This behavior difference depending on the participants capability reveals the limitation of the model; we assume that the co-players are identical and ignore the individual dependence.

Leaderness Dynamics

We describe the dynamics of the leaderness from the middle of the figures. First, we focus on Fig. 5.5. In the first and the last part, the leaderness is almost uniformly distributed meaning that the ensemble has no leaders. In the second part, the leaderness of A is the highest and those of B and C are low. Thus, A is the leader, and B and C are the followers in the part. In the third part, the B becomes the leader in the first half, then the explicit leader disappears in the last half. The reason is that the tempos are converged as shown around 60sec in the top figure. In the fourth part, the A and C become the follower and the leader, respectively. However, the B still have a high leaderness because B keeps a tempo far from $\omega_s(t)$.

Next, we focus on Fig. 5.6. The leaderness is uniformly distributed in the first and the last part, similar to Fig. 5.5. In the beginning of the second and the third part, the leaderness is biased to the participant given a target tempo. In this case, the bias quickly disappears because the IOIs of the participants are converged quickly. This indicates that the pair tapping task is easier to synchronize than the triad.

Standard Deviation

The bottom of Figs. 5.5 and 5.6 shows the tempo convergence. The std of IOIs is high at the beginning of all parts, then, it decreases over time. This indicates that the IOIs converge to the ensemble tempo, as in Eq. (5.2).

Onset Errors and IOI Errors

The onset errors among participants are summarized in Table 5.3. The onset error definition is the same as Eq. (5.10) for the pair tapping and the triad tapping. Comparing the average errors of them, we find that the triad task is more difficult because the error of the triad is larger than the pair. The errors are especially large when the target IOI is slow. This is because the motion becomes less rhythmic

Table 5.3: Onset errors in multiperson tapping task

	Pair tapping		Triad tapping		
Trial	Mean	Std	Mean	Std	
1	123	53	247	60	
2	147	101	230	56	
3	188	50	204	66	
Average	153	68	227	61	[msec]

Table 5.4: Mean and median of IOI and onset estimation errors

	IOI error		Onset error		
Condition	Median	Mean	Median	Mean	
Pair tapping	33	63	181	204	
Triad taping	110	154	241	246	
Average	72	109	211	225	[msec]

in the slower tempo. Therefore, predicting the onset timings in the slow tempo is more difficult than in the high tempo.

The error increases as we conduct more trials for the pair tapping; it decreases for the triad tapping. Because the amount of the participants and the trails are insufficient, we need to increase them to obtain a reliable result of this learning effect.

The IOI and onset prediction errors are summarized in Table 5.4. The IOI error is the mean absolute error, and the onset error is the same as Eq. (5.10). The median IOI errors are 33msec for the pairs, and 110msec for the triads. The error in triad tapping is equivalent to humans. The median onset errors are 181msec for the pairs, and 241msec for the triads, i.e., a co-player robot controlled by our method will have these onset errors. Recalling that the onset errors among humans were 153msec for the pairs and 227msec for the triads, the robot can play to a comparable accuracy with humans.

5.6 Summary

This chapter presented a state space model of a multiperson ensemble and its estimation of onset timings, tempos, and leadership. Our model has a relationship to psychological model of a co-player in ensemble. The main contribution is that we quantified the multiple and time-varying leaders in an ensemble using the leaderness. The simulation experiment showed that our model was stable for a wide range of ensemble size and initial tempos. The multiperson tapping task showed that the leaderness can capture the changing leadership, and out method can predict the onset timings equivalent accuracy to humans.

We have three future plans; implementing our method on real human robot ensemble, integrating the visual information, i.e., modify the observation model, and introducing hierarchical oscillators to represent a complex rhythm.

Chapter 6

Sound Imaging: Spatio-Temporal Analysis of Frog Choruses

This chapter discusses the second case study of frog choruses. In this case, measurement itself is difficult because the frogs call at night in held. We present the measurement and analysis system of the frog choruses. The key idea is to *see the sound*. By converting the calls to the lighting pattern using the novel device *Firefly*, we visualize the frog chorus. Experimental results show the characteristics of the system and the synchronization state in the frog choruses.

6.1 Introduction

Spatio-temporal structure of animal choruses is an important clue to understand the acoustic communication [5, 65]. They call for various purposes, for example, mating, territory maintenance, and localizing preys. Because the acoustic communication is invisible, investigation methodology limits the information that the researchers can obtain. Especially for small nocturnal animals such as frogs and crickets, observing the communication is difficult because experiment must be held in the dark, and they stop calling if the researchers get too close to them. Here, we define the *chorus* as the set of many individuals call at the same time place. Therefore, a general-purpose observation method should satisfy the following requirements:

1. to work in animals' habitat,

2. to detect multiple and simultaneous calls automatically,

3. to minimize the influence on the animal behavior, and

4. to work with various spatial distributions of animals.

We present an inexpensive and easy-to-use *sound imaging system* for bioacoustic investigations. The system consists of dozens of sound-to-light converting devices named *Firefly* and an off-the-shelf video camera. The Firefly is composed of a microphone, an amplifier, and a light-emitting diode (LED). It converts the nearby sounds captured by the microphone into the light of the LED. The sound imaging system follows the procedure: Fireflies are placed on the ground around the sound sources, and their light emissions are recorded with a video camera. The recorded emissions are analyzed to obtain a visualization of the animal spatio-temporal calling behaviors. This visualization enables us to estimate when and from where the animals called. The system works in the three steps: deploy Fireflies, record their light intensities using an off-the-shelf video camera, and analyze the video. The system satisfies all the four requirements because (1) it visualizes the time and place of each animal call in their habitat, (2) Fireflies at different locations respond to different calls because each one responds to only nearby sounds, (3) observers do nothing once starting the recording, and (4) the system can be used for different species by changing the number of Fireflies and inter-Firefly distances.

This chapter is organized as follows: In 6.2, details of hardware of Firefly, procedure of experiments, and analysis method are described. In 6.3, we conduct the detailed analysis of Firefly, simulation experiment to reveal the limitation of the system, and frog chorus visualization experiments in indoor and fields.

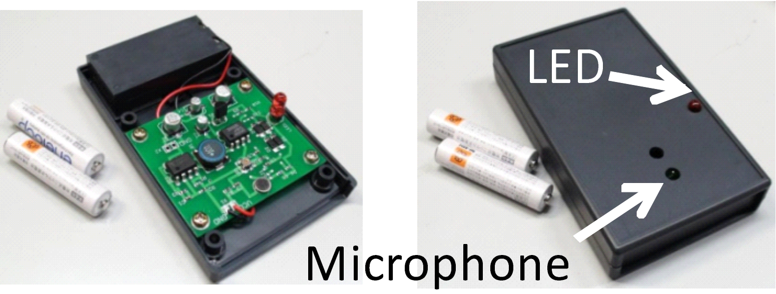


Figure 6.1: Pictures of Firefly

6.2 Materials and Methods

6.2.1 Data Acquisition Equipment

The sound imaging system consists of (1) an off-the-shelf video camera and peripheral equipment and (2) a set of sound-to-light converting devices called Fireflies.

We use a HandyCam video camera (Sony, HDR-XR520; 29.97 fps) with a wide angle (0.7x) conversion lens (Sony, VCL-HGA07) on a tripod (Sony, VCT-80AV). The video camera is used with a high definition mode (1480×1080 resolution). The wide conversion lens is used to widen the range 1.4 times. The tripod is used to position the camera in a fixed location.

A Firefly mainly consists of a microphone, a variable resistor, a light emitting diode (LED), and a plastic box (LM-100C, Takachi Electronics, Enclosure, Co. Ltd). It requires two AAA batteries. When a Firefly detects a sound nearby, it drives the LED to indicate the existence of the sound. Figure 6.1 shows the picture of a Firefly. The left picture depicts a cover-opened Firefly for depicting circuit implementation, and the right picture depicts Firefly used in experiments. Figure 6.2 shows the top-view schematic drawing of a Firefly shielded by a plastic box.

The block diagram of a Firefly is shown in Figure 6.3. The circuit consists of four modules: a power source, a microphone, an amplifier, and a sound-to-light conversion. The power source module generates two voltages, 2.4V and 5V, for other modules. The microphone module, which consists of an electret condenser

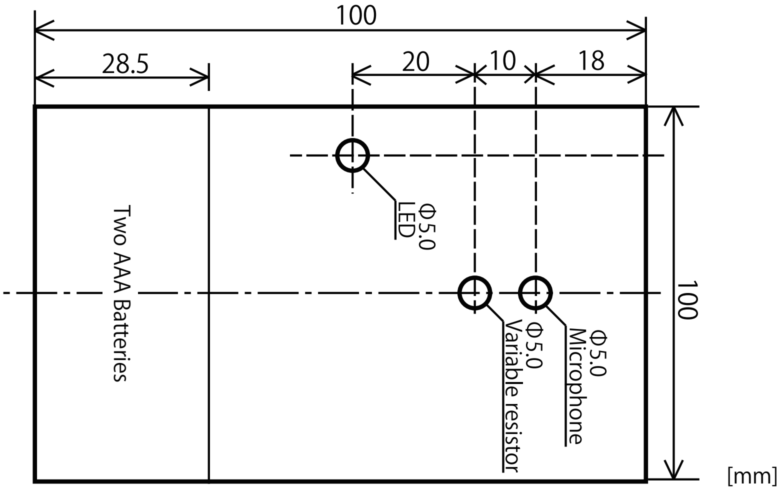


Figure 6.2: Schematic drawing of a Firefly

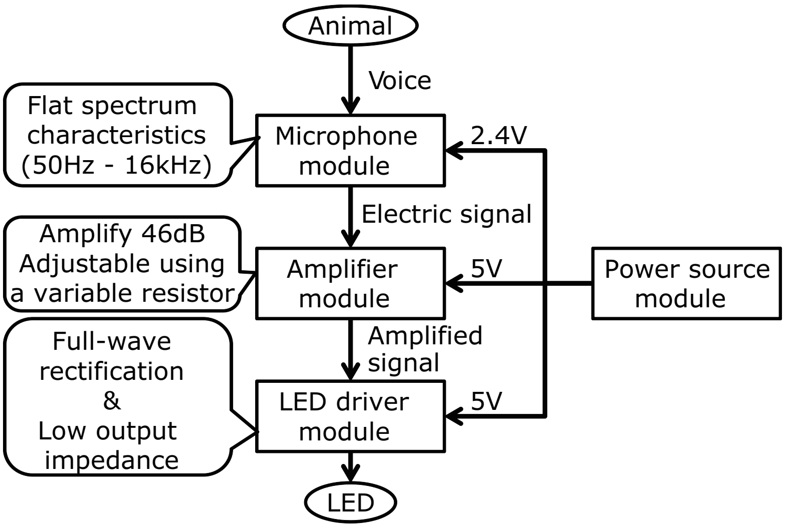

Animal
Voice
Flat spectrum characteristics (50Hz - 16kHz)
Microphone module
2.4V
Electric signal
Amplify 46dB Adjustable using a variable resistor
Amplifier module
5V
Power source module
Full-wave rectification & Low output impedance
Amplified signal
LED driver module
5V
LED

Figure 6.3: Block diagram of a Firefly

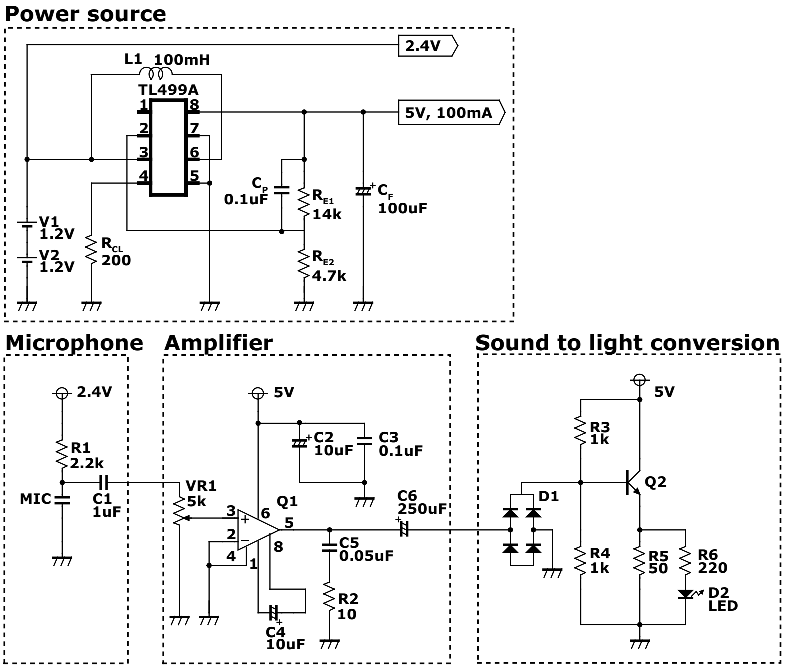


Figure 6.4: Circuit diagram of Firefly

microphone and its peripheral circuit, captures nearby sound. Then, the amplifier module increases the amplitude of the sound's waveform by 46dB in maximum. This gain can be adjusted using a variable resistor. Finally, the sound-to-light conversion module drives the LED using the rectified waveform.

The circuit diagram of a Firefly is depicted in Figure , and its components list is in Table 6.1. The main specifications of the electronic components are as follows. The batteries (two AAA batteries, each of which is 1.2V) provide the power source of 2.4V, and the power supply controlling integrated circuit, TL499A, generates 5.0V from the battery voltage. The microphone is omnidirectional, WM-E13UY by Panasonic, which has almost flat frequency response from 50 Hz to 15k Hz. For the amplifier module, an operational amplifier integrated circuit, NJM386, is used with the 46dB gain. A red LED, OSR5PA5201A by OptoSuuply, typically 40,000mcd, is used as an indicator of the sound existence. The red color is selected based on the observation that photopositive animals are insensitive to the red color [110].

Table 6.1: List of components for Firefly

Component description	Symbol in Fig. 6.2.1
200Ω resistor	R_{CL}
14kΩ resistor	R_{E1}
4.7kΩ resistor	R_{E2}
100mH inductor	$L1$
100μF electrolytic capacitor	C_F
0.1μF ceramic capacitor	C_P
Panasonic Eneloop AAA rechargable battery	$V1, V2$
TL499A, power-supply controller IC	$TL499A$
2.2kΩ resistor	$R1$
1μF ceramic capacitor	$C1$
WM-E13UY, electret microphone	MIC
10Ω resistor	$R2$
5kΩ variable resistor	$VR1$
10μF electrolytic capacitor	$C2$
0.1μF ceramic capacitor	$C3$
10μF electrolytic capacitor	$C4$
0.05μF ceramic capacitor	$C5$
250μF electrolytic capacitor	$C6$
NJM386, amplifier IC	$Q1$
1kΩ resistor	$R3$
1kΩ resistor	$R4$
50Ω resistor	$R5$
220Ω resistor	$R6$
2SC1815, NPN Transistor	$Q2$
Diode bridge	$D1$
OSR5PA5201A, Red LED	$D2$

Because Japanese tree frogs are photopositive in our experience, we assumed that a red LED has less effect than other colors. A linear taper variable resistor is used to adjust the sound-pressure-level to light-intensity characteristics of a Firefly. Here, the word *adjust* means that rotating the angle of the variable resistor manually to maximize the dynamic range of the light intensity in response to a nearby sound. The resistor can be rotated from 0 to 270°, corresponding to 0 to 5k Ω. The effect of the rotation is measured experimentally later. All the resistors and capacitors are 5% tolerance.

