IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received December 20, 2019, accepted January 18, 2020, date of publication January 24, 2020, date of current version January 31, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2969277

Model-Free Control for Dynamic-Field Acoustic Manipulation Using Reinforcement Learning

KOUROSH LATIFI, ARTUR KOPITCA, AND QUAN ZHOU, (Member, IEEE)
Department of Electrical Engineering and Automation, Aalto University, 02150 Espoo, Finland

Corresponding author: Quan Zhou (quan.zhou@aalto.fi)

This work was supported in part by the Academy of Finland under Grant 296250 and Grant 328239, in part by the Aalto Doctoral School of Electrical Engineering, and in part by FinELib consortium.

ABSTRACT Dynamic-field acoustic manipulation techniques benefit numerous applications in microsystem assembly, pattern formation, biological research, tissue engineering, and lab-on-a-chip. These techniques generally rely on a theoretical dynamic model of particle motion in the acoustic field. Accordingly, success of the manipulation task highly depends on the accuracy of the employed dynamic model. However, modelling such dynamic behavior is a great challenge in more advanced acoustic manipulation devices and requires significant simplifications. Here, we introduce a model-free control method based on reinforcement learning for highly-dynamic acoustic manipulation devices. In our method, the controller does not need a prior knowledge of the acoustic field and learns the optimal control policy for each manipulation task by merely interacting with the acoustic field. As a proof-of-concept, we apply our method to a classic acoustic manipulation device, a Chladni plate consisting of a centrally-actuated vibrating plate. By employing the controller, we demonstrate successful manipulation of single and multiple particles towards target locations on the plate surface. The model-free control method is not limited to the Chladni plate and can be potentially applied to a broad range of acoustic manipulation devices as well as other forms of field-based micromanipulation systems, where accurate theoretical modelling of the field is challenging.

INDEX TERMS Acoustic manipulation, Chladni plate, dynamic-filed acoustic device, model-free control, real-time control, reinforcement learning.

I. INTRODUCTION

Contactless transport and handling of matter is of paramount importance in numerous scientific and technological applications. Typical methods of contactless material handling rely on electromagnetic principles, e.g., electrostatic [1], magnetic [2], and optical [3] methods. They offer interesting capabilities but also impose clear limitations. In particular, they are subjected to inherent requirements of specific material properties: magnetic techniques are mostly limited to magnetic particles, optical techniques need a transparent environment to operate, and electrostatic techniques are subjected to electrostatic property dependencies. In contrast to electromagnetic-based methods, acoustic manipulation methods move objects by sound and are material-independent [4]. Those methods are mechanical in nature, providing unique capabilities for contactless material handling.

The associate editor coordinating the review of this manuscript and approving it for publication was Yang Tang.

Acoustic manipulation has been rapidly developed during the last decade, enabling a broad range of applications in the biomedical research [5], particle manipulation [6]–[8], pattern formation [9], microassembly [10], and lab-on-a-chip [11]. Traditionally, acoustic manipulation devices operate by forming simple patterns of standing pressure waves on a solid surface [12], in a chamber or channel [6], [11], [13]–[15]. Those classical acoustic manipulation devices can facilitate a limited set of manipulation tasks such as forming parallel lines or dots of particles [6], [12], [13]. More advanced manipulation tasks, e.g., manipulating particles along user-specific trajectories, require more versatile and reconfigurable devices [11].

The last decade has also seen the emergence of a new class of acoustic manipulation devices which capitalize on dynamic acoustic fields, commonly referred to as dynamic-field acoustic manipulation devices [11]. Generally, those devices are able to form acoustic traps which attract and hold the nearby objects. To move the acoustic traps and

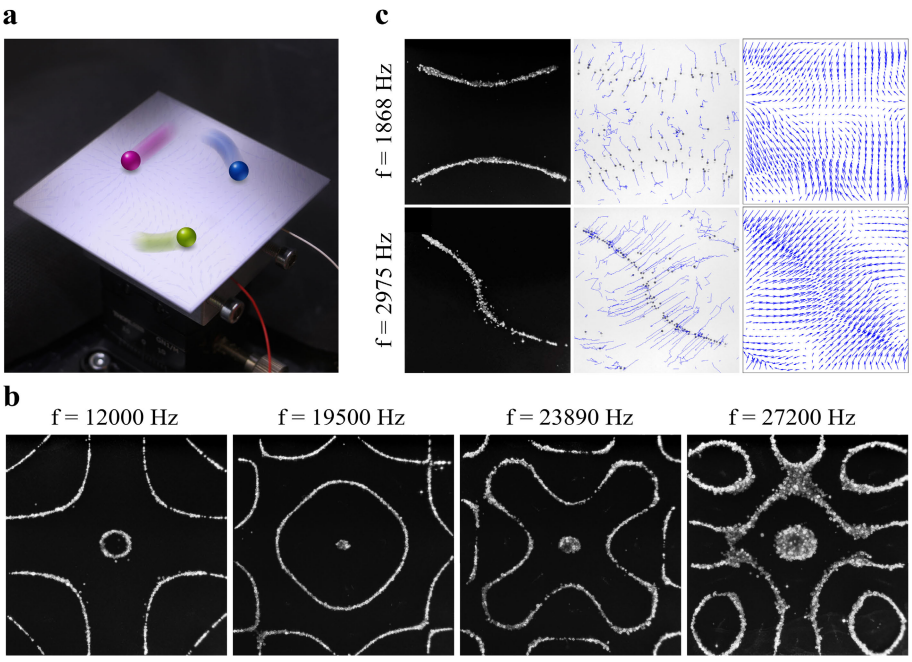

FIGURE 1. (a) Schematic of a Chladni plate: When excited, a Chladni plate can generate complex two-dimensional force fields and move particles over its surface. (b) Experimental Chaldni patterns: A Chladni plate is able to generate diverse force fields resulting in specific patterns at each resonant frequency. (c) Pattern formation at non-resonant frequencies: A Chladni plate is also able to generate complex patterns at non-resonant frequencies resulting in diverse motion of particles. The shape of the patterns are highly related to the motion of the particles over the plate (©2017 IEEE Reprinted with permission from [18]).

accordingly the objects, those devices dynamically reshape the acoustic field by either controlling the phase of hundreds of transducers [16], [17] or switching between several resonant modes [5]. Those devices have shown remarkable manipulation capabilities, such as moving biological organisms along pre-defined trajectories [5], or manipulating multiple objects independently [17]. To operate in the whole workspace, the devices need to create trapping points in that space, which often requires complex hardware with hundreds of transducers [16], [17].

