Ultrasound Rendering of Tactile Interaction with Fluids

Héctor Barreiro, Stephen Sinclair and Miguel A. Otaduy
Universidad Rey Juan Carlos, Madrid, Spain

***Abstract*— When we interact with fluid media, e.g., with our hands, we experience a spatially and temporally varying pressure field on our skin, which depends on the density and viscosity of the fluid, as well as the relative motion between our hands and the surrounding flow. Ultrasound phased arrays stimulate skin in mid air by controlling pressure waves at particular spatial locations. In this work, we explore the connection between the pressure-based stimulation of ultrasound haptics and the actual pressure field experienced when interacting with fluid media, to devise a novel algorithm for ultrasound-based rendering of tactile interaction with fluids. Our algorithm extracts the target pressure field on a virtual hand from an interactive fluid simulation, and formulates the computation of the rendered pressure as an optimization problem. We have designed an efficient solver for this optimization problem, and we show results of interactive experiments with several fluid simulations.**

I. Introduction

Ultrasound haptics enjoys the ability to produce direct touch sensations on the skin in mid air, i.e., without the need to hold or wear a haptic device [12], [19]. It employs an array of ultrasound transducers as actuators, which produce high-frequency pressure waves in the space around the device. By modulating the activation of the transducers, it is possible to aggregate the pressure waves at specific points in space, and thus create focal pressure points. Pressure reaches perceivable values at such focal points, and produces a touch sensation in mid air. By eliminating the need to hold or wear a haptic device, ultrasound haptics promises the possibility of a more immersive and scalable virtual touch experience. Ultrasound haptics also suffers evident limitations, however. The forces they exert on skin are subtle and cannot impose constraints on the user's motion.

Fluids appear as an interesting phenomenon to be rendered using ultrasound haptics, as they can be moved around without constraining the user's motion. When we interact with fluids, e.g., with our hands, we experience a temporally and spatially varying pressure field on our skin. This pressure field is the combined result of our own motion, the inherent properties of the fluid (i.e., density and viscosity), and the dynamic state of the fluid (i.e., velocity).

Rendering the tactile interaction with fluids using ultrasound haptics can be formulated as a problem of dynamically reproducing the pressure field on the user's skin. This problem comes with two major challenges. One is to perform a real-time fluid simulation with moving objects (i.e., the user's hands) and dynamically extract the pressure field on the skin. The other one is to dynamically optimize the actuation of the transducers to approximate the pressure field on the skin.

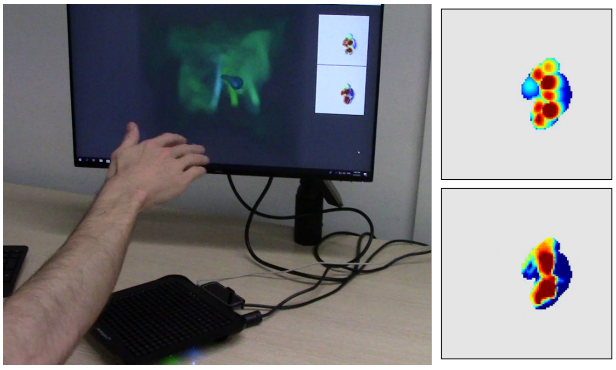

Fig. 1: On the left image, a user interacts with a fluid simulation. His hand is tracked and mapped to a virtual hand that stirs the simulated fluid. We propose a novel tactile rendering algorithm that extracts the pressure field on the virtual hand (bottom right), and optimizes a pressure field (top right) that is rendered to the user with the ultrasound phased array shown in the left image.

In this work, we propose an algorithm for ultrasound rendering of tactile interaction with fluids. We characterize the actuation of ultrasound haptics using a set of focal points, and we optimize the location and intensity of such focal points to best approximate the pressure field on the skin. We devise efficient methods to extract the skin pressure field from the fluid simulation and optimize the focal points at high update rates, and hence produce a responsive experience while dynamically interacting with a virtual fluid.

We have implemented our ultrasound tactile rendering method on an Ultrahaptics STRATOS device. We demonstrate example interactions where the user interacts with a fluid container of $100 \times 100 \times 100$ cells, with the fluid simulation running at 90Hz and tactile rendering at 30Hz.