The LED is covered with a silicon cap as a light diffuser. It is made of white rubber, and the dimension is the same as LED. Without the cap, the light intensity is less when its lighting pattern is captured from the side [111] because of the LED's directivity. The cap reduces its directivity by scattering the LED light. The effect of the cap on sound-pressure-level to light-intensity characteristics is measured later.

All sound pressure levels are recorded using a sound level meter NL-32 by RION, Co., Ltd. We used it with the flat response curve mode and the fast mode which has the shortest time constant. The shortest time constant is selected because the frogs have short note durations of the advertisement calls.

6.2.2 Data Acquisition Procedure

The data acquisition procedure comprises of two steps: deploy the Fireflies on the field and record them with the video camera. To use our system in light rain, we wrap the video camera in cling film or place in a water-resistant housing (Sony, SPK-HCG).

The data acquisition for field experiments are performed taking care to the following points: (1) the Fireflies are deployed along the edge of a rice paddy where Japanese tree frogs call, (2) the video camera is fixed on a tripod, and (3) all the experiments were performed after sunset since they call after sunset.

We can assume that the frogs call on the edge of the paddy field. The stiffness of the rice stalks is insufficient to cling in the frogs' mating season because the rice has not matured yet. Therefore, they cannot call inside of the paddy field.

6.2.3 Data Analysis

Visualizing the light pattern of Fireflies requires two functions: (1) detecting a weak light emitted by each small device (2) eliminating the individual differences among devices. The key ideas to fulfill these requirements are (1) increasing the contrast between the emitted lights and the background using frame averaging and (2) subtracting its mean from the light intensity time series for each light. The procedure is described as follows.

The data analysis comprises of six steps: (1) split a video sequence into frames, (2) calculate the mean of the frames, (3) detect the LEDs in the frame, (4) generate a mask covering the LEDs, (5) extract time series of the light intensities, and (6) detect the times and places of calls.

(1) Video Split

We divide the video file into individual frames using a movie processing tool, TMPGEnc by Pegasys Inc. Open source software (`ffmpeg`) can also be used for this process.

(2) Mean Frame Calculation

We convert the frames into gray-scaled frames since we are interested in the frame intensity. The conversion is based on the equation for calculating the brightness of a color pixel defined by the National Television System Committee (NTSC) [112]. Let $I_g(x, y, t)$, $I_R(x, y, t)$, $I_G(x, y, t)$, and $I_B(x, y, t)$ be the brightness and the red, green, and blue components of pixel (x, y) in the tth frame. The equation is

$$I_g(x, y, t) = 0.2989 I_R(x, y, t) + 0.5870 I_G(x, y, t) + 0.1140 I_B(x, y, t) \tag{6.1}$$

where $I_g(x, y, t) \in [0, 255]$.

We calculate the mean of the frames using

$$\bar{I}(x,y) = \sum_{t=1}^{N} I_g(x,y,t) \tag{6.2}$$

(3) LEDs Detection

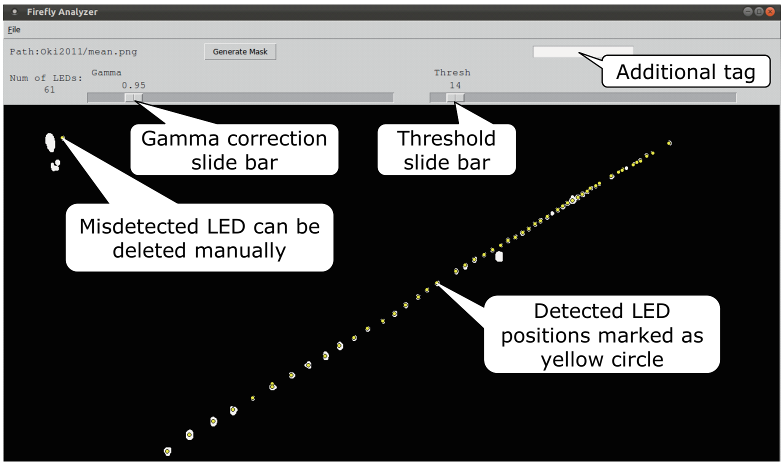
We binarize the $\bar{I}(x,y)$ to detect the LED areas. The discriminant analysis based binarization [113], which is the most popular binarization method, cannot be used because it assumes that the histogram of pixels of an image is bimodal. However, our data have almost unimodal histogram; a majority of small values corresponding to the background area while a minority of large values corresponding to the LEDs area. This step consists of an automatic LED detection considering this characteristics, and manual LED position correction using a newly developed a graphical user interface (GUI)-tool.

For automatic LED detection, we first estimate the threshold θ to distinguish the LEDs and the background areas. The algorithm has two steps: set the initial value of θ with the peak of the histogram, then increase θ as long as the number of occurrences decreases. The formal description is the following: Let $I_{mean}(x,y) \in [0,255]$ be a pixel value at (x,y) in the mean frame, and $h(i)$ be the number of pixels whose value is i defined as $h(i) = |\{I_{mean}(x,y)|I_{mean}(x,y) = i\}|$ where $|A|$ denotes the number of elements in the set A. The algorithm is as follows:

```
$\theta \leftarrow \underset{i}{\text{argmax}}\ h(i)$
**while** $h(\theta) > h(\theta+1)$ and $\theta < 255$ **do**
    $\theta \leftarrow \theta + 1$
**end while**
```

Mean frame $\bar{I}$ is binarized using threshold θ.

Next, we assign a label to each four-connected component in the binary frame using `scipy.ndimage.label` in the Python library `scipy`. An equivalent function `bwlabel(BW, n)` in the MATLAB image processing toolbox can also be used [111].



Path:Oki2011/mean.png

Num of LEDs: 61

Gamma 0.95

Thresh 14

Additional tag

Gamma correction slide bar

Threshold slide bar

Misdetected LED can be deleted manually

Detected LED positions marked as yellow circle

Figure 6.5: Screenshot of GUI-based mask correction tool

For each assigned label, the center of gravity (c_{xi}, c_{yi}) is calculated ($i \in [1, ..., M]$, where M denotes the number of LEDs detected).

For manual LED position correction, we developed the GUI-based tool shown in Figure 6.5. The user can select the Gamma correction parameter and threshold by the slide bars for legibility. All the centers of gravity of masks are displayed in the window as yellow circles. The users can correct them by adding a missing mask by left clicking on the image, moving the mask by dragging, or deleting a misestimated mask by right clicking.

(4) Mask Generation

Using the detected and corrected LED positions and the following bottom-up algorithm, the system generates a mask image containing all the LED positions. We define *mask image* as a binary image in which the values of the pixels covering the LEDs are 1, and the values of the others are 0. Let M be the number of masks detected, *result* be the set of coordinates in the mask, and *pos* be the initial value of mask generation, i.e., the seed for mask generation. Note that threshold θ is determined in the third step. The bottom-up mask generation procedure is:

```
procedure maskGeneration($result$, $I_{mean}$, $pos$, $\theta$)
```

```
        if $I_{mean} > \theta$ then
            append $pos$ into $result$
            maskGeneration($result, I_{mean}, pos + (1, 0), \theta$)
            maskGeneration($result, I_{mean}, pos + (-1, 0), \theta$)
            maskGeneration($result, I_{mean}, pos + (0, 1), \theta$)
            maskGeneration($result, I_{mean}, pos + (0, -1), \theta$)
        end if
    end procedure
```

Calling the procedure for each center of gravity, the system generates mask image $I_{mask}(x, y)$

```
    initialize $I_{mask}$ all zero
    for $i = 1$ to $M$ do
        maskGeneration($result, I_{mean}, (c_{xi}, c_{yi}), \theta$)
        $I_{mask}(x, y) = \begin{cases} 1 & (x, y) \in result \\ I_{mask}(x, y) & (x, y) \notin result \end{cases}$
    end for
```

(5) Time Series Extraction

The time series of each device's light intensity $l(m, t)$ is extracted by using the masks:

$$l(m, t) = \sum_{x=0}^{W} \sum_{y=0}^{H} \Big(I_{mask}(x, y) I_g(x, y, t) \Big) \tag{6.3}$$

where W and H denote the frame width and height, respectively, and m and t denote the indexes of the LED and the frame, respectively.

(6) Call Timings and Places Detection

Next, the timing and position of each call are localized. In preprocessing, we set any component less than threshold α to zero to eliminate noise; in our analysis, we empirically set α to 0.1.

Table 6.2: Sound pressure levels of advertisement calls of *H. japonica*

Sound pressure level [dB SPL]	Snout-vent length [mm]	Weight [g]
85-88	31.3	1.9
83-86	36.7	3.0
85-90	32.5	2.5
90-95	32.8	2.4
88-93	33.9	2.6

Peaks are detected by sliding a square window of size L along the time series $l(m, t)$. If the center of the windowed time series is the maximum value, we say the position corresponds to a call. In this algorithm, for every (m, t), a call is made at location m and time t if (m, t) satisfy $l(m, t) \geq l(m', t')$ for all $m' \in [m - L, \ldots, m + L]$ and $t' \in [t - L, \ldots, t + L]$. We empirically set L to 5.

6.3 Experiments

In this section, we evaluate the Firefly and entire sound imaging system. We measure the light intensity in response to various sounds in 6.3.1, reveal the limitation of the system regarding to the inter-frog distances and inter-Firefly distances by simulation in 6.3.2. We then conduct two experiments using *H. japonica* in indoors (Section 6.3.3) and in their habitat (Section 6.3.4).

6.3.1 Device Analysis

Sound-Pressure-Level to Light-Intensity Characteristics

We measure the light intensity of a Firefly in response to various sounds. We use six kinds of test sounds (*H. japonica* call, 1k, 2k, 4k, 8k, and 16k Hz pure tones) from 30cm far from a Firefly's microphone. The distance is almost the same width of the dike of our target paddy field. The *H. japonica* call is used to evaluate the response to our target animal call, and 1k-16k Hz pure tones are used to evaluate the wide range of sounds. The results of pure tones can be used for other species having different frequency spectrum. Table 6.2 shows the sound pressure levels of

H. japonica's advertisement calls. This is measured at 30cm distance from the frogs on 24 Mar. 2012 in Iwakura, Kyoto, Japan. The sound pressure levels are from 80 to 95 [dB SPL], and 86.8 [dB SPL] on average, and the main frequency components are 1.7 and 3.0 kHz. We use five angles of the variable resistor (0, 90, 135, 180, and 270°) to cover the entire range of the rotation, ten sound pressure levels (From 70 [dB SPL] to 100 [dB SPL], measured at 30cm distance from the loudspeaker.), with or without a silicon cap. For each condition, we use five Fireflies to compensate the individual difference of Firefly. For each sound pressure level, the sound is given to a Firefly for two seconds, and the light intensity for each condition is taken as the average. The light intensity is captured by the video camera and analyzed using the procedure described in the data analysis section.

Figure 6.6 shows the result. Vertical axes denote the light intensities in pixel value. Horizontal axes denote the sound pressure levels of the test sound in dB SPL. The figures (a)-(f) are the result with a plastic cap, and the figures (g)-(l) are without the cap. From the left, each figure corresponds to each test sound, *H. japonica* call, 1k, 2k, 4k, 8k, and 16k Hz pure tone. For each figure, all lines correspond to an angle of the variable resistor. The error bars denote the standard deviation of all Fireflies.

For any conditions, the best angle of the variable resistor exists between 90 to 135°, which maximizes the dynamic range of the light intensity. Comparing the figures (a)-(f) and (g)-(l), we notice two advantages of the silicon cap; it increases the dynamic range of the light intensity, and it decreases the error bar, i.e., the individual difference of Fireflies. Therefore, we confirmed that adding a silicon cap contributes to make LED detection easier.

Directional Characteristics

We evaluate the light intensity from different directions, i.e., the directional characteristics. We play the *H. japonica* call at ten different sound pressure levels (from 70 [dB SPL] to 100 [dB SPL], measured at the microphone of the Firefly) from