Recently, the concept of out-of-trap acoustic manipulation has been introduced [7], [8], [18] where the objects can be manipulated directly using the acoustic force field instead of the trapping points or lines. This has considerably simplified the hardware requirements, and even a single-transducer device is shown to be sufficient to peform complex tasks such as manipulating multiple particles or a swarm of particles along user-specific trajectories [7], [8].

Predicting the particle motion in a dynamic-field acoustic device is of great difficulty due to the fact that the generated acoustic fields are spatially highly-complex. Typically, those devices rely on the theoretical dynamic model of particle motion inside the acoustic field. Modelling such dynamic behavior is challenging, and requires major assumptions and simplifications. It also implies that if those dynamic models are not detailed enough, the manipulation task cannot be performed accurately, or would fail. More recently, data-driven methods have been suggested to predict the particle motion [7], [18]. However, a substantial effort is required to collect experimental data for building a motion model of the device at different frequencies. In those experiments, tens of particles should be distributed in the whole workspace, and their motion after exciting the acoustic field at different frequencies should be recorded. Performing such experiments is time-consuming and could be practically challenging for certain acoustic manipulation devices.

In this study, we introduce a novel approach to tackle these problems. We consider the dynamic-field acoustic manipulator as a robotic system, enabling us to leverage recent advances in machine learning for optimal robot control. We then propose a model-free method for controlling the particle motion that does not require a prior model of the acoustic field. By merely interacting with the field, the proposed control method learns the optimal control policy using reinforcement learning (RL). We apply the control method to a Chladni plate (Fig. 1a), a classic acoustic manipulation device, which can create frequency-dependent two-dimensional acoustic fields on the plate surface [7], [18] (see Section II-A for further explanation). We place the particle on the plate and excite the plate with various frequencies for a certain number of steps. Meanwhile, the RL controller collects observations and rewards corresponding to a specific target from the system. We perform multiple episodes of such experiments either with the real hardware

or a simulation framework, and use an RL algorithm similar to Neural Fitted Q Iteration (NFQ) [19] to learn an optimal control policy for the manipulation task. We observed that the performance of the controller improves with the number of episodes. By using the controller, we demonstrate successful manipulation of a single and multiple particles on the plate.

This paper is organized as follows. In Section II, we introduce the problem of motion control in a dynamic-field acoustic manipulation device, and formulate it in a Markov decision process (MDP) framework. In Section III, we explain the experimental setup. In Section IV, we explain the RL-based control method which includes the learning algorithm and the closed-loop control method. The experimental results to evaluate our control approach are presented in Section V. Finally, conclusions and perspectives of this work are discussed in Section VI.

II. PROBLEM FORMULATION

A. DYNAMIC-FIELD ACOUSTIC MANIPULATION ON A CHLADNI PLATE

We use a centrally-actuated Chladni plate as the apparatus for dynamic acoustic manipulation [7], [8]. A Chladni plate consists of a plate mounted on a vibrational source, as schematically shown in Fig. 1a. When the vibrational source is driven at a certain frequency, it generates flexural waves in the plate which create a two-dimensional force field over the surface of the plate. The shape of the force field depends on the driving frequency of the vibrational source. If one sprinkles particles on the plate, e.g., sand or salt, the force field moves the particles towards seemingly specific directions, and forms patterns. The patterns formed at the resonant frequencies of the plate are commonly called Chladni patterns (Fig. 1b). Pattern formation is not limited to resonant frequencies, and comparably complex patterns can also be formed at the non-resonant frequencies [7], [8], [18] (see Fig. 1c). As shown in Fig. 1c, the pattern shape is associated with the direction of the particle motion at each specific frequency. Such diverse frequency-dependent displacement fields potentially provide great capability for motion control.

To manipulate an object to a desired location on the plate, a controller is required to plan and control the motion using a proper force field related to a specific frequency [7], [8]. Such a controller could rely on a detailed model of the particle motion or force field on the plate. Since theoretical models, e.g. PDE eigenvalue solutions [20] or solving the two-dimensional inhomogeneous Helmholtz equation [21] cannot accurately predict the particle motion, extensive data-driven modelling efforts is needed to obtain the displacement field of the particles, e.g. hundreds of experiments with particles covering the whole workspace [7], [8].

In this study, we suggest a model-free approach for motion control. We first formulate the problem in an MDP framework (see Section II-B), and solve it using reinforcement learning algorithms (see Section IV-A) by manipulating the particles directly on the hardware device or using a simulator.

B. MARKOV DECISION PROCESS FRAMEWORK

Regardless of the hardware, we can formulate the problem of manipulating a particle towards a target point subjected to dynamic force fields in an MDP framework. In this section, we explain such formulation.

An MDP is described by the tuple $\langle \mathcal{S}, \mathcal{A}, p, R, \gamma \rangle$ with a continuous set $\mathcal{S}$ of states, a finite set $\mathcal{A}$ of actions, a transition probability function p, an immediate reward function R, and a discount factor γ. In our case, the two-dimensional positions of the particles on the plate represent the state of MDP, and the actuation signal frequency denotes the action. The transition probability function p sets the probability that action A_t (playing a signal with a specific frequency) in state S_t at time t will lead to state S_{t+1} at time $t+1$ as follows,

$$p(s, a) = \mathbb{P}[S_{t+1} = s' \mid S_t = s, A_t = a], \tag{1}$$

where s and a represent the state and the action at time instant t, and s' denotes the next resulted state. In our case, the transition probability function p describes the stochastic motion of the particles on the plate subjected to plate vibration at a specific frequency.

An immediate reward function $R : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to \mathbb{R}$ is defined assigning a reward $R(s, a, s')$ every time a transition from s to s' occurs after taking action a. The formulation of the reward function is explained in details in Section IV-A. We also fix a discount factor $\gamma \in [0, 1]$ which discounts the future rewards compared to the immediate reward. We define the return G_t as the total discounted reward from time-step t as

$$G_t = \sum_{k=0}^{\infty} \gamma^k R_{t+k+1}. \tag{2}$$

A policy $\pi : \mathcal{S} \to \mathcal{A}$ is defined as a distribution over actions given states. The policy function π sets the probability that action a is selected at state s at any time t as follows,

$$\pi(a \mid s) = \mathbb{P}[A_t = a \mid S_t = s]. \tag{3}$$

We define an action-value function Q, or the Q-value function, which is the expected return given state s and action a under policy π as

$$Q^{\pi}(s, a) = \mathbb{E}_{\pi}\left[G_t \mid S_t = s, A_t = a\right]. \tag{4}$$

Solving the MDP involves determining a policy π^* that maximizes the action-value function $Q(s, a)$ where the optimal action-value function Q^* is the maximum action-value function over all policies as

$$Q^*(s) = \max_{\pi}(Q(s, a)). \tag{5}$$

The optimal action-value function represents the expected total discounted reward along a trajectory starting at state s obtained by choosing a as the first action and following the optimal policy thereafter. The optimal policy is then defined as

$$\pi^*(s) = \arg\max_{a \in \mathcal{A}} Q^*(s, a). \tag{6}$$

The optimal policy gives the best possible action a at state s, which is essentially a map between the positions of the particles (state) and the frequency of the excitation signal (action). We then use the acquired optimal policy in a closed-loop control scheme to manipulate particles toward the desired target points.