II. Related Work

A. Ultrasound Haptics

Ultrasound phased arrays achieve mid-air stimulation through a phenomenon known as *acoustic radiation pressure*. Multiple ultrasound transducers, producing an ultrasound wave of the same frequency, are modulated in phase to achieve maximal combined pressure intensity at a certain location in space, known as the focal point. The focus of the ultrasound wave is determined by its wavelength (e.g., 8.5mm for 40kHz ultrasound). Iwamoto et al. [14] were the first to exploit this principle for mid-air haptic stimulation, limited to a fixed focal point. Hoshi et al. [12] leveraged

the approach to produce moving focal points by temporally varying the modulation of the transducers.

The mid-air interaction capabilities of ultrasound haptics have enabled some unprecedented applications. One of them is the interaction with floating images produced with 3D displays [20], giving to the user the illusion of seeing and touching a virtual object in a fully co-located and natural manner. By tiling ultrasound phased arrays in 3D, the approach has been extended to enable tactile interaction with holograms [13].

The pressure wave at a focal point can be expressed as a complex-valued linear function of the amplitudes and phases of the waves emitted by the transducers. Given target pressure waves at a set of focal points, it is possible to obtain the optimal emitted waves by solving a quadratic optimization. However, typically only the target amplitudes are given, which makes the problem nonlinear. Inoue et al. [13] compare solutions to this problem from the field of scattering diffraction imaging. Long et al. [19] found an efficient solution that instead uses the original linear function directly, finding first optimal phases at the control points by solving an eigenproblem. They refer to their method for controlling the pressure intensity at focal points as *amplitude modulation*.

Korres and Eid [16] proposed a new control method for ultrasound phased arrays. Instead of matching a target pressure at a set of focal points, they moved a single focal point at high speed to produce shapes in 3D. This approach has also been applied by Kappus and Long [15], who call it *spatiotemporal modulation*. In this control strategy, given a target shape, there is a dependency between the refresh rate and the velocity of the focal point. These parameters must be tuned for optimal sensitivity.

Neither amplitude modulation nor spatiotemporal modulation alone satisfy the needs of our rendering problem. They cannot control a spatially varying pressure field. As discussed later in Section III-A, we base our rendering algorithm on amplitude modulation, but we optimize the location and intensity of focal points in a dynamic manner to match the interaction occurring in a fluid simulation.

Several works have analyzed perceptual aspects of ultrasound haptic stimulation, as well as performance implications of algorithmic parameters. When interacting with skin tissue, the pressure exerted by the acoustic waves induces shear deformation of the skin, which triggers mechanoreceptors. To maximize tactile sensation, the ultrasound waves are modulated at a frequency corresponding to the peak sensitivity of mechanoreceptors (200Hz to 300Hz). Carter et al. [2] studied how to position secondary focal points of low pressure to eliminate spurious pressure maxima within the device workspace. They also analyzed the ability of subjects to discriminate focal points. Wilson et al. [26] studied the ability of subjects to locate focal points and the perception of motion, and they found, for example, an average localization error of 8.5mm. Hasegawa and Shinoda [11] have analyzed the pressure fields actually produced by ultrasound phased arrays, as well as perceptual detection thresholds.

Recent studies have analyzed the effects of rendering parameters of spatiotemporal modulation. Frier et al. [8] found that there is some optimal focal point speed (between 5 and 8 m/s in their experiments) to maximize skin sensitivity. Recently, they have analyzed the combined influence of spatial and temporal sampling in spatiotemporal modulation [9].

B. Other Mid-Air Actuation Methods

Ultrasound is not the only stimulation technology for mid-air interaction. Other possibilities are to emit air vortex rings [23], [10], or to induce thermoelastic effects using a laser [17]. Some authors have also researched the use of electromagnetic fields, although this approach requires the user to wear magnetic disks [28].

Beyond mid-air actuation methods, other actuation technologies also offer the possibility to stimulate skin directly, mimicking direct touch. A recent survey is provided in [22].

C. Haptic Rendering of Fluid Media

Haptic rendering of tool-based interaction with fluids has received large attention. The various existing methods address the challenge of running interactive fluid simulations, and they propose different coupling strategies between the haptic device and the simulated tool. Baxter and Lin [1] developed a 2D fluid simulation with haptic interaction, where they applied to the device the forces aggregated on the boundary of the simulated tool. Dobashi et al. [6] developed a method that combines a real-time computation of linear force terms with precomputed nonlinear force terms. Mora and Lee [21] computed a real-time 3D fluid simulation, and mapped viscous forces to the haptic device.