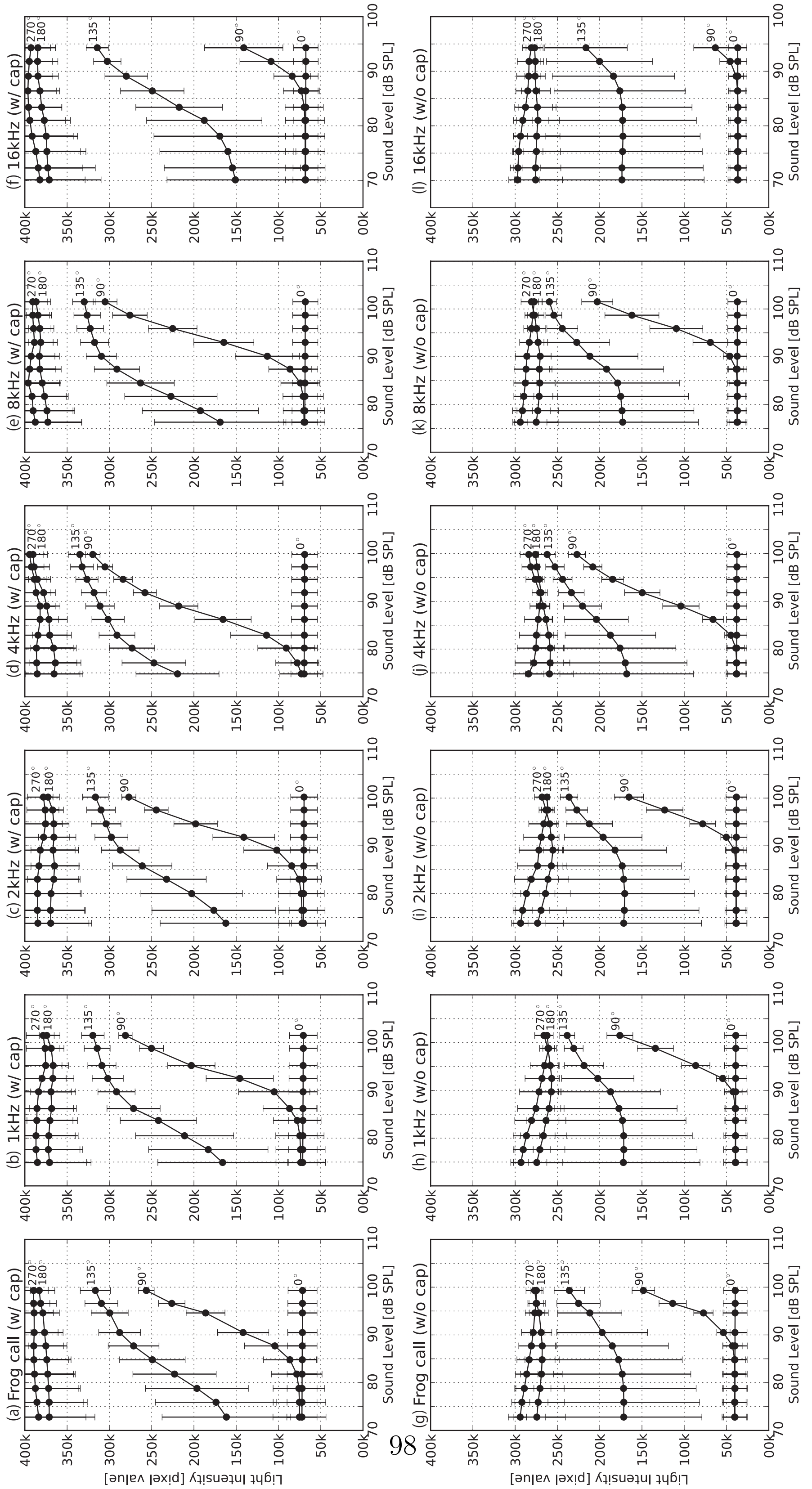


Figure 6.6: Sound-pressure-level to light-intensity characteristics

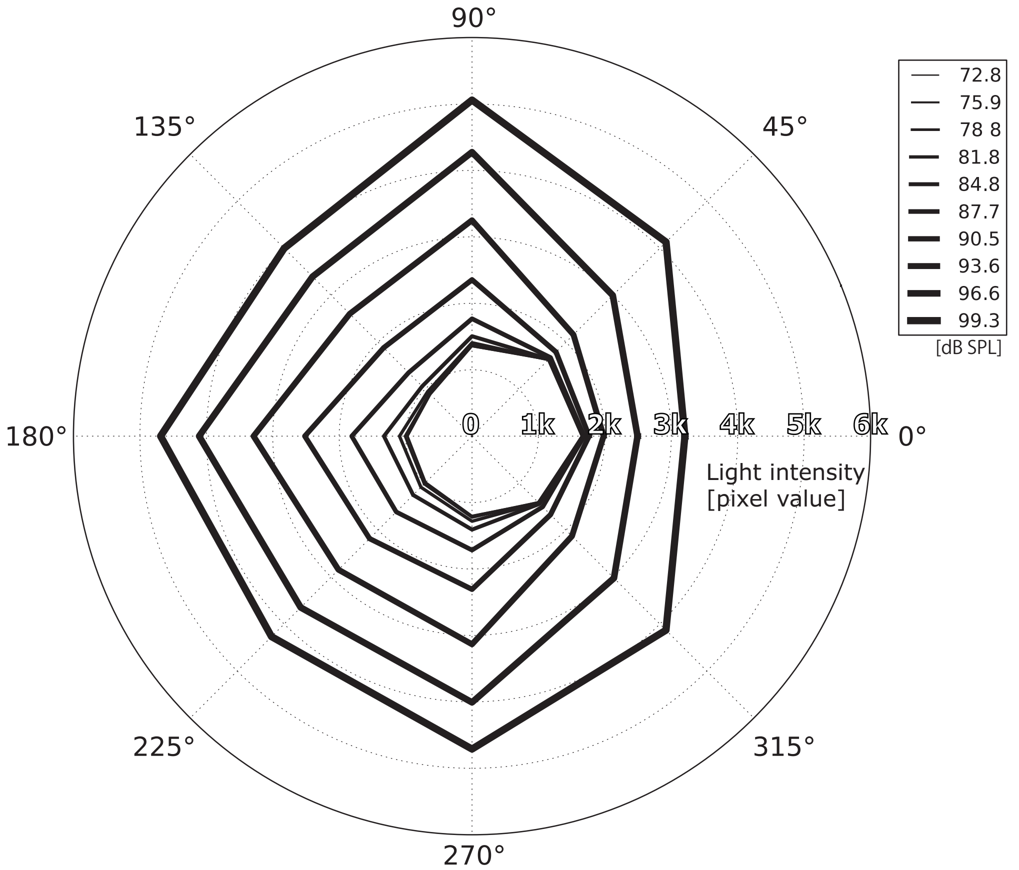


Figure 6.7: Directional characteristics of Firefly

eight directions (0°, 450°, ..., 315°, the right side of Figure 6.2 is 0°, and the top side of the figure is 90°). The distance between the loudspeaker and the microphone of the Firefly is 30cm. For every direction, we use five different Fireflies to compensate the individual differences. The variable resistors are set to 135°. The light intensity of the Firefly is captured and analyzed in the same process as the sound-pressure-level to light-intensity characteristics evaluation.

Figure 6.7 shows the directional sensitivity. The directions 0° and 90° correspond to the right and top of Figure 6.2. The lengths correspond to the total pixel values of the mask of LED. Each line corresponds to different sound pressure level of the test sound. The thickest line corresponds to the loudest, and the thinnest corresponds to the quietest. The standard deviation of all conditions is 384 [pixels]. According to the figure, the light intensity is the largest if the sound comes from 180° (corresponding to the left side of Figure 6.2).

Practical Aspects

Some important practical aspects of the sound imaging system are evaluated.

Battery life The analysis of 20 Fireflies indicates that the Fireflies work for 221.4 min on average with two full-charged standard 750[mAh] batteries.

Waterproof The left column of Table 6.3 shows the humidity of our past experiments. The data is collected by the experiments held from the field experiments held from 17 to 21 June 2010 and from 9 to 17 June 2011 on Oki Island, Shimane-prefecture, Japan, and from 14 May to 17 June 2010 on the experiment farm of Kyoto University, including three days of the light rain. The humidity was from 96 to 53.5%.

As the table suggests, our system is robust against high humidity and light rain. Note that, no Fireflies malfunctioned among 100 copies during four-year experiments except the stripped thread of the variable resistor. The reason is twofold: a Firefly is made of discrete parts, which are more robust against the water, and is almost entirely enclosed in a plastic box except for a hole for the variable resistor. Therefore, rain rarely reaches the Firefly in case of light rain.

Temperature resistance The right column of Table 6.3 shows the temperature of our past experiments. The temperature was from 12 to 24°C.

The data suggests that the system works for typical temperatures in habitats of Japanese tree frogs. However, the range of investigated temperatures is not wide enough to judge whether our system is robust against the temperature. A detailed investigation under a wider range of temperatures is needed for

Table 6.3: Humidity and temperature histograms

Humidity [%]	Frequency	Temperature [°C]	Frequency
40 - 50	1	10 - 15	2
50 - 60	2	15 - 20	11
60 - 70	7	20 - 25	15
60 - 80	5		
80 - 90	7		
90 - 100	6		

other species. Note that this is not the problem for Japanese tree frogs since they mainly call during the rainy season of early summer.

6.3.2 Simulation Experiments

We evaluate the number of localization error and the absolute localization error with various inter-frog distances in simulation. In the simulation, we assume that all the animals call at the same time. This is unrealistically difficult condition because *H. japonica* calls alternately. This in-phase synchronization situation is the most difficult because the temporal sparseness does not exist. Therefore, we can use only the light intensity difference. We used this condition to reveal the limitation of our system.

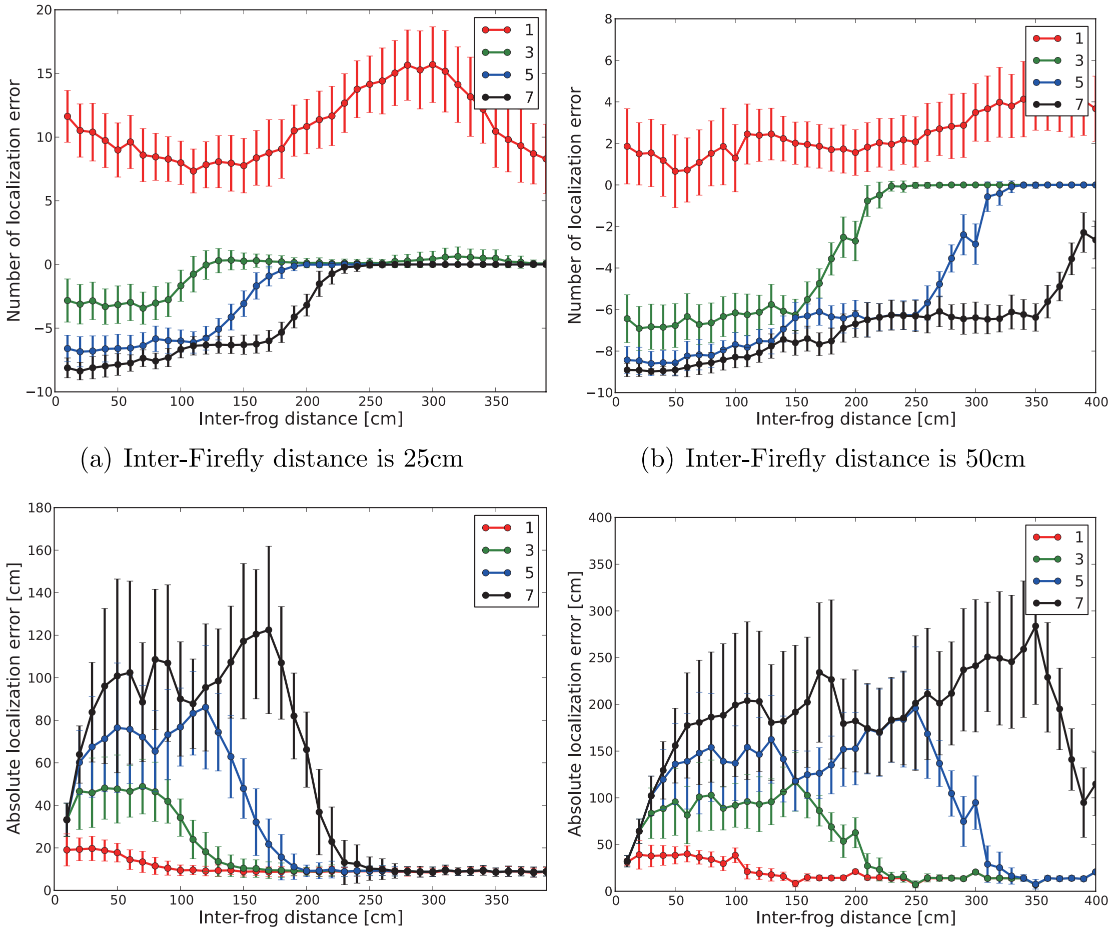
The simulation setup is as follows: (1) the number of frogs is set to 10 and the inter-frog distance changes from 10 to 400cm. The frog positions are equally placed at the given inter-frog distance first, then, perturbed at most ± 25cm randomly. (2) The number of Fireflies are set to 100, and the inter-Firefly distance is set to 25cm or 50cm. (3) The sound pressure level of the frogs are randomly selected from 79 to 81 [dB SPL]. (4) The sound-pressure-level to light-intensity characteristics of the Fireflies are set using the result (Figure 6.6) of the condition of *H. japonica* call, with a cap, and the 135° variable resistor angle. We then add ± 10% error for all Fireflies to represent the individual differences of them. Then, we calculated the received sound pressure level for each Firefly based on the fact that the sound pressure level decays in proportion to the inverse square of the distance. For each placement of frogs and Fireflies, the peaks of light intensities are detected by sliding a window (see the 6th step). We changed the window length L from 1 to 7.

We use two evaluation criteria: the number of localization error and the absolute localization error. The purpose of the former is to evaluate whether the number of frogs are accurately estimated. We defined it as $|M - N|$ where M and N denote the correct number of frogs (10 in this case) and the number of estimated peaks. The purpose of the latter is to evaluate how precisely the frogs locations are estimated. The definition is as follows: Let N be the number of detected frogs, M be the correct number of frogs, $(\hat{x}_i, \hat{y}_i)$ be the coordinates of the detected frog positions, and (x_j, y_j) be the coordinates of the correct frog positions. Then, the criterion is:

$$\frac{1}{N}\sum_{i=1}^{N} \min_{j=1\ldots M} \left((x_j - \hat{x}_i)^2 + (y_j - \hat{y}_i)^2\right)^{1/2} \tag{6.4}$$

Figure 6.8 shows the result. Each line corresponds to the window length for peak picking ($L = 1, 3, 5, 7$). The standard deviations are used for error bars. (a) and (b) show the number of localization error. The positive value means too many sound localization, and a negative value means too less. (c) and (d) show the absolute localization error in centimeters. The less error means better performance. Note that we assumed that all sounds are produced simultaneously.

In Figure 6.8(a) and (b), L must be larger than 1 to detect the number of frogs accurately. For $L > 1$, the number of frogs error reaches zero when the inter-frog distance is wider than 125cm ($L = 3$), 175cm ($L = 5$), and 225cm ($L = 7$) for 25cm inter-Firefly distance, and 225cm ($L = 3$) and 325cm ($L = 5$) for 50cm inter-Firefly distance, respectively. In Figure 6.8(c) and (d), the absolute localization error reaches the minimum when the inter-frog distance is wider than 150cm, 200cm, and 250cm for 25cm inter-Firefly distance, and 225cm and 325cm for 50cm inter-Firefly distance, respectively. In summary, the best L is 3 because it is the shortest window length so that the best performance of the criteria. Then,



(a) Inter-Firefly distance is 25cm

(b) Inter-Firefly distance is 50cm

(c) Inter-Firefly distance is 25cm

(d) Inter-Firefly distance is 50cm

Figure 6.8: Simulation results

the inter-frog distance should be longer than 6 and 6.5 times for 25cm and 50cm inter-Firefly distances, respectively.

Note that we assume the most difficult situation, that is all frogs call simultaneously. However, they call in anti-phase synchronization rather than in-phase synchronization. Therefore, “inter-frog distance” in this simulation is wider than the actual one.