III. APPARATUS

Figure 2 shows our experimental platform consisting of a Chladni plate. The plate has dimensions of 50 mm × 50 mm × 500 μm, diced from a silicon wafer and glued on a piezoelectric actuator (Piezomechanik, PSt 150/2×3/20) using cyanoacrylate adhesive. The piezoelectric actuator is mounted on a dual-axis goniometer (Thorlabs, GN2/M). We use pressed solder balls (Martin Smt/VD90.5106, Sb96.5Ag3Cu0.5, Ø600 μm) as manipulation specimens. The particle and plate are imaged from above by a video camera (ImperX, IGV-B1621C-KC000 with Infinity/InfiniMite Alpha lens). The camera is connected to an embedded controller (National Instruments, PXIe-8135) via an Ethernet interface module (National Instruments, PXIe-8234) to feedback the position of the particle. A strip of LEDs (NEXTEC, LS5300NWIP20) is mounted horizontally around and slightly above the plate for better vision contrast. The plate is excited with sinusoidal signals with a frequency in the range of 1-20 kHz from Chromatic musical scale (52 distinct frequencies). During particle manipulation, the frequency of the signal is selected by a closed-loop controller. The signal is then generated in an arbitrary waveform generator (AWG) (National Instruments, PXI-5412), amplified by a linear amplifier (Piezo Systems, EPA-104-230), and sent to the piezoelectric actuator. The embedded controller, the Ethernet interface module, and the AWG are mounted on a Chassis (National Instruments, PXIe-1071).

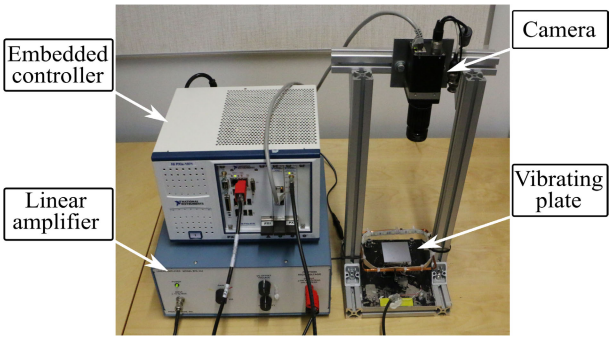

FIGURE 2. Experimental setup: A vibrating plate is mounted on a piezoelectric actuator. Location of the particle is recorded by a camera, and based on the current position of the particle, an embedded controller calculates a control signal that moves the particle towards the target position.

IV. CONTROL METHOD

In this section, we explain the main elements of our control method including the procedure for learning the optimal policy, and the motion control method using the acquired optimal policy.

Algorithm 1 NFQ Algorithm

Input: Starting position s_{init}, a set of actions $\mathcal{A}$, number of actions N_a, network structure of Q-value function Q_{init}, terminal positions including the target point and plate edges s_T, greedy action probability ϵ, training set $\mathcal{E}$, number of episodes N_e, number of steps N_t
Output: Q-value function Q_N
for $n = 1$ to N_a **do**
  $Q^n = Q_{init}$
  $\mathcal{E}^n \leftarrow \{\}$
end for
for $k = 1$ to N_e **do**
  $terminal = 0$
  $t = 1$
  $s \leftarrow s_{init}$
  while $terminal \neq 1$ and $t \leq N_t$ **do**
    Choose $a \in \mathcal{A}$ using ϵ-greedy algorithm
    Execute action a and record the new state s'
    Calculate the immediate reward R according to Equation 8
    if $s' = s_T$ **then**
      $terminal = 1$
    end if
    $input_t^k \leftarrow \langle s, a \rangle$
    $output_t^k \leftarrow R + \gamma \arg\max_{b \in \mathcal{A}} Q_{k-1}^a(s', b)$
    $\mathcal{E}^a \leftarrow \mathcal{E}^a + \langle input_t^k, output_t^k \rangle$
    $s \leftarrow s'$
    $t \leftarrow t + 1$
  end while
  for $n = 1$ to N_a **do**
    $Q_k^n \leftarrow \mathrm{LM}(\mathcal{E}^n)$
  end for
end for
$Q_N \leftarrow Q_{N_e}$

A. LEARNING ALGORITHM

We use reinforcement learning algorithms to solve the MDP for acoustic manipulation. In particular, we use an algorithm similar to NFQ [19] which is a variant of Q-learning algorithm with a neural network function approximator. Algorithm 1 shows our implemented reinforcement learning algorithm. The underlying idea of the algorithm is that the parameter update is performed off-line considering an entire set of experiences. Therefore, the algorithm consists of two major phases: the generation of the training set $\mathcal{E}$, and regression of action-value function Q within a multi-layer neural network. In the data generation phase, experiences are collected in the triples of the form $\langle s, a, s' \rangle$ by interacting with the real or simulated system, resulting in the set of experiences $\mathcal{E}$. In the regression phase, the regression algorithm is realized by a multi-layer perceptron.

The training process is performed in several episodes of experiments. In every episode, we first place the particle close to the starting position. Then, we excite the plate with

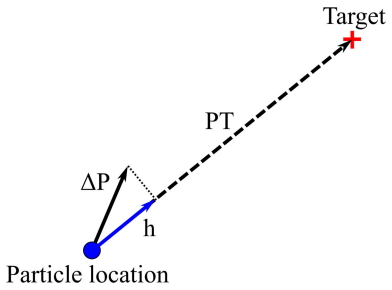

FIGURE 3. Immediate reward based on the projection of the displacement vector ΔP.

various actions for a finite number of steps N_t. In every step, we choose the action using an ϵ-greedy algorithm that follows the greedy policy with probability 1-ϵ and selects a random action with probability ϵ. We then record the state s' and calculate the immediate reward for the experience $\langle s, a, s' \rangle$.