The advent of GPUs as computational platforms has allowed richer interactive methods. Yang et al. [27] leveraged this technology, and implemented methods to accumulate grid forces on the simulated tool directly on the GPU. Cirio et al. [4] performed a particle-based simulation of the fluid on the GPU, and used a virtual coupling method to transfer fluid forces on the simulated tool to the haptic device.

As far as we know, all previous work on haptic interaction with fluid media is limited to tool-based interaction, and no work has addressed direct-touch interaction, let alone mid-air interaction through ultrasound haptics. One tangentially related work is the use of vibrotactile feedback to render the interaction with splashing fluids, conveying the vibrations produced by air bubbles [3].

III. Rendering Based on Pressure Field Optimization

Without loss of generality, we assume that the user interacts with the simulated fluid using one hand. Then, the hand of the user is tracked in real-time, and a virtual replica of the hand is moved within the fluid simulation. The motion of the virtual hand, together with the properties and the motion of the fluid, produce a temporally and spatially varying pressure field on its surface. Ultrasound phased arrays are capable of producing a spatial pressure field. Therefore, we choose to render the tactile interaction between the user and the

simulated fluid by matching the pressure field produced by the ultrasound device to the pressure field on the user's virtual hand.

We start this section by characterizing the pressure field produced by the device, according to the control method of choice. Then, we define the target pressure field, which is constrained to the surface of the hand visible from the device. Finally, we formulate an optimization problem to compute the rendering output, and we provide an efficient solution algorithm.

A. Command and Stimuli of Ultrasound Haptic Devices

As discussed in the previous section, ultrasound phased arrays can be controlled in two ways. In amplitude modulation mode, the ultrasound device is controlled by setting the amplitude of the pressure wave at a small number of focal points. In spatiotemporal modulation mode, the ultrasound device is controlled by indicating a region in space where the pressure wave is active. In both modes, an internal optimization computes the required activation patterns of the ultrasound transducers.

In our rendering problem, we wish to modulate a pressure field in space. One approach would be to extend emitter phase modulation to support arbitrary target regions with spatiotemporally varying pressure; another could be to modulate high-velocity spatiotemporal control points to maximally cover the target field. However, these approaches would require the solution to a complex optimization of high dimensionality (i.e., the activation pattern of each transducer, or long-term point trajectories), running at high update rates.

Instead, we opt for an approach based on amplitude modulation, which allows direct control of pressure values, albeit at a small number of points. We leverage the observation that, in reality, pressure is not concentrated at focal points, but exhibits a smooth fall-off determined by the wavelength of the ultrasound signal (e.g., 8.6mm for the 40kHz of our test device). This fall-off can be well approximated by a Gaussian function ϕ [12]. Then, given a focal point at position $\mathbf{x}_i$ with nominal pressure p_i, and a pressure fall-off with standard deviation σ, the pressure at position $\mathbf{x}$ can be characterized as

$$p(\mathbf{x}) = p_i \, \phi(\|\mathbf{x}-\mathbf{x}_i\|) = p_i \, \mathrm{e}^{-\frac{\|\mathbf{x}-\mathbf{x}_i\|^2}{2\sigma^2}}. \tag{1}$$

If focal points are distant enough, we can safely assume that the pressure at every location depends only on the closest control point. Then, given a set of focal points $\{\mathbf{x}_i\}$, each one of them defines the pressure field over its Voronoi region R_i according to (1).

B. Target Pressure Field

Given a fluid simulation defined on a volume domain $\mathbf{D}$, we are interested only in the pressure field on the surface of the user's hand. Moreover, parts of the surface of the hand may be occluded from the ultrasound device, hence it is pointless to try to match their simulated pressure. Consequently, we define the target pressure field $p^*(\mathbf{x})$ on a target domain $\mathbf{R} = \{\mathbf{x} \in \mathbb{R}^3\}$ formed by the portion of the surface of the virtual hand that is visible from the ultrasound device. Since the ultrasound phased array is capable of producing pressure intensities, we clamp to zero the negative target pressure values that occur when the hand moves away from the flow.