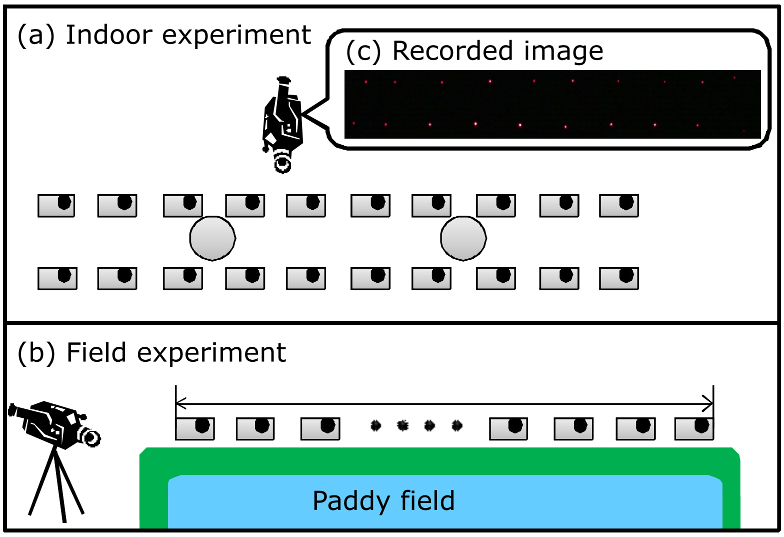


Figure 6.9: Experimental conditions

6.3.3 Frog Choruses Visualization in Indoor

Experimental Setup

The indoor experiment was conducted on 8 June 2009. Two Japanese tree frogs were caught in the field of the experimental farm of Kyoto University, Japan, and placed in separate cages. The snout-vent lengths were 38.83mm and 32.85mm, and their respective weights were 2.4g and 2.5g.

The experiment setup is shown in Fig. 6.9(a). In the indoor experiment, we used 10 Fireflies (rectangles, 20 Fireflies in total) in each row at a 21.5cm interval, and installed an off-the-shelf video camera above the experimental area at a height of 5.13m. Fig. 6.9(c) shows an example of an image recorded. Then, we placed one frog between the 3rd and 4th Fireflies and the other one between the 7th and 8th Fireflies (shown in gray circles in the figure.) The humidity was 54% and the temperature was 25.0°C. We recorded the Firefly emissions throughout one night.

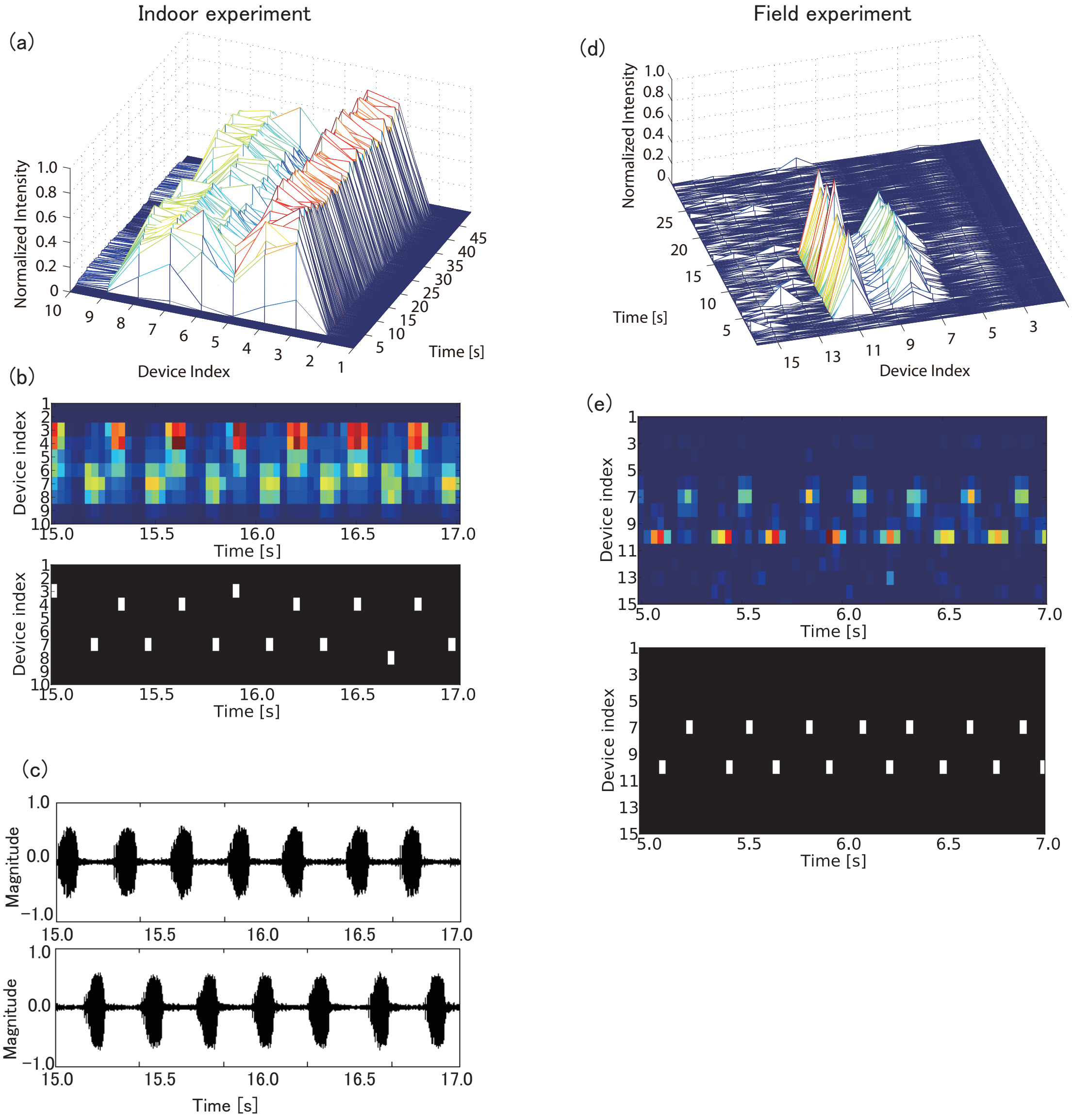


Figure 6.10: Spatio-temporal visualization of the calls of two frogs (2009)

Results

Fig. 6.10(a)-(c) show the indoor experimental results. Fig. 6.10(a) is the visualization of calling behaviors for the indoor experiment. The x-, y-, and z-axes denote time, the device index, and the normalized intensity, i.e., the sound intensity of the call, respectively. The upper panel in Fig. 6.10(b) is a magnification of Fig. 6.10(a) from 15.0 to 17.0 s. The lower panel shows the localization; each white pixel represents a call. The anti-phase synchronized calling behavior is evident. Figure 6.10(c) shows the waveforms corresponding to the calls; they were separated using independent component analysis.

The two peaks in the sound imaging results (Fig. 6.10(a) shows an excerpt of the result) around the 3rd and 7th Fireflies correspond to the frogs' locations. This means that our method correctly visualize the sound source locations. The magnification in Fig. 6.10(b) for 15.0 to 17.0s clearly shows the sequences for the two peaks. The alternation of the peaks means that the two frogs called alternately. This anti-phase synchronization behavior is the same as that reported by [14,114].

Figure 6.10(c) shows the waveforms recorded for the calls of the two frogs. They were recorded using two microphones placed near the frogs to enable us to evaluate the temporal accuracy of our system. Because their calls were loud, the recorded waveforms were a mixture of the two frogs' calls. We separated them using an independent component analysis [16,88]. Note that the ICA is applicable to this case, because (1) the number of sound sources is the same as that of microphones and (2) the amplitude histogram of Japanese tree frog calls is non-Gaussian [114]. The waveform in the upper panel in Fig. 6.10c corresponds to the sequence of peaks for the 3rd Firefly, and the lower one corresponds to that for the 7th Firefly. Calling detections illustrated as white pixels (the lower panel of Fig. 6.10b) coincide with the peaks of the waveform (Fig. 6.10c). This means that our method correctly visualize the temporal aspect of the calling behaviors. These results demonstrate that our method is capable of visualizing the spatio-temporal calling behavior, i.e., both the call times and locations, of Japanese tree frogs.

The lower panels in Figs. 2b and 2e show the times of the calls and the locations of the two frogs estimated from the corresponding upper panels. The average locations of the frogs were 131.2cm and 61.3cm for the indoor experiment and 221.1cm and 315.8cm for the field experiment. The standard deviations were 7.1cm and 7.6cm for the indoor experiment and both zero for the field experiment. These values were calculated by multiplying the device index by the Firefly intervals (21.5cm for the indoor experiment and 31.6cm for the field experiment).

We calculated the estimation error for the indoor experiment because the locations of the frogs were known. They were near the 3rd and 7th Fireflies, which correspond to 64.5cm and 150.5cm, respectively. Therefore, the mean estimation errors were 3.2cm and 19.3cm, respectively. These errors are less than the device interval.

6.3.4 Frog Choruses Visualization in Field

Experimental setup

We conducted two field experiments in 2009 and 2011. For each year, the experiments are conducted in the same island, but different paddy fields. Note that in this case, the frog locations are unknown.

For experiments in 2009, we conducted alongside a rice paddy field in Oki Island, Shimane Prefecture, Japan, on 3 July 2009. The humidity was 76%, and the temperature was 24°C. The experimental setup is shown in Fig. 6.9(b). We placed 20 Fireflies in a row along the edge of the paddy along the edge of a rice paddy 6m long, and the off-the-shelf video camera on the side of the paddy.

For experiments in 2011, we conducted the experiments at a paddy field in Oki Island from 9 to 16 June 2011 (Figure 6.11). The field was covered with moist soil and grasses of 10cm high. We deployed the Fireflies with 40cm inter-Firefly interval along the edge of the paddy field indicated by the arrow in Figure 6.11. We deployed them for 33.8m length from the red circle point in the figure. We then

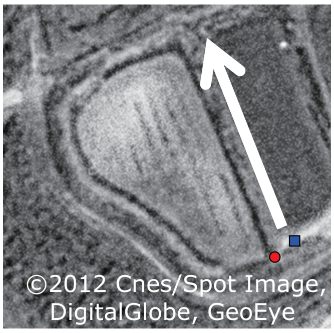


Figure 6.11: The paddy field of our field experience.

captured them by the video camera from the blue square point. This placement is determined because the frogs typically call on the edge of the paddy field with a few meters inter-frog distances. This sparse distribution is validated from our experience and two investigations; (1) the prolonged breeders have sparse spatial distribution [115], and (2) *H. japonica* has long breeding season (typically three months) [116]. Therefore, the distribution of *H. japonica* is sparse. The Firefly at the red circle corresponds to the origin, i.e. 0[m], of the visualization results. The video camera is placed around the blue square in Figure 6.11, where is about 2m higher than the paddy field.

Results

For field experiment in 2009, we recorded 20 frog-calling sequences. The average number of calls per sequence was 238. Figure 6.12 shows a duration histogram of the sequences. Each duration is defined as the difference between the starting time of the first call in a sequence and ending time of the last call in the same sequence. The typical duration was between 20 and 40s, and the longest was 108s.

Figures 6.10(d) and 6.10(e) show the results for the field experiment the axes are the same as Fig. 6.10(a) and (b). As visualized in Fig. 6.10d, two frogs near the 7th and 11th Fireflies started calling at 2.5s and ended at 15.0s. The sequence of the peaks for their call from 5.0s to 7.0s (Fig. 2e) shows that the two frogs

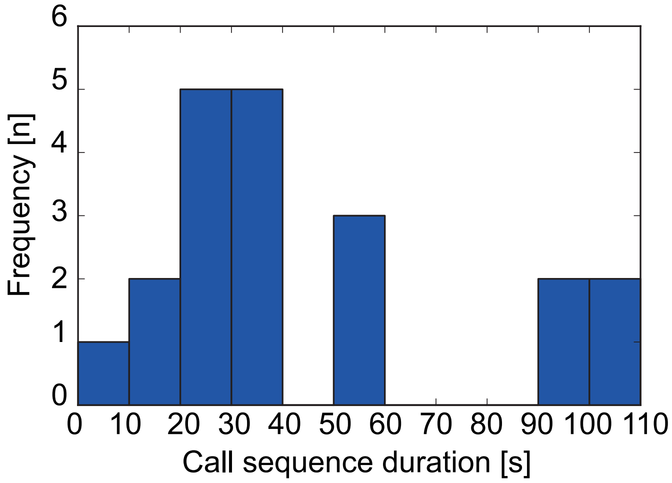


Figure 6.12: Histogram of the call sequence durations:

called alternately, i.e., in anti-phase synchronization. This result confirms that two Japanese tree frogs call in anti-phase synchronization even in their natural habitat. We thus succeeded in observing the calling behavior of Japanese tree frogs in their natural habitat with high temporal (1/30s) and spatial resolutions (21.5cm indoors and 31.6cm in the field). Although the locations of the frogs were unknown in the field experiment, we evaluate the distance between them. The distance was calculated by multiplying the device index by the intervals to be 102.6cm. This is consistent both with the finding of a previous study that Japanese tree frogs are sparsely distributed in a field [115, 116], and with our empirical observation that they call at a distance of 1m to 3m.

Figure 6.13 shows a visualization of simultaneous calls in the field experiment. The axis labels are the same as in Figure 6.10(b). From 1.0s to 3.0s, the system separately localized the overlapping calls of two frogs. This indicates the visualization capability of overlapping calls of the system in animals' habitat.

For field experiment in 2011, we present two sets of results to demonstrate the system's call visualization capability. Figure 6.14 shows the results of multiple frog visualization. The upper panel shows time series of normalized light intensi-

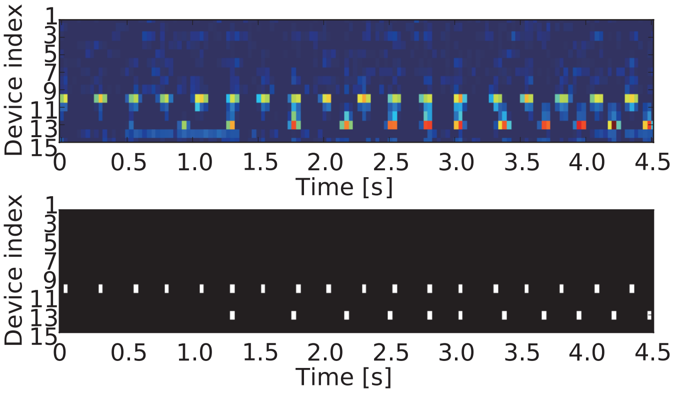


Figure 6.13: Visualization example for temporal overlapping

ties, the middle shows the time and locations of calls, and the bottom shows the corresponding sound recorded by the video camera. Vertical axis of the upper and the middle denote the distance from Firefly closest to the video camera, and that of the bottom denotes the magnitude of the waveform. The horizontal axes of all panels denote time.

In Figure 6.14, we can find six sequential calls at 0, 1, 3, 4, 6, and 8[m] meaning that this chorus consists of six frogs. For later discussion, we name these frogs as a, b, c, d, e, and f. Based on the inter-frog distance, they form three pairs at (a, b), (c, d), and (e, f), respectively. These three pairs call alternately, i.e., in anti-phase synchronization. This synchronization state is well known behavior [111, 114]. Therefore, we confirmed that even in large frog choruses, the frogs can synchronize in anti-phase with neighboring frogs. Note that the pair of (e, f), the state of anti-phase synchronization is less stable than the others. For example, they are anti-phase synchronized from 5 to 6 [sec], are in-phase synchronization from 10 to 11.5[sec].