To assign the immediate reward R after each experience, we first compute $\mathbf{h}$, the projection of the displacement vector $\Delta\mathbf{P}$ over a vector that connects the current particle position to the target location $\mathbf{PT}$ (Fig. 3) as follows,

$$\mathbf{h} = \frac{\mathbf{\Delta P} \cdot \mathbf{PT}}{\|\mathbf{PT}^2\|}\mathbf{PT}, \tag{7}$$

and then we calculate reward R according to the magnitude and the direction of the projection vector $\mathbf{h}$ as

$$R = \text{sgn}(\mathbf{h} \cdot \mathbf{PT})\|\mathbf{h}\|. \tag{8}$$

To estimate the optimal action-value function Q^*, we use a two-layer feed-forward neural network with sigmoid hidden neurons and linear output neurons for each note. The input tuple *input* of each training network consists of the state s_l and the action a_l of the training experience l. The output value is computed by the sum of the immediate reward R and the expected maximal trajectory rewards for the successor state s', computed on the basis of the current estimate of the action-value function Q_k as follows,

$$output_t^k \leftarrow R + \gamma \arg\max_{b \in \mathcal{A}} Q_{k-1}^a(s', b), \tag{9}$$

where k represents the episode number. After every episode of training, we update the weight and bias values of the networks Q^n for all actions according to Levenberg-Marquardt (LM) backpropagation method. We observed that Q^n converges after a certain number of episodes which gives us an approximation of the optimal action-value function Q^*.

B. CLOSED-LOOP CONTROLLER

We use a closed-loop controller to control the motion of particles on the plate. Before starting the control experiments, we perform the learning experiments, either with the real hardware or the simulator, to collect the required learning datasets. After that, we calculate the optimal policy π^* for the manipulation task according to Section IV-A. We then transfer the acquired policy to an embedded real-time controller (National Instruments, PXIe-8135). During the control experiment, we place the manipulation specimens on the plate and capture the top-view of the plate with a camera. We repeatedly measure the position of the objects on the plate. In every step, we use the acquired policy, similar to Equation 6, to choose a note that moves the particles on the optimal trajectory. After playing the best note, the position is sampled again and the calculations are repeated for the new state.

The embedded real-time controller includes a three-layer software architecture implemented using NI LabVIEW system design software, similar to [22]. The architecture includes a user interface layer, a process layer, and a hardware layer.

The user interface layer includes a menu for setting the parameters of manipulation. It also displays the live image data from the camera, the controller state, the signal generation state, and the coordinates of the particle on a host PC.

The process layer provides the algorithms for the manipulation system by using LabVIEW 2017 Real-Time Module. This layer processes the video images, detects the particle, and makes control decisions based on the optimal policy π^*.

The hardware layer handles the low-level driver functions required to operate the apparatus including the camera interface and the configuration of the AWG. The camera interface captures the camera recording via NI Vision Acquisition Software at a frame rate of approximately 42 fps and a quality of 8-bit grayscale. Every frame is then sent to the process layer where the particle detection is performed. The AWG produces precise sine waveforms of the manipulation frequency using National Instruments PXI 5412. When the controller decides which frequency to play, the signal generation is initiated and continued until the new control command is received. If the new control command still includes the previous frequency, the AWG continues to generate the same signal without stopping. However, if the controller commands to switch the frequency, the AWG switches the signal frequency with a relatively short delay in the range of approximately 2 milliseconds.

V. RESULTS

In this section, we explain the main results of this work including the evaluation of the learning algorithm, as well as the manipulation experiments on the Chladni plate.

A. EVALUATION OF THE LEARNING ALGORITHM

To avoid excessive interaction with the real hardware, we build a simulation framework that replicates a Chladni plate similar to the experimental platform, simulating the transition probability function p (see Equation 1). To build the simulation framework, we collect experimental data from the real hardware similar to [7], and then apply particle tracking velocimatry and LOESS regression (locally weighted scatterplot smoothing) to such experimental data, again similar to [7]. Such simulation framework takes the positions of the particles and the frequency of a note as the input, and estimates the resulted positions of the particles as the output. It also adds a random vector to the resulted positions to

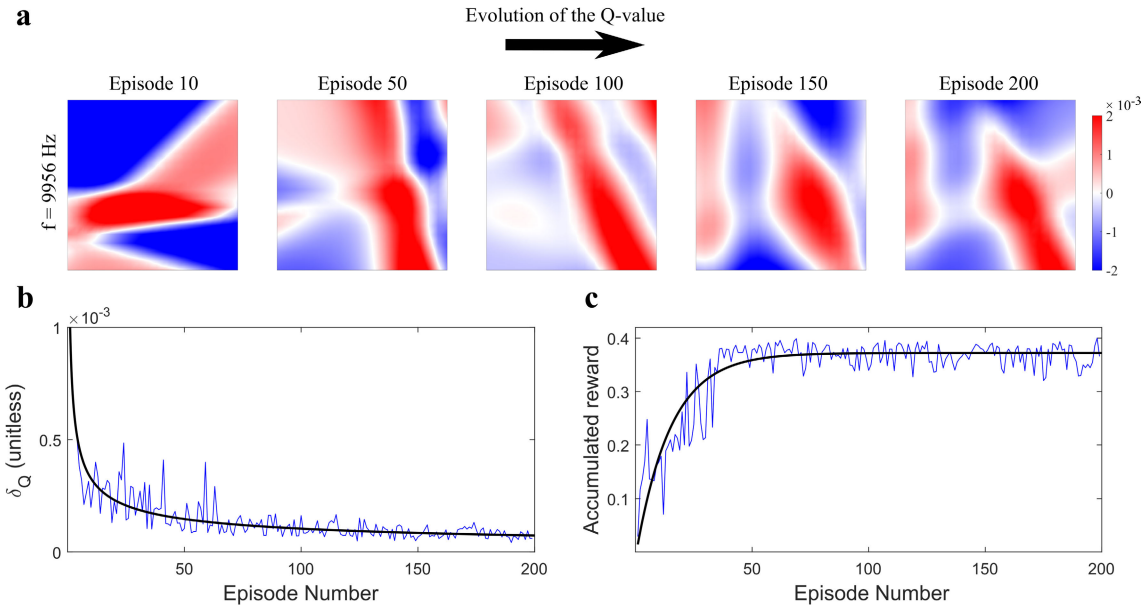

FIGURE 4. (a) Evolution of the Q-value function: the plots show the Q-value function for a specific action ($f = 9956$ Hz) after 10, 50, 100, 150, and 200 episodes of training. The function converges and stabilizes after several episodes of training. (b) δ_Q vs. episode number: δ_Q generally decreases with more episodes of training, showing the convergence of the Q-value function. (c) Learning curve: the blue line shows the accumulated reward vs. episode number during a point-to-point manipulation experiment. The black lines guide the eye in (b) and (c).

simulate the stochastic behavior of the system. The random vector is drawn from a standard normal distribution according to the uncertainty of the resulted motion by playing each note.

We first implemented and tested our learning method in the simulation framework. The simulation was carried out in a Lenovo X230 laptop with a 2.9 GHz Intel Core i7 processor, 16GB of RAM, and running Windows 7 Professional 64 bit. The learning algorithm was programmed using MATLAB R2019a as the development platform.