In practice, we sample the target domain $\mathbf{R}$ with a set of points. In Section IV-B we describe an efficient GPU-based algorithm to extract the target domain $\mathbf{R}$ and the target pressure field $p^*(\mathbf{x})$ from a fluid simulation.

C. Pressure Field Optimization

Based on all the ingredients described thus far, we define the rendering problem as the search of N focal points and their pressure magnitudes, such that the difference between rendered and target pressures, $p(\mathbf{x}) - p^*(\mathbf{x})$, is minimized over the target domain $\mathbf{R}$. Following the assumption that the focal points are distant and hence the rendered pressure at each point is defined only by the closest focal point, we partition the target domain into the Voronoi regions of the focal points, i.e., $\mathbf{R} = \bigcup_i R_i$. Within each Voronoi region, the summed pressure difference depends only on the pressure magnitude of the corresponding focal point.

We propose an approximate solution to the pressure optimization problem that works in two steps. First, we compute the positions of focal points following a clustering approach. Second, we estimate the pressure magnitude of each focal point to best match the target pressure within its Voronoi region.

1) Optimization of Focal Points: We formulate a clustering problem using target pressure values as weights. Formally, this amounts to minimizing the following objective function:

$$f(\{\mathbf{x}_i\}) = \sum_i \sum_{\mathbf{x} \in R_i} p^*(\mathbf{x}) \, \|\mathbf{x}-\mathbf{x}_i\|^2. \tag{2}$$

This objective function corresponds to a weighted k-means clustering problem, which can be solved efficiently using Lloyd's algorithm [18]. The algorithm iterates steps where it computes the weighted centroids of Voronoi regions and then updates those Voronoi regions, until convergence.

At every render frame, we initialize the iterative algorithm by placing the N focal points at distant high-pressure locations. In practice, we search for the N points in $\mathbf{R}$ with highest target pressure, such that they are separated by a distance larger than σ.

2) Optimization of Pressure Magnitudes: Once the focal points and hence the Voronoi partition are fixed, we optimize the pressure magnitude for each focal point independently. For each Voronoi region, we formulate an objective function based on the summed pressure difference between the target pressure $p^*(x)$ and the actual pressure rendered by the device, accounting for its fall-off as described in (1). This is a simple quadratic function of the form

$$f(p_i) = \sum_{\mathbf{x} \in R_i} \left(p_i \, \phi(\|\mathbf{x}-\mathbf{x}_i\|) - p^*(\mathbf{x})\right)^2. \tag{3}$$

By the optimality condition $\nabla f = 0$, we compute the pressure magnitude of each focal point as

$$p_i = \frac{\sum_{\mathbf{x}\in R_i} p^*(\mathbf{x})\,\phi(\|\mathbf{x}-\mathbf{x}_i\|)}{\sum_{\mathbf{x}\in R_i} \phi(\|\mathbf{x}-\mathbf{x}_i\|)^2}. \tag{4}$$

Once focal points and their pressure magnitudes are computed, we output them to the driver of the ultrasound device. The driver then executes internally the optimization of transducer waves [19].

IV. Fluid Simulation and Rendering Pipeline

In this section, we describe our fluid simulation solver, as well as the method for extracting the pressure field on the hand's surface.

A. Fluid Simulation

We apply our algorithm to the rendering of gaseous media such as smoke. As in the work of Fedkiw et al. [7], we assume that the simulated fluid is inviscid and incompressible. Hence, its motion is described by the incompressible Euler equations:

$$\begin{aligned} \nabla\cdot\mathbf{u} &= 0, \\ \frac{\partial\mathbf{u}}{\partial t} &= -(\mathbf{u}\cdot\nabla)\,\mathbf{u} - \frac{1}{\rho}\nabla p + \mathbf{f}, \end{aligned} \tag{5}$$

where $\mathbf{u}$ corresponds to the velocity of the fluid, p is the pressure, ρ is the (constant) density, and $\mathbf{f}$ accounts for external body forces such as gravity. We solve these equations following the standard advection-projection scheme used in computer graphics [24], [7], [5], [27].