For quantitative discussion of this chorus, we analyzed the call sequences of the neighboring frogs using the stroboscopic technique [1]. Figure 6.15 shows the sketch of this technique. Assume that we have two call timings sequences of two frogs, named Sequence A and Sequence B. We define the phase of the sequence A

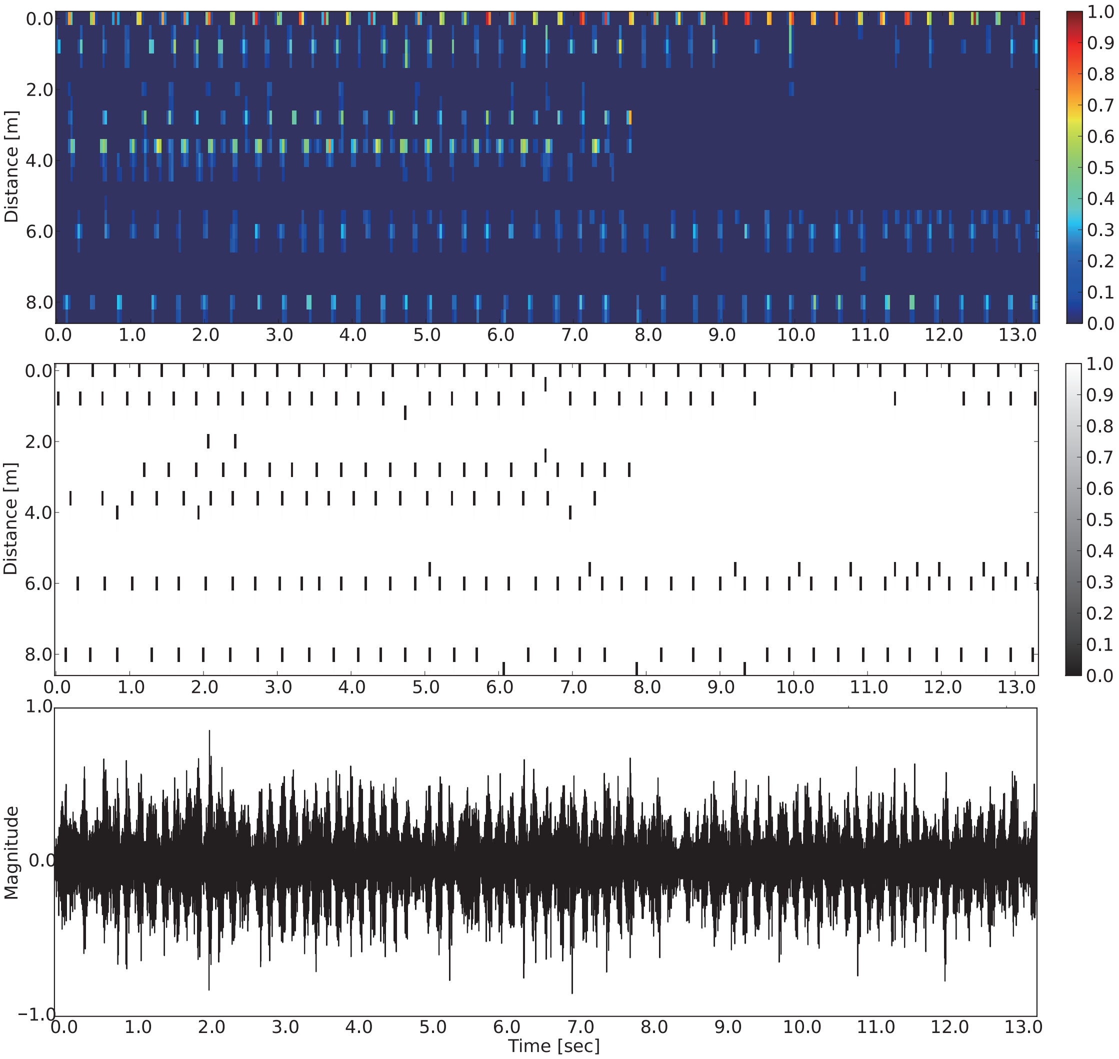


Figure 6.14: Visualization of a frog chorus consisting of six frogs (12 June 2011)

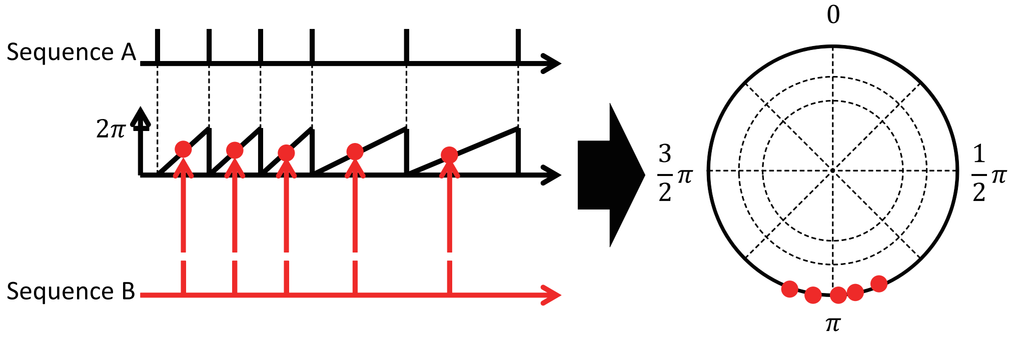


Figure 6.15: Overview of the stroboscope method

shown in the middle graph of Figure 6.15 as follows: the phase is zero at the call timing, and it linearly increases as time advances so that it becomes 2π just before the next onset timing. Then, we extract the phase values for each call timing of the sequence B. Therefore, the number of the phase values is the same as that of the calls of sequence B (The red circles in Figure 6.15).

Using the phase values, we can examine whether these sequences are synchronized or not (Right side in Figure 6.15. If the call timings of two sequences are in-phase synchronized, the mean phase becomes zero. If the call timings of two sequences are anti-phase synchronized, the mean phase becomes π. Note that the mean phase is calculated by the mean direction [117], which is used for directional statistics. We can statistically test the phase values using the Rayleigh test that tests whether the phase is uniformly distributed or not [117]. If the phase distribution is significantly biased, we can interpret that the sequences A and B are synchronized.

We calculated the phase values for all neighboring frog calls and tested whether they are synchronized or not. Figure 6.16 shows the histogram of the phases and Table 6.4 summarizes the phases. From the left of the table, the columns mean that the name of the calling frogs, their distance, mean direction, and the p value for the Rayleigh test. The p value columns has $*$ if the phase is significantly biased ($p < 0.01$). From the table, we can find that two frog pairs (a, b) and (c,

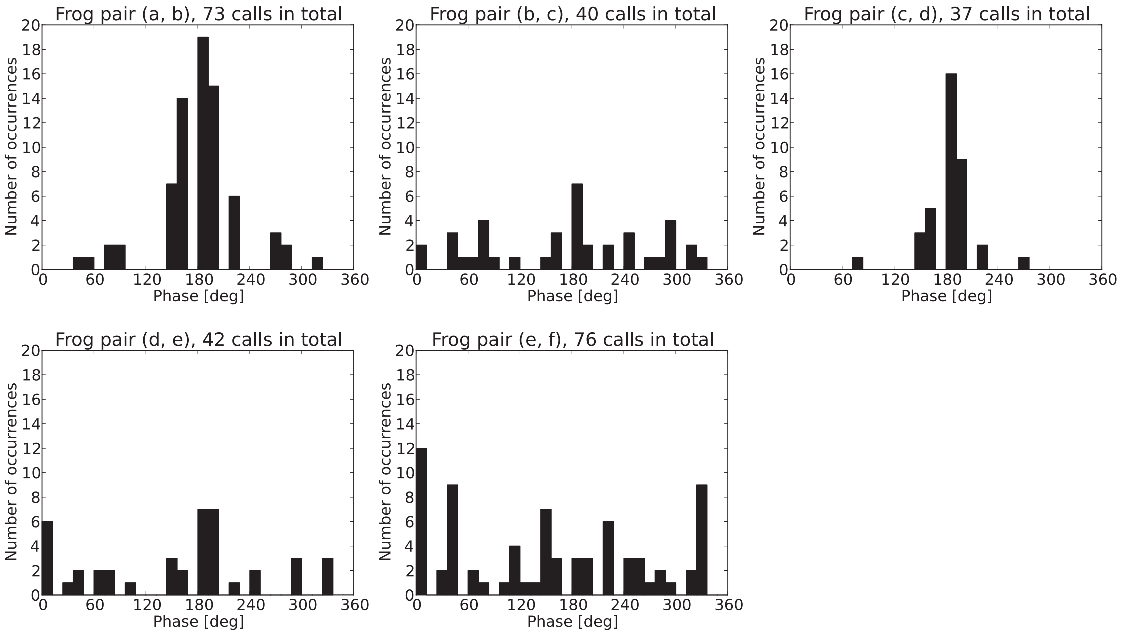


Figure 6.16: The histogram of the phases for each frog pair. According to Table 6.4, the pairs (a, b) and (c, d) called in anti-phase synchronization

Table 6.4: Statistical analysis of synchronization ($p < 0.01$)

Frog A	Frog B	Distance[m]	Mean direction[deg]	p-value
a	b	0.8	179.4	¡ 0.01*
b	c	2.0	179.5	0.13
c	d	0.8	181.2	¡ 0.01*
d	e	2.4	185.1	0.32
e	f	2.0	4.5	0.34

d) called anti-phase synchronization whereas the pair (e, f) was not significantly synchronized. Therefore, this can be interpreted that the pair (e, f) located too far, 2.0m, to synchronize.

Figure 6.17 shows another example of 1:2 anti-synchronization of three frogs discussed in [114]. The horizontal and vertical axes are the same as Figure 6.14. The calls of two frogs at 4.5 m and 7.2 m were synchronized in-phase, and those of the frog at 6.2 m and of two others were synchronized anti-phase. Thus, the chorus of the three frogs is 1:2 anti-synchronization. This result suggests that the

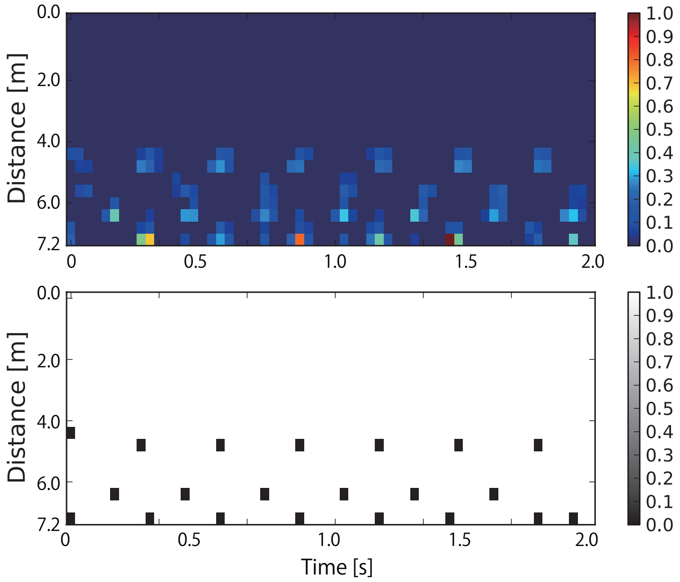


Figure 6.17: Visualization of 1:2 anti-synchronization calling in the field

model proposed by Aihara [114] is also valid for natural habitats.

6.4 Discussion

6.4.1 Requirements of General-purpose Investigation Method

We have shown that the sound imaging system satisfies all the requirements for a general investigation method of acoustic communication. (1) It can be used in the natural habitat. (2) It can detect multiple animal calls (Figures 6.14 and 6.17). (3) It has little influence on the animals because the observers can be away from the site after setting up the system. The influence is further reduced by selecting an LED color that is difficult for the animals to detect. (4) It can be applied to different species that have a two-dimensional distribution. Even if the target animal has a different distribution density, it can be applied by changing the design parameters, such as the number of Fireflies, the inter-Firefly distance, the color of the LED, and/or the microphone's frequency response, waterproof, and directivity.

6.4.2 Characteristics of Firefly

In Figure 6.6, we can find the angle of the variable resistor that maximizes the dynamic range of the light intensity for any sounds from 1 kHz to 16 kHz not only for *H. japonica* calls. This suggests that, if we apply the system to other species, we can find the best tuning to visualize their calls from Figure 6.6 as far as their frequency components are from 1 kHz to 16 kHz.

The result of Figure 6.6 from other distance can be calculated based on the fact that the sound energy decays in proportional to the square of the distance. Note that Figure 6.6 is measured using the sounds at 30cm distance. For example, if the Figure 6.6 is shifted 3 [dB SPL] to the left, the figure corresponds to the result of the sound from 60cm.

We adopted a continuous resistor for the gain adjustment component rather than a discrete one because we need to control the gain precisely to absorb the environmental effects, e.g., the sound pressure levels of the animal calls, the wind noise, and the rustling sound of trees. A discrete resistor such as a switch-selected attenuator is easy to adjust and record the adjusted value. However, the size of the attenuator becomes larger if we make the step size smaller. To take advantage of the discrete gain control, we are developing a new version that uses digital gain setting via a wireless communication.

The reason of the non-omnidirectional sensitivity of Firefly is its asymmetric shape. For example, in the condition that the loudspeaker is at 180°, the distance from the *edge* of Firefly is the shortest than other directions even if the distance from the *microphone* is kept 30cm. This changes the sound propagation speed because the plastic box brings the sounds with less decay than in the air. Note that, for practical use, this directional sensitivity gives less effect to the result by deploying Fireflies with the same direction.

6.4.3 Synchronizations in a Frog Chorus

Figure 6.14 suggests that even in large choruses, frogs interact with only the nearest neighbor. From the viewpoint of mating, it is natural that two frogs call in anti-phase synchronization because it maximizes the call timing difference from the other so that the females can localize the male frogs' calls. In contrast, non-nearest frog pairs have no such synchronization. This phenomenon can be interpreted by the selective attention [118, 119], which is an aspect of the frog behavior that they are affected by only neighboring calls having loud calls.

In the Figure 6.14, the first two pairs (a, b) and (c, d) call in anti-phase synchronization significantly. However, the synchronization of the last pair (e, f) was insignificant. This is because the distance of the last pair is longer than the others. The selective attention of frogs can explain this difference; when the pair is at close place, their interaction is strong because they pay more attention each other. In contrast, when the pair is at the far place, their interaction is weak because they pay less attention each other.