In the simulation experiment, we considered a Ø600 μm pressed solder ball moving from a position close to the upper left corner of the plate towards a traget close to the upper right corner. In each training episode, the starting position is randomly set in the space between the starting and target locations. We chose the following values in the learning algorithm: number of episodes $N_e = 200$, number of steps in each episode $N_t = 110$, $\epsilon = 0.8$, and $\gamma = 0.9$. For each note, we used a two-layer feed-forward neural network with a hidden layer size of 7 to capture the essential details of the action-value function Q^*.

We used the simulation experiments to acquire the learning curves of the system (Fig 4b-c). After each episode, we trained the action-value neural networks according to the learning algorithm, and stored the trained networks. Figure 4a shows the evolution of the Q-value function for an specific action ($f = 9956$ Hz) after 10, 50, 100, 150, and 200 episodes of training. As can be seen in Fig. 4a, the Q-value function converges and stabilizes after several episodes of training. Notably, the Q-value function after 150 and 200 episodes of training remains relatively unchanged. To quantify the convergence of the Q-value function, we introduce the error δ_Q which calculates the difference between the Q-value function after k episodes of training Q_k and Q_N, that is, the Q-value function at the end of the training experiments as follows,

$$\delta_Q = \frac{1}{L \cdot W} \iint_M |Q_N - Q_k| \, dx \, dy, \tag{10}$$

where x and y represent the two dimensions of the motion, L and W denote the side length and width of the plate, and M represents the manipulation space, that is, a sub-space of the state space covering a selected neighborhood of the starting and target positions. Figure 4b shows δ_Q for an specific action ($f = 9956$ Hz) after k episodes of training. As Fig. 4b shows, δ_Q generally decreases with more episodes of training, demonstrating the convergence of the Q-value function.

We then performed a control experiment using the acquired policy after each episode, and recorded the accumulated reward after 60 control steps. Figure 4c shows the accumulated reward that the system gains during point-to-point control experiments after each training episode. It shows that the performance of the controller improves with the number of episodes. In this particular example, the agent learns the optimal policy after approximately 50 episodes of training, which is equivalent to approximately 46 minutes of training experiments with the real hardware. Using the methods explained in [7], [18], similar manipulation task requires almost 390,000 data points for training, while the model-free controller needs just 5,500 training data points, which is a significant reduction in the required effort for data collection.

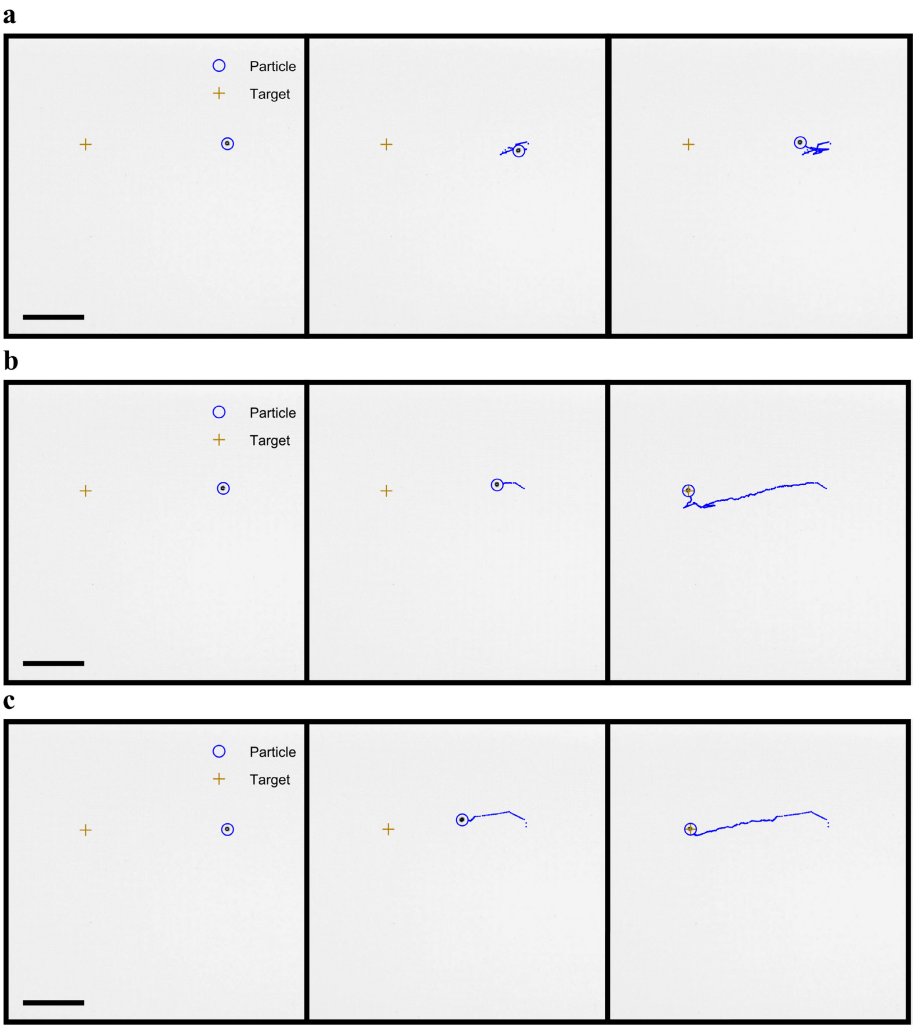

FIGURE 5. Manipulation of a 600 μm pressed solder on the plate: (a) The manipulation task is unsuccessful after 10 episodes of training. (b) Manipulation is successful after 50 episodes of training. (c) After 200 episodes of training, the controller could successfully perform the task more than two times faster than experiment (b). Scale bar, 10 mm.

B. MANIPULATION EXPERIMENTS

We have successfully demonstrated motion control of a single particle using the closed-loop controller, as shown in Fig. 5 and Video SV1. We first used the simulation framework to learn the optimal policy. We then transferred the acquired policy to the embedded controller. We performed the experiments for the policies after 10, 50, and 200 training episodes. During the manipulation experiments, once a frame is captured by the camera, the current position of the particle is measured. The measured position is then fed to the controller. The controller calculates the optimal action according to Equation 6, and selects the note that directs the particle towards the current waypoint. The selected note is played on the vibrating plate until the next image frame is received by the controller. The controller continues exciting the plate until the distance between the particle and its target is less than a predefined threshold.

According to our experiments, the control was unsuccessful after 10 episodes of training (Fig. 5a). Nevertheless, the performance of the controller enhanced after 50 episodes of training and the controller successfully performed the manipulation task in 44 seconds (Fig. 5b). After 200 episodes of training, the controller could successfully perform the task in only 18 seconds (Fig. 5c).