First, we compute an intermediate velocity field $\mathbf{u}^*$ by solving the self-advection equation in (5) using a semi-Lagrangian advection scheme [24] and integrating the external body forces $\mathbf{f}$ with explicit Euler. Next, to ensure mass conservation, the velocity field is projected onto a divergence-free state by solving the pressure Poisson equation

$$\nabla^2\cdot p = \frac{\rho}{\Delta t}\nabla\cdot\mathbf{u}^*, \tag{6}$$

with Neumann boundary conditions $\left(\frac{\partial p}{\partial \mathbf{n}} = 0\right)$ at boundaries with normal $\mathbf{n}$. Prior to the projection to the divergence-free state, we explicitly enforce free-slip boundary conditions ($\mathbf{u}\cdot\mathbf{n} = \mathbf{u}^*\cdot\mathbf{n}$), i.e., we set the normal velocities at fluid boundaries equal to those of the obstacles $\mathbf{u}^*$.

Following the state of the art, we discretize the simulation domain $\mathbf{D}$ using a staggered grid, with fluid pressure defined at cell centers, and fluid velocities defined component-wise at the centers of cell faces. We achieve interactive simulation times by implementing the entirety of the fluid solver on the GPU. As in the work of Crane et al. [5], we enable massive parallelism by solving the pressure Poisson equation using Jacobi relaxation.

Due to semi-Lagrangian advection, numerical dissipation might dampen interesting features of the flow, such as vortices and eddies, which could be perceptually relevant. We use vorticity confinement [25], [7] to inject additional kinetic energy at existing vortices, and thus alleviate the effects of numerical dissipation.

For visualization purposes, we define immersed media, such as smoke, through secondary density fields. We advect these density fields at every frame using the same semi-Lagrangian advection method.

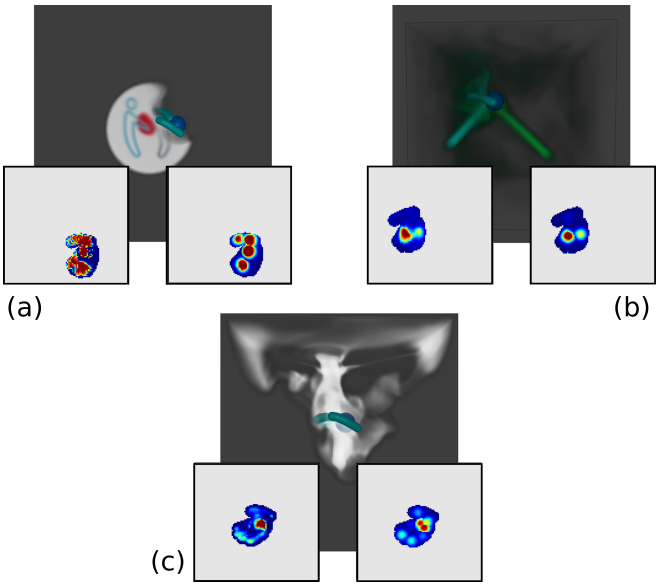

Fig. 2: Examples of fluid interaction showing target pressure values extracted from the device's view of the hand from below (lower left inset) compared with reconstruction (lower right inset). Examples are (a) stirring the fluid, (b) hand in a smoke jet, and (c) creating smoke plumes.

B. Adding the Hand and Extracting the Target Pressure Field

In the fluid simulation, the user's hand is treated as a moving obstacle with known velocity. On every simulation step, we rasterize a signed-distance representation of the hand, along with its corresponding velocity field. This representation enables an efficient classification of interior/exterior points of the domain, as well as the extraction of the surface normal, all under the same compact storage.

As outlined in Section III-B, we wish to extract the fluid pressure on the portion of the boundary of the hand $\mathbf{R}$ visible from the ultrasound device. To do this, we assume that the ultrasound device is placed on one of the walls of the domain $\mathbf{D}$. Then, for every cell of this wall, we march inward into the domain until we hit the first cell that is within one unit of the hand, and we store this cell's position $\mathbf{x}$ and pressure $p^*(\mathbf{x})$.

