Extended Abstracts of the
25th International Symposium on Mathematical Theory of Networks and Systems
Bayreuth, Germany, 12-16 September 2022

Content Rendering for Acoustic Levitation Displays via Optimal Path Following

Viktorija Paneva * **Arthur Fleig** * **Diego Martínez Plasencia** **
Timm Faulwasser *** **Jörg Müller** *

* *Department of Computer Science, University of Bayreuth, DE 95447 (e-mail: vpaneva@acm.org, arthur.fleig@uni-bayreuth.de, joerg.mueller@uni-bayreuth.de).*
** *Computer Science, University College London, UK WC1E 6BT (e-mail: d.plasencia@ucl.ac.uk)*
*** *Department of Electrical Engineering and Information Technology, TU Dortmund University, DE 44227 (e-mail: timm.faulwasser@ieee.org).*

Abstract: Recently, volumetric displays based on acoustic levitation have demonstrated the capability to produce mid-air content using the Persistence of Vision (PoV) effect. In these displays, acoustic traps are used to rapidly move a small levitated particle along a prescribed path. This note is based on our recent work *OptiTrap* (Paneva et al., 2022), the first structured numerical approach for computing trap positions and timings via optimal control to produce feasible and (nearly) time-optimal trajectories that reveal generic levitated graphics. While previously, feasible trap trajectories needed to be tuned manually for each shape and levitator, relying on trial and error, *OptiTrap* automates this process by allowing for a systematic exploration of the range of contents that a given levitation display can render. This represents a crucial milestone for future content authoring tools for acoustic levitation displays and advances volumetric displays closer toward real-world applications.

Keywords: Ultrasonic levitation, Minimum time problems, Path following, Optimal control

1. INTRODUCTION

Acoustic levitation displays use ultrasonic waves to trap small particles in mid-air, acting as volumetric pixels (voxels). Several practical aspects have been investigated around these displays, such as low-latency particle manipulation (Bachynskyi et al., 2018), and content detection and initialisation (Fender et al., 2021).

The ability to move single (Hirayama et al., 2019) or multiple (Plasencia et al., 2020) levitated particles at very high speeds was instrumental for achieving dynamic and freeform volumetric content. However, this was nonetheless limited to relatively small sizes and simple vector graphics (Fushimi et al., 2020). Little effort was made towards optimising the levitated visual content while considering the system dynamics of such displays, particularly for challenging content such as the one created by levitated particles moving at PoV speeds. Paneva et al. (2020) proposed an interactive simulation of a levitation interface, using a model of the particle movement in such a display. The application operates in a feed-forward manner, simulating the dynamics of the particle given a specific path for the traps, however, it does not address the inverse problem.

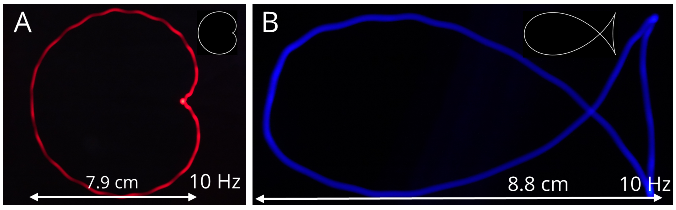

Fig. 1. Shapes involving sharp edges and significant changes in curvature demonstrated using acoustic levitation.

OptiTrap (Paneva et al., 2022) is the first algorithm allowing the definition of generic PoV content, requiring only a geometric definition (i.e., shape to present, no timing information) and optimising it according to the capabilities of the device and the dynamics of the trap-particle system. *OptiTrap* automates the definition of levitated PoV content, computing physically feasible and nearly time-optimal trap trajectories given only a reference path. This allows for larger shapes than previously demonstrated, as well as shapes featuring significant changes in curvature and/or sharp corners (Figure 1).