The system is also capable of learning to manipulate multiple particles simultaneously. We have successfully demonstrated motion control of two (Fig. 6a and Video SV2) and three (Fig. 6b and Video SV3) particles on the plate. We again used the simulation framework to learn the optimal policy. We performed successful manipulation experiments for the policies after 2,000 and 50,000 training episodes for two and three particles, respectively.

It should be noted that as we control the position on a two-dimensional plate surface with x and y directions, the number of states for n_p particles becomes $2n_p$, where n_p represents the number of particles. Consequently, the state-space in single-particle manipulation has two dimensions, two-particle manipulation four dimensions, and three-particle manipulation six dimensions. In reinforcement learning, the required number of learning episodes increases with the

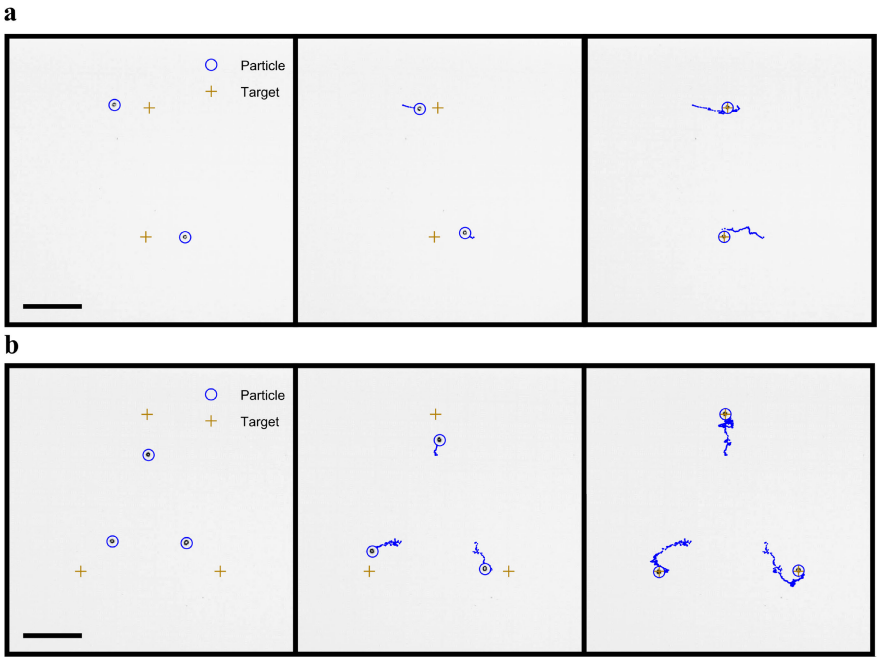

FIGURE 6. Simultaneous manipulation of (a) two and (b) three 600 μm pressed solder balls on the plate. Scale bar, 10 mm.

number of states in a positive power-law relation. As the states in our problem are x and y components of position, the required episodes for learning the task increases almost exponentially with the number of particles.

We emphasize that we used the same algorithm, Algorithm 1, also for multi-particle manipulation, which is still fundamentally a single-agent reinforcement learning method. Here, the state of the agent is a list of the particle positions. Therefore, it should not be confused with the multi-agent reinforcement learning methods.

VI. CONCLUSION

In this paper, we provide a novel control method for model-free acoustic manipulation. Here, we have exploited recent advances in reinforcement learning, which have been applied to macroscale robots, for manipulating sub-mm objects using acoustics. The control method is model-free in the sense that it does not require a dynamic model of the particle motion in the acoustic field. The controller learns the optimal control policy by exciting various acoustic fields and collecting observations of the resulted particle motion.

As a demonstration, we have applied the control method to manipulate particles on the surface of a Chladni plate. The results are quite promising in terms of controller stability and convergence. We have observed that the performance of the controller enhances by interacting with the system. We have successfully demonstrated manipulation of a single and multiple metallic particles on the plate. Our method is also applicable to other types of particles with different shapes and materials, similar to the ones reported in [7], [18].

Our method can also be applied to several other dynamic-field acoustic manipulation methods. Fundamentally, particle manipulation in many dynamic-field acoustic devices is similar to a Chladni plate. Typically, there is one or several acoustic sources that can generate diverse acoustic fields to move particles, similar to our device. Thus our control method is generally realizable for those systems. To give a few examples, our method is hypothetically applicable to surface acoustic wave (SAW) devices [5], acoustic levitators [17], [23], and in-fluid acoustic devices [8], [9], [11], [24]–[26], where accurate theoretical modelling of the acoustic field involves major challenges and difficulties.

Even though model-free control and reinforcement learning have been broadly applied to many macroscale robotic applications [27], those methods have been barely applied to microrobotics except for a few contact micromanipulation systems [28]–[30]. Robot learning algorithms have never been applied to field-based micromanipulation systems before, and this work is the first attempt that combines the two realms of robot learning and field-based micromanipulation. Therefore, we believe our work will benefit other forms of field-based manipulation systems regardless of the actuation type, such as magnetic and electrostatic, by introducing the applicability of robot learning algorithms. In particular, our control method would be broadly applicable to those systems, where the untethered robots, with no on-board sensing and computing, are either directly manipulated or manipulating other objects indirectly.

Future work would involve developing autonomous shaping and structuring methods which are highly programmable by employing nonlinear acoustic fields. Such methods would require solving complex sequential decision making problems. Recently, deep reinforcement learning methods have shown great advances in solving rather complex decision making problems [31], [32]. We believe such methods and the point-of-view of this paper can potentially build