V. Results

A. Summary of the Complete Rendering Pipeline

In our runtime pipeline, we set different refresh rates for the haptic rendering loop and for the fluid simulation loop. We set the haptic refresh rate at 30Hz for smooth rendering. At this rate, we sample the hand tracker, update the location of the hand in the fluid simulation, extract the target pressure

(a) With emitter

	$\sigma = 8$ mm	16 mm	24 mm	32 mm
$N = 2$	0.357	0.283	0.220	0.197
$N = 4$	0.334	0.242	0.193	0.186
$N = 8$	0.315	0.227	0.189	0.157

(b) Without emitter

	$\sigma = 8$ mm	16 mm	24 mm	32 mm
$N = 2$	0.335	0.268	0.206	0.180
$N = 4$	0.313	0.228	0.180	0.170
$N = 8$	0.296	0.210	0.173	0.168

TABLE I: The RMSE between target and reconstructed pressure for different numbers of points N and different radii σ of the pressure model (columns) for two simulation conditions. Lower values indicate higher reconstruction quality. In (a) the hand interacts with a smoke emitter; in (b) the hand stirs the smoke and there is no emitter.

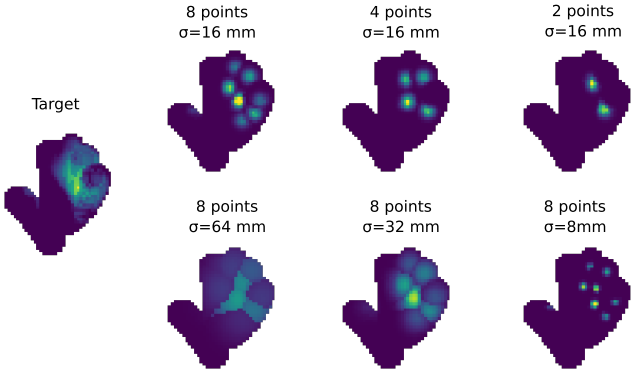

Fig. 3: How pressure reconstruction changes with differing numbers of focal points (top) and focal radii (bottom).

field, optimize the focal pressure points and magnitudes, and output these commands to the haptic device driver.

We set the fluid simulation rate at the highest possible multiple of 30Hz; in our implementation, 90Hz. In this way, we maximize the hand speed that can be robustly handled by the simulation. With a domain $\mathbf{D}$ of $0.5 \times 0.5 \times 0.5$m discretized by $100 \times 100 \times 100$ cells, the CFL condition translates into a maximum hand velocity of 0.45m/s. In practice, due to the numerical dissipation of the simulation, we support even higher hand speed.

We run the fluid simulation and the haptic rendering on the same thread, with 3 fluid simulation steps per haptic update. We add a one-frame delay to the tracked hand positions, and interpolate them at in-between fluid simulation steps.

We have executed our rendering algorithm on an AMD Ryzen 7 2700 8-core 3.20 GHz PC with 32 GB of RAM and a Nvidia GeForce GTX 1080 Ti GPU with 11 GB of RAM. For ultrasound rendering, we have used an Ultrahaptics STRATOS Explore (USX) device, running at 40kHz, which also features a Leap Motion device used for hand pose tracking.

Fig. 2 shows screen-captures of example interactions. In all the captures, we compare the target pressure fields (lower left insets) and the pressure fields resulting from our optimization algorithm and reconstructed according to the model described in Section III-A (lower right insets). The results have been produced with $N = 8$ focal points and a pressure fall-off with $\sigma = 16$mm.

B. Algorithm Evaluation

According to the specification of the STRATOS device, it supports amplitude modulation of up to 8 focal points and a frequency of 40kHz, which corresponds to a focal diameter of 8.6mm. Based on these values, we have evaluated the rendering quality of our algorithm with different numbers of focal points N, as well as different fall-off distances σ (which could be related to the focal radius). We used the manufacturer-recommended setting of 200 Hz for amplitude modulation, and typically settled on $N = 4$ focal points, since the power of individual points diminishes as more points are used. Using pre-recorded hand trajectories for the smoke stirring scene and the smoke jet scene, we have computed the root mean square error (RMSE) of the pressure field reconstruction under the different parameter settings.

Table I summarizes the RMSE results. A clear trend is visible towards a better reconstruction of the target values for an increasing number of points and larger radii. However, in practice, reconstruction radii must be based on the area of the hand affected by an individual focal point and thus selected based on the device capabilities.

Fig. 3 compares the reconstruction quality under the various parameter settings for one particular target pressure field. The images show that as focal point radius increases with respect to the Voronoi region size, the reconstructed values flatten and thus the edges between boundaries decrease in realism with respect to rendering. Thus, the focal point radii should be selected to give good results with respect to their distribution as well as separation.