6.4.4 Advantages of Sound Imaging System

The main advantage of the sound imaging system compared to multichannel microphone recording is that no inter-channel synchronization is required. This is because the speed of light is sufficiently faster than that of sound to make the time difference of arrival negligible. Therefore, users can easily increase the number of Fireflies without having the equipment to synchronize the recording, as required for microphones.

The frogs' call behavior has been widely studied. [5] discusses many studies of various kinds of calls, such as territorial calls, aggressive calls, and advertisement calls, and their spectrum pattern and roles for territory keeping. However, the main focus on these works is individual characteristics and one-to-one interaction of frogs. In contrast, our goal is to investigate the whole interaction in frog choruses of many frogs.

Recently, microphone arrays are used for bioacoustic investigation [80, 81, 87]. The advantage of microphone-array processing is that rich information is available e.g., spectrogram. The disadvantage is that precise synchronization must be used to apply signal processing techniques. The advantage of our method compared to these methods is that the video camera and Fireflies are physically separated. This makes the flexible Firefly placement depending on the target animal distribution, not considering to, e.g., the length of a microphone cable. This also enables researchers to use our method intuitively; just deploy Fireflies of the target field, and capture all lights with a video camera.

Some companies are producing sound visualization systems named *acoustic camera*, e.g., High Definition Acoustic Camera of LMS International, visualizes the existence of sound precisely (for example, see [120]). The advantage of our method is that it can visualize the sound existence of wide depth (16m in our experiment). Because microphone array processing requires precise synchronization of microphones, microphones are not distributed in a wide range. Therefore, the distance between the microphones and the sound sources is long for far sounds. This degrades the localization performance because of the decaying sound energy and other interfering sounds such as other frogs and background noise. In contrast, our method can be distributed along the large field because our method needs no synchronization of microphones. This is because we use light, which propagates so fast. Therefore, we can assume that the times of a LED flashing and a video receiving are the same.

6.4.5 Limitations of Sound Imaging System

First limitation of the system is about the field. We discuss the feature of the field that is difficult to use the system. The field must have a place for a video camera where it can capture all the LEDs. For the vegetation, the grasses on the ground must be short enough to ensure that all LEDs can be seen. If the grasses are tall, attaching a stick on the back of Fireflies to heighten the LED positions may solve the problem. For the terrain, the ground must be plain enough to capture all the LEDs. Even if it is too rough to capture all the LEDs, it will be still possible to use the system by capturing through multiple video cameras and common reference points. Obviously, the video cameras must be synchronized in advance. For the weather, too strong wind or rain will disturb the system because all the LEDs will light by the blowing air or hitting raindrops regardless of the sounds existence. In addition, the strong rain will break also a video camera. According to our experience, the weak rain and the wind has not significant effect to the visualization result because the temporal lighting pattern is different from animal calls.

Another limitation is about animal distribution. First, according to the simulation result, the inter-animal distance must be wider than 6-6.5 times than inter-Firefly distance. Therefore, the inter-animal distance should be roughly known in advance, and the Fireflies must be deployed densely than it. Note that *H. japonica* tend to call alternately. Therefore, the inter-frog distance can be wider in a practical situation. Second, dimension of the distribution must be linear or planar, i.e., one or two dimensional. If the animal distributes linear as the same as our experiment, the system can visualize the spatio-temporal structure of their chorus. For the animals having planar distribution, the system still can visualize although more Fireflies are required to cover the habitat than the linear distribution case. However, for the animals having a tridimensional distribution such as bats and birds, the system is incapable of visualizing because Fireflies must be placed on the ground.

The Fireflies respond to all nearby sounds between 50 Hz to 15 kHz according to the data sheet of the microphone without any distinction. Therefore, the Fireflies are incapable of distinguishing the coexisting frog calls, e.g., the calls of *R. rugosa* and *H. japonica* in the paddy field of Kyoto University, Japan. To solve this problem, we are developing a new Firefly that has band-pass filters to realize the specie-specific sensitivity based on the frequency component.

The system can retrieve only the positions and timings, not sounds themselves. This limitation can be overcome by combining the synchronized microphone array recording. Because some sound separation methods such as beamforming [121] requires the sound locations, the system can give the sound locations for the methods.

Current problem is that the sound-pressure-level to light-intensity characteristics is adjusted manually. Therefore, the human errors can occur and giving exactly the same adjustment is difficult. To solve this problem, we are developing a new version of Firefly whose gain can be adjusted via a remote controller. Because the remote controller sends the gain by infrared communication, this new version will achieve no-manual adjustment.

Because the system has no function to distinguish individuals, it is impossible to relate calls of the different times at the different locations. As for *H. japonica*, which calls at a short cycle and does not move during calling, we can say that one sequence of calls is produced by the same frog. However, for example, the situation that two animals call at the almost same place simultaneously is beyond the capability of the system.

6.5 Summary

In this chapter, we presented a sound-to-light converting device, Firefly, and a sound imaging system using Fireflies. We analyzed the device and system limitation, and presented indoor evaluation and field experiments. According to the field experiments, we successfully observed the synchronization in frog choruses of Japanese tree frogs. These observations support the mathematical models of the frog choruses.

We have some plans to improve the current system. For example, add another LED to indicate the battery level, add another LED color to distinguish different calls. When we use multiple colors, exploiting LED characteristics will be useful to distinguish colors. We also plan to apply the sound imaging system to other species to evaluate the feasibility.

Chapter 7

Discussion

This chapter discusses the two cases. First, we summarize the result and the contribution for each case study. Next, we discuss the temporal synchronization model of the interaction in general. Finally, we discuss the remaining work.

7.1 Observations

7.1.1 Dynamic Characteristics of Music Instruments

A technical problem with the music robot was the dynamic characteristic tracking of the music instrument. We developed both a parametric model of the theremin and its tracking method by using the UKF. The experiments using a simulation and real robots showed that the robot can keep playing the theremin correctly by using our method.

The contribution of the theremin playing robot is that it can be ported to other robots. Therefore, if a researcher wants to start the study of a co-player robot, he/she can select the robot on the basis of the research purpose. In addition, comparing control methods in a different embodiment is possible because the theremin playing system can work with different robots.

7.1.2 Human-Robot Ensemble Model

For the human robot ensemble, we constructed a state space model of the timings and tempos of the co-players in a multiperson ensemble. We used a coupled oscillator model for representing onset generation and designed leaderness in order to represent the influence on the others. The leaderness is used as the coupling strength of the coupled oscillator model. In the experiment, we implemented the model in the Robot Thereminist system and achieved a lower onset error than the conventional methods. We also validated the design of the leaderness by analyzing the multiperson tapping task.

Our contribution is twofold: the first is the coupled oscillator model. In conventional studies, the robot co-player predicted the human's next onset timing by extrapolation. This means that the robot assumes that the human's playing tempo as constant; in other words, the human ignore the robot's playing. Our coupled oscillator model involves the robot's effect on the human by using the coupling term. Experimental results showed the effectiveness of the onset prediction accuracy.

The second is the leaderness estimation. Conventional ensemble studies assumed that only one leader exists at the same time. Moreover, many studies also assumed that the leader is given in advance. However, leadership will be transferred dynamically and partially among the participants. In other words, a co-player can be "more" leader and "less" leader. We designed the quantitative metric of leadership, *leaderness*, and its estimation method. Analysis of the human-human tapping task validated the leaderness design.

7.1.3 Spatio-temporal Visualization of Frog Choruses

In conventional studies, frog choruses are studied in indoor experiments or qualitatively. This is because audio analysis in the field is difficult, e.g., much noise exists such as that of other species and blowing wind, and a heavy and expensive multichannel recording system is required because of the large target area.

Our contribution is that we developed an inexpensive and easy-to-use visualization system for spatio-temporal structures. By using the system, we visualized the spatio-temporal structure of frog choruses at a temporal resolution of 1/30 seconds and a spatial resolution of 20 - 30 cm. We observed (1) an example of three frogs calling in 1:2 anti-synchronization and, (2) an example of multiple pairs calling alternatively in a chorus. The second observation suggests that they select who to interact depending on the distance, i.e., selective attention.

7.2 General Discussion

We used the coupled oscillator model for both the human-robot ensemble and the frog choruses. We assumed that the individuals generate a sequence of events to others and receive it from the others. For the human-robot ensemble, we constructed a state space model by assigning each co-player to each oscillator. In our design, we assigned the co-players' onsets to the generated events and their tempos to the natural frequencies, and the coupling strength was estimated by using the leaderness. For the frog choruses, we validated a mathematical model of the interaction by using a coupled oscillator model [14]. The model assigns each frog to each oscillator, their calls to the events, and the coupling term as the place. We developed the sound imaging system for the spatio-temporal structure analysis. From the field experiments, we confirmed the synchronization states predicted by the model: anti-phase and 1:2 anti-phase synchronization.

A common observation in both cases is that as the number of participants increases, the natural frequency decreases. For the first case, the simulation result of the multiperson ensemble in section 5.4.2 shows that the convergent IOI increased as the number of co-players increased. This means that the stable tempo became slow when the ensemble size is large. For the second case, Horai *et al.* found that the inter-call interval of solo-calling is shorter than that of anti-phase synchronization [122].

The leader/follower problem is also the common aspect in the both cases. For the first case, we designed the leaders to be the co-players who change the current tempo and keep it constant. Then, we designed the quantified metric of the leadership, *leaderness*, and confirmed that it captures the dynamically changing leaders by using the multiperson tapping task. For the second case, many researchers defines that the leader is the frog that calls the first time [5]. However, from the intuition of the leaderness design, another leader can exist; for example, a leader frog might change the synchronization state, such as from in-phase to anti-phase.

7.3 Remaining Work

7.3.1 Human-Robot Ensemble

Score Following

This work is focused on the timings and tempos of music, in other words, the *local* position information in time. Therefore, if a robot misses a beat to play, it is difficult to recover. In contrast, if the robot can recognize where in the score it is playing, i.e., *global* position information, recovering is possible. This problem is called *score following* [123]. Some on-line score process methods such as [124] will be a promising method to integrate.

Multimodal Ensemble

In this thesis, we focused only on audio information, whereas the ensemble is essentially a multi-modal interaction. The auditory information becomes unreliable in some phases, for example, the beginning, because no sounds exist before an ensemble, and the ending, because an ensemble typically finishes with a *fermata*, a lengthened note. Some studies developed an audio and visual integration for ensemble robots, e.g., beat tracking [63, 125], and gesture recognition [62]. Integrating such techniques will make the ensemble system more reliable.

7.3.2 Frog Choruses

Leadership of Frog Choruses

Many researchers have pointed out the existence of the leader in frog choruses [5]. Here, the leader means the male frog that can attract females the most. A typical definition of the leader is a frog that calls the first time. This is because calling is a dangerous activity; it consumes the energy, and predators can find the caller. However, other definitions of the leader can exist, for example, the frog that has the largest territory or the frog that changes the current tempo. The sound imaging system has the potential to reveal the true “attracting” calling behavior because it can capture not only their timings but also their locations.

The ideas of the leaderness, that is, the leader is an interaction participant who changes the tempo, will be applicable to frog choruses. The frog’s calling behavior have a tradeoff between female attracting and energy consumption, corresponding to positive and negative reasons of calling, respectively. On the other hand, all frogs have a positive reason to call in alternation to be localized by a female. If the “leader” frog changes the calling tempo of the chorus based on its internal state, the chorus becomes profitable to the frog. Therefore, the “leader” frog in our definition will have an advantage of both survival and reproduction.

By using these experimental results, we can estimate the network structure of a frog chorus. In theoretical studies of coupled oscillator models, the estimation of the network structure, including coupling strength estimation, is actively studied [126] [127]. These techniques can be used for the estimation.

Application of Sound Imaging to Other Species

Although we tested the sound imaging system by using only Japanese tree frogs, it can be applied to many species. For example, it is known that many other frogs also from choruses [5], and crickets [128] also calls for mating.

The system should be customized to apply to the situation. For example, to visualize the choruses of ultrasound frogs [69], we need to replace the microphone with an ultrasound microphone. For crickets living in tall grass, the LEDs of the Fireflies should be extended so that the video camera can capture them. If multiple species call in the same field, we need to distinguish them by using their difference. For species having different spectrums, band-pass filters will be useful [129]. For species having different temporal structure, e.g., the call duration of *Rana rugosa* is longer than that of *H. japonica*, the species can be distinguished from visualization.

7.3.3 Temporal Synchronization Model

The theoretical limitation of the coupled oscillator model is that this model targets the succeeding interaction. Therefore, current model is incapable of representing "When the interaction starts/ends?" and "When the synchronization transits to other state?" To answer this question, we need to model the dynamics of the parameters of the coupled oscillator model such as the coupling strengths. The frog's internal energy or the co-player's intention will be the clue of the modeling.

Chapter 8

Conclusion

The final goal of this thesis is to establish a model of synchronization among interacting individuals. We tackled this problem through two case studies: human-robot ensembles and frog choruses. These two cases have different factors, for example, the size, the place, the stationary state, and the participants. For the first case, we developed a co-player robot, the thereminist Robot system, for solo-playing and the oscillator-based two-person ensemble model for co-playing. Then, we constructed a multiperson ensemble model with leadership estimation. We performed human-robot ensemble experiments to demonstrate the capability of the co-player robot and analyzed human-human multiperson tapping task to evaluate the leadership estimation method. For the second case, we developed a sound imaging system for the animal choruses and analyzed the spatio-temporal structure of the *H. japonica* choruses. The sound imaging system consists of a sound-to-light converting device named *Firefly*, an off-the-shelf video camera, and a visualization method for the captured video. We evaluated the characteristics of the device in indoor experiments and revealed the limitation of the system in a simulation. In field experiments, we observed the anti-phase synchronization and 1:2 anti-phase synchronization in the choruses in the frogs' habitat. Therefore, we validated the mathematical model of the choruses.

In both cases, we used the coupled oscillator model. This is widely used to describe synchronization in interaction. For the first case, we adopted coupled oscillators for the onset prediction of co-players and developed coupling strength estimation as the leaderness estimation. For the second case, we validated the mathematical model of the frog choruses by using the coupled oscillators. Since the model was validated only in indoor experiments and a simulation, we observed the same synchronization state in their noisy habitat.

Bibliography

[1] A. Pikovsky, M. Rosenblum, and J. Kurths. *Synchronization: A Universal Concept in Nonlinear Sciences*. Cambridge University Press, 2001.

[2] Y. Kuramoto. *Chemical Oscillations, Waves, and Turbulence*. Dover Publications, 2003.

[3] S. H. Strogatz. *SYNC: the Emerging Science of Spontaneous Order*. Hyperion, 2003.

[4] M. S. Paoletti, C. R. Nugent, and T. H. Solomon. Synchronization of oscillating reactions in an extended fluid system. *Physical Review E*, 96(124101), 2006.

[5] K.D. Wells. *The Ecology and Behavoir of Amphibians*. The University of Chicago Press, Chicago, 2007.