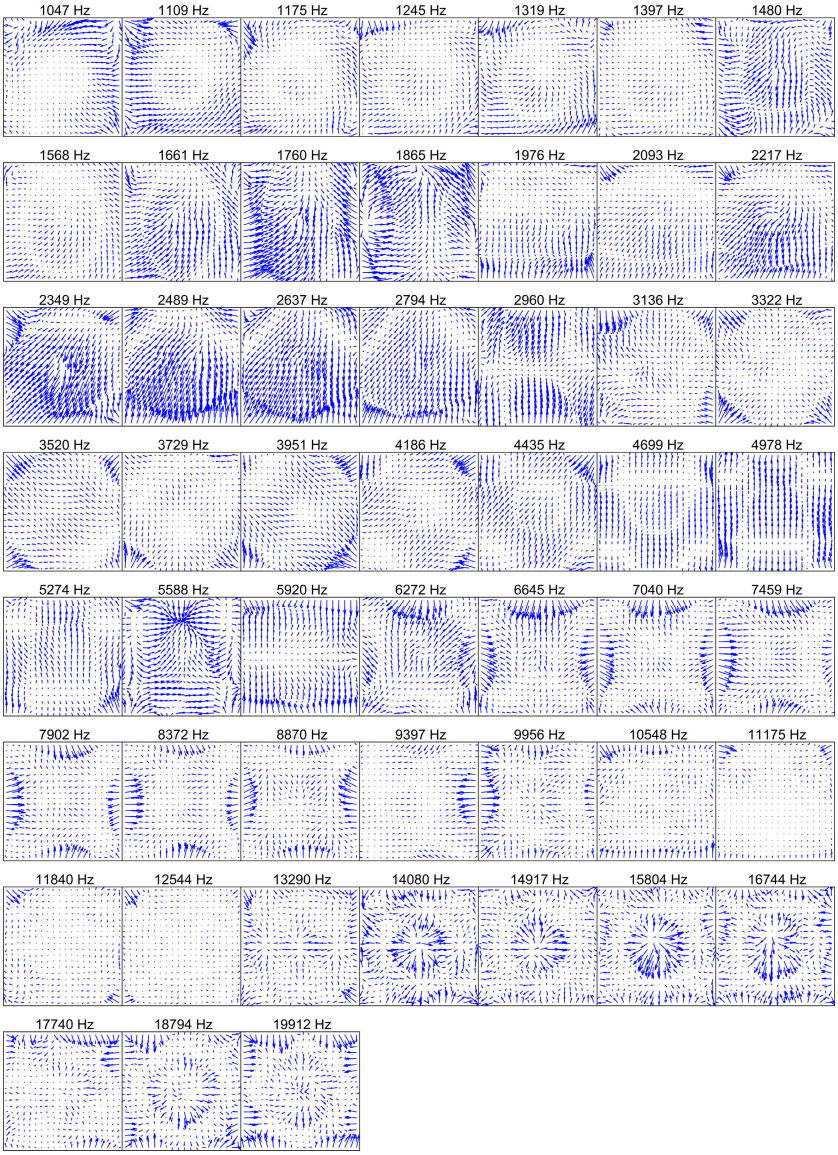

FIGURE 7. **Frequency-dependent displacement fields on a Chladni plate.**

the foundation of a new paradigm for autonomous matter forming.

APPENDIX

We emphasize that optimal control of particle motion in dynamic-filed acoustic devices is a challenging decision making problem. The reason is that the generated acoustic fields are highly diverse; thus to operate in an optimal way, the controller should carefully select and generate the best possible acoustic field. For instance, Figure 7 shows the diverse displacement fields that a Chladni plate can produce at the frequencies selected from Chromatic musical scale in the range of 1-20 kHz.

ACKNOWLEDGMENT

The authors acknowledge Micronova Nanofabrication Centre for providing laboratory facilities for microfabrication. The authors would also like to thank M. Hazara, D. Harischandra, and H. Wijaya for the fruitful discussions during the course of this work.

REFERENCES

[1] M. Krishnan, N. Mojarad, P. Kukura, and V. Sandoghdar, "Geometry-induced electrostatic trapping of nanometric objects in a fluid," *Nature*, vol. 467, no. 7316, pp. 692–695, Oct. 2010.

[2] S. Tasoglu, E. Diller, S. Guven, M. Sitti, and U. Demirci, "Untethered micro-robotic coding of three-dimensional material composition," *Nat. Commun.*, vol. 5, no. 3124, pp. 1–9, 2014.

[3] D. G. Grier, "A revolution in optical manipulation," *Nature*, vol. 424, no. 6950, pp. 810–816, Aug. 2003.

[4] D. Foresti, M. Nabavi, M. Klingauf, A. Ferrari, and D. Poulikakos, "Acoustophoretic contactless transport and handling of matter in air," *Proc. Nat. Acad. Sci. USA*, vol. 110, no. 31, pp. 12549–12554, Jul. 2013.

[5] X. Ding, S.-C.-S. Lin, B. Kiraly, H. Yue, S. Li, I.-K. Chiang, J. Shi, S. J. Benkovic, and T. J. Huang, "On-chip manipulation of single microparticles, cells, and organisms using surface acoustic waves," *Proc. Nat. Acad. Sci. USA*, vol. 109, no. 28, pp. 11105–11109, Jul. 2012.