C. Timings

Timing information for the algorithm can be found in Table II. Timing data was collected and averaged over one minute of simulation while looping the same trajectory used in the evaluation described in the previous subsection. Each simulation step takes roughly 6.21ms, and each optimization step 8.5ms. Recall that the simulation runs at 90Hz and the optimization and haptic rendering at 30Hz. The total computation time per haptic update is then 27.13ms.

Timings per simulation step

Obstacles rasterization:	0.44 ms
Advection:	0.42 ms
Pressure:	3.58 ms
Boundaries and vorticity:	1.77 ms

Timings per optimization step

Pressure extraction:	1.50 ms
Transferring to host:	2.65 ms
Optimization:	4.35 ms

TABLE II: Timing data

VI. Conclusions

In this work we introduce the ability to interact with fluid simulations using ultrasound haptics. To achieve this, we have designed a novel method to display a pressure field on the surface of the hand. We propose to find an optimal set of focal points, minimizing the difference between the reconstructed pressure field and the target pressure obtained from the fluid simulation.

Choices of fall-off radius and number of focal points were explored with respect to the expected reconstruction accuracy. However, to fully evaluate the display method, a perceptual evaluation is required, not only in these variables, but also with regards to spatiotemporal aspects that may lead to experience these moving focal points as a representation of a dynamic field. We expect research in this direction to lead to formulations that bridge the amplitude modulation and spatiotemporal modulation control methods for ultrasound haptics.

VII. Acknowledgements

Funding for this project comes from ERC Consolidator Grant "TouchDesign" (Grant ID 772738). The authors would also like to thank Rosa M. Sánchez-Banderas for help with the demonstration video.

References

[1] William Baxter and Ming C Lin. Haptic interaction with fluid media. In *Proceedings of graphics interface 2004*, pages 81–88. Canadian Human-Computer Communications Society, 2004.

[2] Tom Carter, Sue Ann Seah, Benjamin Long, Bruce Drinkwater, and Sriram Subramanian. Ultrahaptics: Multi-point mid-air haptic feedback for touch surfaces. In *Proceedings of the 26th Annual ACM Symposium on User Interface Software and Technology*, UIST '13, pages 505–514. ACM, 2013.

[3] Gabriel Cirio, Maud Marchal, Anatole Lécuyer, and Jeremy R Cooperstock. Vibrotactile rendering of splashing fluids. *IEEE transactions on haptics*, 6(1):117–122, 2013.

[4] Gabriel Cirio, Maud Marchal, Miguel A Otaduy, and Anatole Lécuyer. Six-dof haptic interaction with fluids, solids, and their transitions. In *World Haptics Conference (WHC), 2013*, pages 157–162. IEEE, 2013.

[5] Keenan Crane, Ignacio Llamas, and Sarah Tariq. Real-time simulation and rendering of 3d fluids. *GPU gems*, 3(1), 2007.

[6] Yoshinori Dobashi, Makoto Sato, Shoichi Hasegawa, Tsuyoshi Yamamoto, Mitsuaki Kato, and Tomoyuki Nishita. A fluid resistance map method for real-time haptic interaction with fluids. In *Proceedings of the ACM Symposium on Virtual Reality Software and Technology*, VRST '06, pages 91–99. ACM, 2006.

[7] Ronald Fedkiw, Jos Stam, and Henrik Wann Jensen. Visual simulation of smoke. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pages 15–22. ACM, 2001.

[8] William Frier, Damien Ablart, Jamie Chilles, Benjamin Long, Marcello Giordano, Marianna Obrist, and Sriram Subramanian. Using spatiotemporal modulation to draw tactile patterns in mid-air. In Domenico Prattichizzo, Hiroyuki Shinoda, Hong Z. Tan, Emanuele Ruffaldi, and Antonio Frisoli, editors, *Haptics: Science, Technology, and Applications*, pages 270–281. Springer International Publishing, 2018.

[9] William Frier, Dario Pittera, Damien Ablart, Marianna Obrist, and Sriram Subramanian. Sampling strategy for ultrasonic mid-air haptics. In *CHI Conference on Human Factors in Computing Systems Proceedings*. ACM, 2019.

[10] Sidhant Gupta, Dan Morris, Shwetak N. Patel, and Desney Tan. Airwave: Non-contact haptic feedback using air vortex rings. In *Proceedings of the 2013 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, UbiComp '13, pages 419–428. ACM, 2013.