[6] S. H. Strogatz and I. Stewart. Coupled oscillators and biological synchronization. *Scientific American*, 269(6):102–109, 1993.

[7] L. Glass, M. R. Guevara, A. Shrier, and R. Perez. Bifurcation and chaoas in a periodically stimulated cardiac oscillator. *Physica D: Nonlinear Phenomena*, 7(1-3):89–101, 1983. doi: 10.1016/0167-2789(83)90119-7.

[8] R.Y. Moore. A clock for the ages. *Science*, 284(5423):2102–2103, 1999.

[9] J. Bowers, J. Pycock, and J. O'Brien. Talk and embodiment in collaborative virtual environments. *Proceedings of ACM Conference on Human Factors in Computing Systems (CHI)*, pages 58–65, 1996.

[10] M. Inoue, T. Irino, N. Furuyama, R. Hanada, T. Ichinomiya, and H. Massaki. Manual and accelerometer analysis of head nodding patterns in goal-oriented dialogues. In *Proceedings of the International Conference on Human-Computer Interaction (HCI)*, pages 259–267, 2011.

[11] C. Breazeal. *Sociable Machines: Expressive Social Exchange Between Humans and robots.* PhD thesis, Dept. of Electrical Engineering and Computer Science, MIT, 2000.

[12] J. Solis, K. Taniguchi, T. Ninomiya, K. Petersen, T. Yamamoto, and A. Takanishi. Implementation of an auditory feedback control system on an anthropomorphic flutist robot inspired on the performance of a professional flutist. *Advanced Robotics*, 23(14):1849–1871, 2009.

[13] T. J. Walker. Acoustic synchrony: Two mechanisms in the snowy tree cricket. *Science*, 166:891–894, 1969.

[14] I. Aihara. Modeling synchronized calling behavior of Japanese tree frogs. *PPhysical Review E*, 8:011918, 2009. doi: 10.1103/PhysRevE.80.011918.

[15] T. Otsuka, T. Mizumoto, K. Nakadai, T. Takahashi, K. Komatani, T. Ogata, and H. G. Okuno. Music-ensemble robot that is capable of playing the theremin while listening to the accompanied music. In *Proceedings of International Conference on Industorial, Engineering and Other Applications of Applied Intelligent Systems (IEA/AIE)*, pages 102–112, 2010.

[16] A. Hyvarinen, J. Karhunen, and E. Oja. *Independent Component Analysis.* Wiley-Interscience, New York, 2001.

[17] E. Singer, J. Feddersen, C. Redmon, and B. Bowen. LEMUR's musical robots. In *Proceedings of Conference on New Interfaces for Musical Expression (NIME)*, pages 181–184, 2004.

[18] Y. Kaneko, K. Mizutani, and K.Nagai. Pitch controller for automatic trombone blower. In *Proceedings of International Symposium on Musical Acoustics (ISMA)*, pages 5–8, 2004.

[19] S. Sugano and I. Kato. WABOT-2: Autonomous robot with dexterous finger-arm - finger-arm coordination control in keyboard performance -. In *Proceedings of IEEE International. Conference on Robotics and Automation (ICRA)*, pages 90–97, 1987.

[20] J. Solis, T. Ninomiya, K. Petersen, M. Takeuchi, and A. Takanishi. Development of the anthropomorphic saxophonist robot WAS-1: Mechanical design of the simulated organs and implementation of air pressure feedback control. *Advanced Robotics*, 24(5-6):629–650, 2010.

[21] Y. Ota. Partner robots - from development to business implementation. In Z. Hippe, J. Kulikowski, and T. Mroczek, editors, *Human-Computer Systems Interaction: Backgrounds and Applications 2*, volume 99 of *Advances in Intelligent and Soft Computing*, pages 31–39. Springer, 2010.

[22] K. Shibuya. Violin playing robot and kansei. In J. Solis and K. Ng, editors, *Musical Robots and Interactive Multimodal Systems*, volume 74 of *Springer Tracts in Advanced Robotics*, pages 179–193. Springer, 2011.

[23] G. Weinberg and S.Driscoll. The interactive robotic percussionist - new developments in form, mechanics, perception and interaction design. In *Proceedings of ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, pages 456–461, 2007.

[24] G. Hoffman and G. Weinberg. Shimon: An interactive improvisational robotic marimba player. In *Proceedings of ACM Conference on Human Factors in Computing Systems (CHI)*, pages 3097–3102, 2010.

[25] S. Kotosaka and S. Shaal. Synchronized robot drumming by neural oscillator. *Journal of Robotics Society of Japan*, 19(1):116–123, 2001.

[26] S. Nakaoka, A. Nakazawa, F. Kanehiro, K. Kaneko, M. Morisawa, H. Hirukawa, and K. Ikeuchi. Learning from observation paradigm: Leg task models for enabling a biped humanoid robot to imitate human dances. *International Journal of Robotics Research*, 26(8):829–844, 2007.

[27] T. Nakata, T. Mori, and T. Sato. Analysis of impression of robot bodily expression. *J. of Robotics and Mechatronics*, 14(1):24–36, 2002.

[28] K. Yoshii, K. Nakadai, T. Torii, Y. Hasegawa, H. Tsujino, K. Komatani, T. Ogata, and H. G. Okuno. A biped robot that keeps steps in time with musical beats while listening to music with its own ears. In *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1743–1750, 2007.

[29] K. Murata, K. Nakadai, K. Yoshii, R. Takeda, T. Torii, H. G. Okuno, Y. Hasegawa, and H. Tsujino. A robot singer with music recognition based on real-time beat tracking. In *Proceedings of International Society for Music Information Retrieval Conference (ISMIR)*, pages 199–204, 2008.

[30] M. Goto. An audio-based real-time beat tracking system for music with or without drum-sounds. *Journal of New Music Research*, 30(2):159–171, 2001.

[31] K. Kosuge, T. Takeda, Y. Hirata, M. Endo, M. Nomura, K. Sakai, M. Koizumi, and T. Oconogi. Partner ballroom dance robot -PBDR-. *SICE Journal of Control, Measurement and System Integration*, 1(1):74–80, 2008.

[32] M. P. Michalowski, S. Sabanovic, and H. Kozima. A dancing robot for rhythmic social interaction. In *Proceedings of ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, pages 89–96, 2007. doi: 10.1145/1228716.1228729.

[33] T. Mizumoto, R. Takeda, K. Yoshii, K. Komatani, T. Ogata, and H. G. Okuno. A robot listens to music and counts its beats aloud by separating music from counting voice. In *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1538–1543, 2008.

[34] K. Murata, K. Nakadai, K. Yoshii, R. Takeda, T. Torii, and H. G. Okuno. A robot uses its own microphone to synchronize its steps to musical beats while scatting and singing. In *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 2459–2464, 2008.

[35] S. Kajita, T. Nakano, M. Goto, Y. Matsusaka, S. Nakaoka, and K. Yokoi. VocaWatcher: Natural singing motion generator for a humanoid robot. In *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2011.

[36] D. Overholt, J. Thompson, L. Putanam, B. Bell, J. Kleban, B. Sturm, and J. Kuchera-Morin. A multimodal system for gesture recognition in interactive music performance. *Computer Music Journal*, 33(4):69–82, 2009.

[37] A. Smirnov. Music and gesture: Sensor technologies in interactive music and the theremin based space control systems. In *Proceedings of International Computer Music Conference (ICMC)*, pages 511–514, 2000.

[38] A. Alford, S. Northrup, K. Kawamura, K w. Chan, and J. Barile. A music playing robot. In *Proceedings of International Conference on Field and Service Robotics (FSR)*, pages 29–31, 1999.

[39] F. van der Hulst. Robotic theremin player. In *Proceedings of National Advisory Committee on Computing Qualifications*, page 534, 2004.

[40] T. Mizumoto, H. Tsujino, T. Takahashi, T. Ogata, and H. G. Okuno. Thereminist robot: Development of a robot theremin player with feedforward and feedback arm control based on a theremin's pitch model. In *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 2297–2302, 2009.

[41] Y. Wu, P. Kuvinichkul, P.Y.K. Cheung, and Y. Demiri. Towards anthropomorphic robot thereminist. In *Proceedings of International Conference on Robotics and Biomimetics (ROBIO)*, pages 235–240, 2010.

[42] W. E. Frederickson. Band musicians' performance and eye contact as influenced by loss of a visual and/or aural stimulus. *J. of Research in Music Education*, 42(4):306–317, 1994.

[43] M. M. Wanderley. Non-obvious performer gestures in instrumental music. *Gesture-based communication in human-compter interaction*, 1739:37–48, 1999.

[44] W. F. Thompson, F. A. Russo, and L. Quinto. Audio-visual integration of emotional cues in song. *Cognition and Emotion*, 22(8):1457–1470, 2008.

[45] J. Mac Ritchie, B. Buck, and N. J. Bailey. Visualizing musical structure through performance gesture. In *Proceedings of International Society for Music Information Retrieval Conference (ISMIR)*, pages 237–242, 2009.

[46] H. Haken, J. A. S. Kelso, and H. Bunz. A theoretical model of phase transitions in human hand movements. *Biological Cybernetics*, 51:347–356, 1985.

[47] E. W. Large and M. R. Jones. The dynamics of attending: How people track time-varying events. *Psychological Review*, 106(1):119–159, 1999.

[48] C V. Buhusi and W. H Meck. What makes us tick? functional and neural mechanisms of interval timing. *Nature Reviews Neuroscience*, 6:755–765, 2005.

[49] R. J. Zatorre, J. L. Chen, and V. B. Penhune. When the brain plays music: auditory-motor interactions in music perception and production. *Nature Reviews Neuroscience*, 8:547–558, 2007.

[50] T. Mizumoto, T. Otsuka, K. Nakadai, T. Takahashi, K. Komatani, T. Ogata, and H. G. Okuno. Human-robot ensemble between robot thereminist and human percussionist using coupled oscillator model. In *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1957–1963, 2010.

[51] S. Shaal, S. Kotosaka, and D. Sternad. Nonlinear dynamical systems as movement primitives. In *Proceedings of IEEE/RAS International Conference on Humanoid Robots (Humanoids)*, 2001.

[52] E. W. Large. Beat tracking with a nonlinear oscillator. In *Proceedings of IJCAI Workshop on Artificial Intelligence and Music*, pages 24–31, 1995.

[53] J. Braasch. A cybernetic model approach for free jazz improvisations. *Kybernetes*, 40(7):984–994, 2011.

[54] N. Pecenka and P. E. Keller. Auditory pitch imagery and its relationship to musical synchronization. In *The Neurosciences and Music III: Disorders and Plasticity*, pages 282–286. New York Academy of Sciences, 2009.

[55] P. E. Keller. Joint action in music performance. In *Enacting Intersubjectivity: A Cognitive and Social Perspective on the Study of Interactions*, pages 205–221. IOS Press, Amsterdam, 2008.

[56] R. B. Dannenberg. An on-line algorithm for real-time accompaniment. In *Proceedings of International Computer Music Conference (ICMC)*, pages 193–198, 1984.

[57] C. Raphael. A probabilistic expert system for automatic musical accompaniment. *Journal of Computational and Graphical Statistics*, 10(3):487–512, 2001.

[58] I. Simon, D. Morris, and S. Basu. MySong: Automatic accompaniment generation for vocal melodies. In *Proceedings of ACM Conference on Human Factors in Computing Systems (CHI)*, pages 725–734, 2008.

[59] M. Goto, I. Hidaka, H. Matsumoto, Y. Kuroda, and Y. Muraoka. A jazz session system for interplay among all players - VirJa session (virtual jazz session system). In *Proceedings of International Computer Music Conference (ICMC)*, pages 346–349, 1996.

[60] M. Hamanaka, M. Goto, H. Asoh, and N. Otsu. A learning-based jam session system that imitates a player's personality model. In *Proceedings of International Joint Conference on Artificial Intelligence (IJCAI)*, 2003.

[61] K. Petersen, J. Solis, and A. Takanishi. Musical-based interaction system for the Waseda Flutist Robot: implementation of the visual tracking interaction module. *Autonomous Robotics Journal*, 28(4):439–455, 2010.

[62] A. Lim, T. Mizumoto, T. Ogata, and H.G. Okuno. A musical robot that synchronizes with a co-player using non-v erbal cues. *Advanced Robotics*, 26(3-4):363–381, 2012.

[63] T. Itohara, T. Otsuka, T. Mizumoto, A. Lim, T. Ogata, and H.G. Okuno. A multi-modal tempo and beat tracking system based on audio-visual information from live guitar performances. *EURASIP Journal on Audio, Speech and Music Processing*, 2012. doi:10.1186/1687-4722-2012-6.

[64] G. Weinberg, B. Blosser, T. Mallikarjuna, and A. Raman. The creation of a multi-human, multi-robot interactive jam session. In *Proceedings of Conference on New Interfaces for Musical Expression (NIME)*, pages 70–73, 2009.

[65] H.C. Gerhardt and F. Huber. *Acoustic Communication in Insects and Anurans.* The University of Chicago Press, Chicago, 2002.

[66] P. M. Narins and R. R. Capranica. Communicative significance of the two-note call of the treefrog *Eleutherodactylus coqui.* *Journal of Comparative Physiology A*, 127:1–9, 1978.

[67] A. M. Simmons. Call recognition in the bullfrog, *Rana catesbiana*: Generalization along the duration continuum. *Journal of Acoustic Society of America*, 115(3):1345–1355, 2004.

[68] D. N. Suggs and A. M. Simmons. Information theory analysis of patterns of modulation in the advertisement call of the male bullfrog, *Rana catesbiana.* *Journal of Acoustic Society of America*, 117(4):2330–2337, 2005.

[69] A.S. Feng, P. M. Narins, C-H Xu, W-Y Lin, Z-L Yu, Q. Qiu, Z-M, and J-X Shen. Ultrasonic communication in frogs. *Nature*, 440:2333–2336, 2006.

[70] P. M. Narins, A. S. Feng, R. R. Fay, and A. N. Popper, editors. *Hearing and Sound Communication in Amphibians.* Springer, 2007.

[71] B. Hedwig and JFA Poulet. Complex auditory behavior emerges from simple reactive steering. *Nature*, (430):781–785, 2004.

[72] J. A. Simmons, M. B. Fenton, and M. J. O'Farrell. Echolocation and persuit of prey by bats. *Science*, 203(4375):16–21, 1979.

[73] D. R. Griffin. *Listening in the dark: The acoustic orientation of bats and men.* Comstock Publishing Associates, 1986.

[74] H. Riquimaroux, S. J. Gaioni, and N. Suga. Cortical computational maps control the auditory perception. *Science*, (251):565–568, 1991.

[75] J. A. Thomas, C. F. Moss, and M. Vater, editors. *Echolocation in Bats and Dolphins*. University of Chicago, 2002.