[6] T. Laurell, F. Petersson, and A. Nilsson, "Chip integrated strategies for acoustic separation and manipulation of cells and particles," *Chem. Soc. Rev.*, vol. 36, no. 3, pp. 492–506, Dec. 2006.
[7] Q. Zhou, V. Sariola, K. Latifi, and V. Liimatainen, "Controlling the motion of multiple objects on a Chladni plate," *Nat. Commun.*, vol. 7, no. 12764, pp. 1–10, 2016.
[8] K. Latifi, H. Wijaya, and Q. Zhou, "Motion of heavy particles on a submerged Chladni plate," *Phys. Rev. Lett.*, vol. 122, no. 18, 2019, Art. no. 184301.
[9] G. Vuillermet, P. Y. Gires, F. Casset, and C. Poulain, "Chladni patterns in a liquid at microscale," *Phys. Rev. Lett.*, vol. 116, no. 18, 2016, Art. no. 184501.
[10] J. Goldowsky, M. Mastrangeli, L. Jacot-Descombes, M. R. Gullo, G. Mermoud, J. Brugger, A. Martinoli, B. J. Nelson, and H. F. Knapp, "Acousto-fluidic system assisting in-liquid self-assembly of micro-components," *J. Micromech. Microeng.*, vol. 23, no. 12, Dec. 2013, Art. no. 125026.
[11] B. W. Drinkwater, "Dynamic-field devices for the ultrasonic manipulation of microparticles," *Lab Chip*, vol. 16, no. 13, pp. 2360–2375, May 2016.
[12] L. Y. Yeo and J. R. Friend, "Ultrafast microfluidics using surface acoustic waves," *Biomicrofluidics*, vol. 3, no. 1, Mar. 2009, Art. no. 012002.
[13] M. Wiklund, "Acoustofluidics 12: Biocompatibility and cell viability in microfluidic acoustic resonators," *Lab Chip*, vol. 12, no. 11, p. 2018, 2012.
[14] J. Shi, X. Mao, D. Ahmed, A. Colletti, and T. J. Huang, "Focusing microparticles in a microfluidic channel with standing surface acoustic waves (SSAW)," *Lab Chip*, vol. 8, no. 2, pp. 221–223, Dec. 2007.
[15] Z. Wang and J. Zhe, "Recent advances in particle and droplet manipulation for lab-on-a-chip devices based on surface acoustic waves," *Lab Chip*, vol. 11, no. 7, p. 1280, 2011.
[16] A. Marzo, S. A. Seah, B. W. Drinkwater, D. R. Sahoo, B. Long, and S. Subramanian, "Holographic acoustic elements for manipulation of levitated objects," *Nat. Commun.*, vol. 6, no. 8661, pp. 1–7, 2015.
[17] A. Marzo and B. W. Drinkwater, "Holographic acoustic tweezers," *Proc. Nat. Acad. Sci. USA*, vol. 116, no. 1, pp. 84–89, Jan. 2019.
[18] K. Latifi, H. Wijaya, and Q. Zhou, "Multi-particle acoustic manipulation on a Chladni plate," in *Proc. Int. Conf. Manipulation, Autom. Robot. Small Scales (MARSS)*, Montreal, QC, Canada, Jul. 2017, pp. 1–7.
[19] M. Riedmiller, "Neural fitted Q iteration–first experiences with a data efficient neural reinforcement learning method," in *Proc. Eur. Conf. Mach. Learn. (ECML)*, 2005, pp. 317–328.
[20] M. J. Gander and F. Kwok, "Chladni figures and the Tacoma bridge: Motivating PDE eigenvalue problems via vibrating plates," *SIAM Rev.*, vol. 54, no. 3, pp. 573–596, Jan. 2012.
[21] P. H. Tuan, C. P. Wen, P. Y. Chiang, Y. T. Yu, H. C. Liang, K. F. Huang, and Y. F. Chen, "Exploring the resonant vibration of thin plates: Reconstruction of Chladni patterns and determination of resonant wave numbers," *J. Acoust. Soc. Amer.*, vol. 137, no. 4, pp. 2113–2123, Apr. 2015.
[22] K. Latifi, A. Kopitca, and Q. Zhou, "Rapid mode-switching for acoustic manipulation," in *Proc. Int. Conf. Manipulation, Autom. Robot. Small Scales (MARSS)*, Helsinki, Finland, Jul. 2019, pp. 1–6.
[23] H. Wijaya, K. Latifi, and Q. Zhou, "Two-dimensional manipulation in mid-air using a single transducer acoustic levitator," *Micromachines*, vol. 10, no. 4, p. 257, Apr. 2019.
[24] N. R. Skov, "Modeling of complex acoustofluidic devices," Ph.D. dissertation, Dept. Phys., Technical Univ. Denmark, Lyngby, Denmark, 2019.
[25] T. Lilliehorn, U. Simu, M. Nilsson, M. Almqvist, T. Stepinski, T. Laurell, J. Nilsson, and S. Johansson, "Trapping of microparticles in the near field of an ultrasonic transducer," *Ultrasonics*, vol. 43, no. 5, pp. 293–303, Mar. 2005.
[26] M. Ohlin, A. E. Christakou, T. Frisk, B. Önfelt, and M. Wiklund, "Influence of acoustic streaming on ultrasonic particle manipulation in a 100-well ring-transducer microplate," *J. Micromech. Microeng.*, vol. 23, no. 3, Mar. 2013, Art. no. 035008.
[27] J. Kober, J. A. Bagnell, and J. Peters, "Reinforcement learning in robotics: A survey," *Int. J. Robot. Res.*, vol. 32, no. 11, pp. 1238–1274, 2013.
[28] J. Li, Z. Li, and J. Chen, "Microassembly path planning using reinforcement learning for improving positioning accuracy of a 1 cm^3 omnidirectional mobile microrobot," *Appl. Intell.*, vol. 34, no. 2, pp. 211–225, Apr. 2011.
[29] C. Adda, G. Laurent, and N. Le Fort-Piat, "Learning to control a real micropositioning system in the STM-Q framework," in *Proc. IEEE Int. Conf. Robot. Autom.*, Barcelona, Spain, Apr. 2005, pp. 4569–4574.
[30] G. Laurent, "On-line learning for micro-object manipulation," in *Markov Decision Processes in Artificial Intelligence*, O. Sigaud and O. Buffetpp, Eds. Hoboken, NJ, USA: Wiley, 2010, pp. 361–374.
[31] D. Silver, J. Schrittwieser, K. Simonyan, I. Antonoglou, A. Huang, A. Guez, T. Hubert, L. Baker, M. Lai, A. Bolton, Y. Chen, T. Lillicrap, F. Hui, L. Sifre, G. Van Den Driessche, T. Graepel, and D. Hassabis, "Mastering the game of Go without human knowledge," *Nature*, vol. 550, no. 7676, pp. 354–359, Oct. 2017.
[32] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski, S. Petersen, C. Beattie, A. Sadik, I. Antonoglou, H. King, D. Kumaran, D. Wierstra, S. Legg, and D. Hassabis, "Human-level control through deep reinforcement learning," *Nature*, vol. 518, no. 7540, pp. 529–533, Feb. 2015.

KOUROSH LATIFI was born in Tehran, Iran, in 1984. He received the B.Sc. degree in mechanical engineering from the Amirkabir University of Technology (AUT), Tehran, in 2006, the M.Sc. degree in mechatronics from the Iran University of Science and Technology (IUST), Tehran, in 2009, and the second M.Sc. degree in mechatronics and micromachines from the Tampere University of Technology, Tampere, Finland, in 2014. He is currently pursuing the Ph.D. degree with the Robotic Instruments Research Group, Aalto University, Espoo, Finland, where he has been developing tools and techniques for moving small objects using sound.

His work has been published in major international journals, including *Nature Communications* and *Physical Review Letters*. His current research interests include intelligent micro manipulation, acoustic manipulation, machine learning, and automation.

ARTUR KOPITCA received the B.Sc. degree in automation and electrical engineering, in 2018. He is currently pursuing the M.Sc. degree in control, robotics and autonomous systems with Aalto University, Espoo, Finland.

He is also a Research Assistant with the Robotic Instruments Research Group, School of Electrical Engineering, Aalto University, Finland. His research interests include acoustic manipulation, real-time control systems, and intelligent robotics.

QUAN ZHOU (Member, IEEE) received the M.Sc. and Dr.Tech. degrees from the Tampere University of Technology, Finland.

He was a Professor with Northwest Polytechnical University, Xi'an, China. He is currently an Associate Professor with the Robotic Instruments Group, School of Electrical Engineering, Aalto University, Finland. His work has been published in major international journals, including *Nature Communications*, *Physical Review Letters*, *Advanced Materials*, *Small*, and the IEEE TRANSACTIONS ON ROBOTICS. His current research interests include micro- and nano-manipulation and zautomation methods.

Prof. Zhou received the Anton Paar Research Award for Instrumental Analytics and Characterization. He was also the General Chair of the International Conference on Manipulation, Automation and Robotics at Small Scales, MARSS 2019. He was also the Chair of the IEEE Finland Joint Chapter of Control System Society, Robotics and Automation Society and System Man and Cybernetics Society. He was a Coordinator of the EU FP7 Project FAB2ASM, the First PPP Project of the European Economic Recovery Plan.