[11] K. Hasegawa and H. Shinoda. Aerial vibrotactile display based on multiunit ultrasound phased array. *IEEE Transactions on Haptics*, 11(3):367–377, 2018.

[12] T. Hoshi, M. Takahashi, T. Iwamoto, and H. Shinoda. Noncontact tactile display based on radiation pressure of airborne ultrasound. *IEEE Transactions on Haptics*, 3(3):155–165, 2010.

[13] S. Inoue, Y. Makino, and H. Shinoda. Active touch perception produced by airborne ultrasonic haptic hologram. In *2015 IEEE World Haptics Conference (WHC)*, pages 362–367, 2015.

[14] Takayuki Iwamoto, Mari Tatezono, and Hiroyuki Shinoda. Non-contact method for producing tactile sensation using airborne ultrasound. In *Proceedings of the 6th International Conference on Haptics: Perception, Devices and Scenarios*, EuroHaptics '08, pages 504–513. Springer-Verlag, 2008.

[15] Brian Kappus and Ben Long. Spatiotemporal modulation for mid-air haptic feedback from an ultrasonic phased array. *The Journal of the Acoustical Society of America*, 143(3):1836–1836, 2018.

[16] G. Korres and M. Eid. Haptogram: Ultrasonic point-cloud tactile stimulation. *IEEE Access*, 4:7758–7769, 2016.

[17] H. Lee, J. Kim, S. Choi, J. Jun, J. Park, A. Kim, H. Oh, H. Kim, and S. Chung. Mid-air tactile stimulation using laser-induced thermoelastic effects: The first study for indirect radiation. In *2015 IEEE World Haptics Conference (WHC)*, pages 374–380, 2015.

[18] S. Lloyd. Least squares quantization in pcm. *IEEE Trans. Inf. Theor.*, 28(2):129–137, 2006.

[19] Benjamin Long, Sue Ann Seah, Tom Carter, and Sriram Subramanian. Rendering volumetric haptic shapes in mid-air using ultrasound. *ACM Transactions on Graphics (TOG)*, 33(6):181, 2014.

[20] Yasuaki Monnai, Keisuke Hasegawa, Masahiro Fujiwara, Kazuma Yoshino, Seki Inoue, and Hiroyuki Shinoda. Haptomime: Mid-air haptic interaction with a floating virtual screen. In *Proceedings of the 27th Annual ACM Symposium on User Interface Software and Technology*, UIST '14, pages 663–667. ACM, 2014.

[21] Javier Mora and Won-Sook Lee. Real-time 3d fluid interaction with a haptic user interface. In *3D User Interfaces, 2008. 3DUI 2008. IEEE Symposium on*, pages 75–81. IEEE, 2008.

[22] Miguel A. Otaduy, Allison Okamura, and Sriram Subramanian. Haptic technologies for direct touch in virtual reality. In *ACM SIGGRAPH 2016 Courses*, SIGGRAPH '16, pages 13:1–13:123. ACM, 2016.

[23] Rajinder Sodhi, Ivan Poupyrev, Matthew Glisson, and Ali Israr. Aireal: Interactive tactile experiences in free air. *ACM Trans. Graph.*, 32(4):134:1–134:10, 2013.

[24] Jos Stam. Stable fluids. In *Proceedings of the 26th annual conference on Computer graphics and interactive techniques*, pages 121–128. ACM Press/Addison-Wesley Publishing Co., 1999.

[25] John Steinhoff and David Underhill. Modification of the euler equations for "vorticity confinement": Application to the computation of interacting vortex rings. *Physics of Fluids*, 6(8):2738–2744, 1994.

[26] Graham Wilson, Thomas Carter, Sriram Subramanian, and Stephen A. Brewster. Perception of ultrasonic haptic feedback on the hand: Localisation and apparent motion. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '14, pages 1133–1142. ACM, 2014.

[27] Meng Yang, Jingwan Lu, Alla Safonova, and Katherine J Kuchenbecker. Gpu methods for real-time haptic interaction with 3D fluids. In *Haptic Audio visual Environments and Games, 2009. HAVE 2009. IEEE International Workshop on*, pages 24–29. IEEE, 2009.

[28] Q. Zhang, H. Dong, and A. El Saddik. Magnetic field control for haptic display: System design and simulation. *IEEE Access*, 4:299–311, 2016.