[76] M. Kashino and T. Hirahara. One, two, many – judging the number of concurrent talkers. *Journal of Acoustic Society of America*, 99(4):2596–2603, 1996.

[77] J. Okuyama, K. Kataoka, K. M Kobayashi, O. Abe, K. Yoseda, and N. Arai. The regularity of dive performance in sea turtles: a new perspective from precise activity data. *Animal Behaviour*, 84(2):349–359, 2012.

[78] R. MacCurdy and K. Fristrup. Automatic animal tracking using matched filters and time difference of arrival. *Journal of Communication*, 4(7):487–495, 2009.

[79] P. Tyack. An optical telemetry device to identify which dolphin produces a sound. *Journal of Acoustic Society of America*, 78(5):1892–1895, 1985.

[80] A. M. Simmons, J. A. Simmons, and M. E. Bates. Analyzing acoustic interactions in natural bullfrog (*Rana catesbeiana*) choruses. *Journal of Comparative Physiology*, 122(3):274–282, 2008.

[81] D. L. Jones and R. Ratnam. Blind location and separation of callers in a natural chorus using a microphone array. *Journal of Acoustic Society of America*, 126(2):895–910, 2009.

[82] J. L. Spiesberger and K. M. Fristrup. Passive localization of calling animals and sensing of their acoustic environment using acoustic tomography. *American Naturalist*, 135(1):107–153, 1990.

[83] T. U. Grafe. Costs and benefits of mate choice in the lek-breeding reed frog, *Hyperolius marmoratus. Animal Behavior*, 53:1103–1117, 1997.

[84] J. L. Spiesberger. Locating animals from their sounds and tomography of the atmosphere: Experimental demonstration. *Journal of Acoustic Society of America*, 106(2):837–846, 1999.

[85] C. W. Clark and W. T. Ellison. Calibration and comparison of the acoustic location methods used during the spring migration of the bowhead whale, *Balaena mysticetus*, off Pt. Barrow, Alaska, 1984−1993. *Journal of Acoustic Society of America*, 107(6):3509–3517, 2000.

[86] J. J. Schwartz. *Call monitoring and interactive playback systems in the study of acoustic interactions among male anurans*, chapter 14, pages 183–204. Smithonian Institution Press, Washington and Chigaco, 2001.

[87] D. T. Blumstein, D. J. Mennill, P. Clemins, L. Girod, K. Yao, G. Patricelli, J. L. Deppe, A. H. Krakauer, C. Clark, K. A. Cortopassi, S. F. Hanser, B. McCowan, A. M. Ali, and A. N. G. Kirschel. Acoustic monitoring in terrestrial environments using microphone arrays: applications, technological considerations and prospectus. *Journal of Applied Ecology*, 48(3):758–767, 2011.

[88] H. Sawada, R. Mukai, and S. Araki. Polar coordinate based nonlinear function for frequency-domain blind source separation. *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, 86(3):590–596, March 2003.

[89] F. Asano, H. Asoh, and T. Matsui. Sound source localization and signal separation for office robot "JiJo-2". In *Proceedings of Multisensor Fusion and Integration for Intelligent Systems*, pages 243–248, 1999.

[90] H. Nakajima, K. Nakadai, Y. Hasegawa, and H. Tsujino. Blind source separation with prameter-free adaptive step-size method for robot audition. *IEEE Trans. on Audio, Speech, and Language Processing*, 18(6):1476–1484, 2010.

[91] K. Nakadai, H. G. Okuno, H. Nakajima, Y. Hasegawa, and H. Tsujino. Design and implementation of robot audition system "HARK". *Advanced Robotics*, 24:739–761, 2009. doi:10.1163/016918610X493561.

[92] J. Solis, K. Taniguchi, T. Ninomiya, T. Yamamoto, and A. Takahashi. Development of Waseda flutist robot WF-4RIV: Implementation of auditory feedback system. In *Proceedings of IEEE International. Conference on Robotics and Automation (ICRA)*, pages 3654–3659, 2008.

[93] J. Solis, K. Petersen, T. Ninomiya, M. Takeuchi, and A. Takanishi. Development of anthropomorphic musical performance robots: From understanding the nature of music performance to its application to entertainment robotics. In *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 2309–2314, 2009.

[94] S. J. Julier and J. K. Uhlmann. Unscented filtering and nonlinear estimation. *Proceedings of the IEEE*, 92(3):401–422, 2004.

[95] G. Welch and G. Bishop. An introduction to the kalman filter. In *SIGGRAPH Course*, number 8, 2001.

[96] G. Kitagawa. Monte carlo filter and smoother for non-gaussian nonlinear state space models. *Journal of Computational and Graphical Statistics*, 5(1):1–25, 1996.

[97] A. V. Glinsky. *The Theremin in the Emergence of Electronic Music*. PhD thesis, New York Univ., 1992.

[98] D.W. Marquardt. An algorithm for least-squares estimation of nonlinear parameters. 11(2):431–441, 1963.

[99] A. Camacho. *SWIPE: A sawtooth waveform inspired pitch estimator for speech and music.* PhD thesis, University of Florida, 2007.

[100] N. Schuett. The effects of latency on ensemble performance. Technical report, Department of Music, Stanford University, 2002.

[101] M. Gurevich, C. Chafe, G. Leslie, and S. Tyan. Simulation of networked ensemble performance with varying time delays: characterization of ensemble accuracy. In *Proceedings of International Computer Music Conference (ICMC)*, 2004.

[102] A. Lim, T. Mizumoto, L. Cahier, T. Otsuka, T. Takahashi, K. Komatani, T. Ogata, and H. G. Okuno. Robot musical accompaniment: Integrating audio and visual cues for real-time synchronization with a human flutist. In *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1964–1969, 2010.

[103] R.R. Yager. On ordered wighted averaging aggregation operators in multicriteria decision making. *IEEE Transactions on Systems, Man, and Cybernetics*, 18(1):183–190, 1988.

[104] E. H.-Viedma, F. Herrera, and F. Chiclana. A consensus model for multiperson decision making with different preference structures. *IEEE Transactions on Systems, Man, and Cybernetics*, 32(3):394–402, 2002.

[105] V D. Blondel, J. M. Hendrickx, and J. N. Tsitsiklis. On krause's multi-agent consensus model with state-dependent connectivity. *IEEE Transactions on Automatic Control*, 54(11):2586–2597, 2009.

[106] P. E. Keller. Individual differences, auditory imagery, and the coordination of body movements and sounds in musical ensembles. *Music Perception*, 28(1):27–46, 2010.

[107] M. H. Thaut, R. A. Miller, and L. M. Schauer. Multiple synchronization strategies in rhythmic sensorimotor tasks: phase vs period correction. *Biological Cybernetcis*, 79:241–250, 1998.

[108] B. H. Repp and H. Jendoubi. Flexibility of temporal expectations for triple subdivision of a beat. *Advances in Cognitive Psychology*, 5:27–41, 2009.

[109] E. W. Large and C. Palmer. Perceiving temporal regularity in music. *Cognitive Sceince*, 26:1–37, 2002.

[110] J. P. Hailman and R. G. Jaeger. Phototactic responses to spectrally dominant stimuli and use of color vision by adult anuran amphibians: a comparative survey. *Animal Behavior*, 22:757–795, 1974.

[111] T. Mizumoto, I. Aihara, T. Otsuka, R. Takeda, K. Aihara, and H. G. Okuno. Sound imaging of nocturnal animal calls in their natural habitat. *Journal of Comparative Physiology A*, 197(9):915–921, 2011.

[112] ITU-R. *Recommendation ITU-R BT.606-6: Studio encoding parameters of digital television for standard 4:3 and wide screen 16:9 aspect ratios.* International Telecommunication Union Radiocommunication Sector, 2007.

[113] N. Otsu. A threshold selection method from gray-level histograms. *IEEE Transactions on Systems, Man, and Cybernetics*, SMC-9(1):62–66, 1979. doi: 10.1109/TSMC.1979.4310076.

[114] I. Aihara, R. Takeda, T. Mizumoto, T. Otsuka, T. Takahashi, H. G. Okuno, and K. Aihara. Complex and transitive synchronization in a frustrated system of calling frogs. *Physical Review E*, 83(3):031913 (5 pages), 2011.

[115] M. Matsui. *Natural History of the Amphibia*, pages 150–152. University of Tokyo Press, 1996.

[116] N. Maeda and M. Matsui. *Frogs and Toads of Japan*, pages 36–39. Bun-ichi Sogo Shuppan Co. Ltd., 1999.

[117] K.V. Mardia and P. E. Jupp. *Directional Statistics*. Wiley, 2000.

[118] M. D. Greenfield and A. S. Rand. Frogs have rules: Selective attention algorithms regulate chorusing in *Physalaemus* pustulosus (*Loptodactylidae*). *Ethology*, 106:331–347, 2000.

[119] J. S. Brush and P. M. Narins. Chorus dynamics of a neurotopical amphibian assemblage: Comparison of computer simulation and antural behavior. *Animal Behavior*, 37(33):33–44, 1989.

[120] J. Lanslots, F. Deblauwe, and K. Janssens. Selecting sound source localization techniques for industorial applications. *Sound & Vibration*, June:6–9, June 2010.

[121] S. Haykin. *Adaptive Filter Theory*. Prentice Hall, New Jersey, 4th edition, 2001.

[122] S. Horai, I. Aihara, and K. Aihara. Time series analysis of sound data on interactive calling behavior of japanese tree frogs. *IEEJ Trans. on Electronics, Information and Systems*, 117(10):1692–1698, 2007.

[123] R.B. Dannenberg and C. Raphael. Music score alignment and computer accompaniment. *Communications of the ACM*, 49:38–43, 2006.

[124] T. Otsuka, K. Nakadai, T. Takahashi, T. Ogata, and H. G. Okuno. Real-time audio-to-score alignment using particle filter for co-player music robots. *EURASIP Journal on Advances in Signal Processing*, 2011. doi: 10.1155/2011/384651.

[125] D. R. Berman. *AVISARME: Audio Visual Synchronization Algorithm for a Robotic Musician Ensemble*. PhD thesis, University of Maryland, 2012.

[126] C. F. Cadieu and K. Koepsell. Phase coupling estimation from multivariate phase statistics. *Neural Computation*, 22:3107–3126, 2010.

[127] M. Timme. Revealing network connectivity from response dynamics. *Physical Review Letters*, 98(224101), 2007. doi: 10.1103/PhysRevLett.98.224101.

[128] H. Nocke. Physiological aspects of sound communication in crickets (*Gryllus campestris* L.). *Journal of Comparative Physiology*, 80:141–162, 1972.

[129] T.Mizumoto, H. Awano, I. Aihara, T. Otsuka, and H. G. Okuno. Sound imaging system for visualizing multiple sound sources from two species. In *Tenth International Congress of Neuroethology*, 2012.

List of Publications

Journal Papers

1) **Takeshi Mizumoto**, Hiroshi Tsujino, Toru Takahashi, Kazunori Komatani, Tetsuya Ogata, and Hiroshi G. Okuno: Development of a Theremin Player Robot Based on Arm-Position-to-Pitch and -Volume Models (Title in Japanese: テルミンの音高・音量特性のモデルに基づくテルミン演奏ロボットの開発), Journal of Information Processing Society of Japan, (in Japanese) Vol. 51, No. 10, pp.2007-2019, 2010. → **Chapter 3**.

2) **Takeshi Mizumoto**, Ikkyu Aihara, Takuma Otsuka, Ryu Takeda, Kazuyuki Aihara, and Hiroshi G. Okuno: Sound Imaging of Nocturnal Animal Calls in Their Natural Habitat, Journal of Comparative Physiology A, Vol. 197 No. 9, pp. 915-921, 2011. DOI: 10.1007/s00359-011-0652-7 → **Chapter 6**.

3) **Takeshi Mizumoto**, Ikkyu Aihara, Hiromitsu Awano, Takuma Otsuka, Hiroshi G. Okuno: Sound-to-Light Conversion Devices to Visualize Acoustic Communication among Small Nocturnal Animals, Bioacoustics (*under review*) → **Chapter 6**.

International Conference (Peer-reviewed)

1) **Takeshi Mizumoto**, Ryu Takeda, Kazuyoshi Yoshii, Kazunori Komatani, Ttsuya Ogata, Hiroshi G. Okuno: A Robot Listens to Music and Counts Its Beats Aloud by Separating Music from Counting Voice, *Proceedings*

of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS-2008), pp. 1538-1543, Nice, France, Sept. 2008.

2) **Takeshi Mizumoto**, Hiroshi Tsujino, Toru Takahashi, Tetsuya Ogata, Hiroshi G. Okuno: Thereminist Robot: Development of a Robot Theremin Player with Feedforward and Feedback Arm Control based on a Theremin's Pitch Model (Invited paper), *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS-2009)*, pp. 2297-2302, St. Louis, U.S., Oct. 2009. → **Chapter 3**.

3) **Takeshi Mizumoto**, Takuma Otsuka, Kazuhiro Nakadai, Toru Takahashi, Kazunori Komatani, Tetsuya Ogata, Hiroshi G. Okuno: Human-Robot Ensemble between Robot Thereminst and Human Percussionist using Coupled Oscillator Model, *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS-2010)*, pp.1957-1962, Taipei, Oct. 2010. → **Chapter 4**.

4) **Takeshi Mizumoto**, Angelica Lim, Takuma Otsuka, Kazuhiro Nakadai, Toru Takahashi, Tetsuya Ogata, Hiroshi G. Okuno: Integration of flutist gesture recognition and beat tracking for human-robot ensemble, *Proceedings of IEEE/RSJ Workshop on Robots and Musical Expression*, , Taipei, Taiwan, Oct. 2010. → **Chapter 4**.

5) **Takeshi Mizumoto**, Kazuhiro Nakadai, Takami Yoshida, Ryu Takeda, Takua Otsuka, Toru Takahashi, Hiroshi G. Okuno: Design and Implementation of Selectable Sound Separation on the Texai Telepresene System using HARK *Proceedings of IEEE-RAS International Conference on Robotics and Automation (ICRA-2011)*, pp.2130-2137, Shanghai, China, May, 2011.

6) **Takeshi Mizumoto**, Toru Takahashi, Tetsuya Ogata, Hiroshi G. Okuno: Adaptive Pitch Control for Robot Thereminist using Unscented Kalman Filter. H. Jiang, M. Ali, and M. Li (Eds.), Modern Advances in Intelligent Sys-

tems and Tools, Studies in In Computational Intelligence (IEA/AIE-2012), pp.19-24, Springer, Dalian, China, June, 2012. → **Chapter 3**.

7) **Takeshi Mizumoto**, Tetsuya Ogata, Hiroshi G. Okuno: Who is the leader in a multiperson ensemble? —Multiperson human-robot ensemble model with leaderness—, *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS-2012)*, pp.1413-1419, Vilamoura, Algarve, Portgul, October, 2012. → **Chapter 5**.