2. OPTIMAL CONTROL FOR LEVITATION DISPLAYS

2.1 Hardware Setup

Our setup, illustrated in Figure 2, consists of two arrays of 16 × 16 transducers facing each other, controlled by an FPGA, and an OptiTrack[1] tracking system - Prime 13 motion capture cameras operating at a frequency of 240Hz.

[1] www.optitrack.com

With an update rate of up to 10kHz, the device can create a single twin-trap within the levitation volume using the method from Hirayama et al. (2019). We experimentally determined the vertical and horizontal forces exerted on the particle and found that the vertical force is double the horizontal. This difference needs to be reflected by the model that is introduced in the next section.

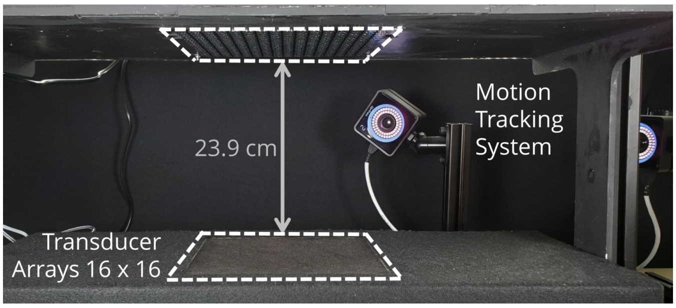

Fig. 2. Overview of the components of our levitation display.

2.2 *Modelling the Trap-Particle Dynamics*

To model the trap-particle dynamics, we use simple Newtonian mechanics, i.e.,

$$m\ddot{\mathbf{p}}(t) = F(\mathbf{p}(t), \dot{\mathbf{p}}(t), \mathbf{u}(t)), \tag{1}$$

where $\mathbf{p}(t) = (p_x(t), p_y(t), p_z(t))^\top \in \mathbb{R}^3$ represents the particle position in Cartesian (x, y, z) coordinates at time $t \in \mathbb{R}_0^+$, m is the particle mass, $\mathbf{u}(t) = (u_x(t), u_y(t), u_z(t))^\top \in \mathbb{R}^3$ is the control input specifying the position of the acoustic trap, and F is the net force acting on the particle. Following Hirayama et al. (2019), we neglect drag and gravitational forces due to the dominating acoustic radiation forces.

In many cases, the acoustic force can be described by the gradient of the Gor'kov potential (Bruus, 2012). Our specific setup (top-bottom transducer placement and vertical twin traps) and objective (find where to place the acoustic trap to produce a specific force) allows to consider only a region around the peak forces. In this region, the forces in our setup distribute mostly axis-symmetrically. Hence, we approximate the force as

$$F(\mathbf{p}, \mathbf{u}) = \begin{pmatrix} F_x(\mathbf{p}, \mathbf{u}) \\ F_y(\mathbf{p}, \mathbf{u}) \\ F_z(\mathbf{p}, \mathbf{u}) \end{pmatrix} := \begin{pmatrix} F_r(\mathbf{p}, \mathbf{u}) \cos\phi \\ F_r(\mathbf{p}, \mathbf{u}) \sin\phi \\ F_z(\mathbf{p}, \mathbf{u}) \end{pmatrix}, \tag{2}$$

where

$$F_r(\mathbf{p}, \mathbf{u}) := \mathcal{A}_r \cdot \cos\left(\mathcal{V}_z \cdot (u_z - p_z)\right) \cdot \sin\left(\mathcal{V}_{xr} \cdot \sqrt{(u_x - p_x)^2 + (u_y - p_y)^2}\right), \tag{3a}$$

$$F_z(\mathbf{p}, \mathbf{u}) := \mathcal{A}_z \cdot \sin\left(\mathcal{V}_z \cdot (u_z - p_z)\right) \cdot \cos\left(\mathcal{V}_{zr} \cdot \sqrt{(u_x - p_x)^2 + (u_y - p_y)^2}\right), \tag{3b}$$

$$\phi = \arctan\left(\frac{u_y - p_y}{u_x - p_x}\right),$$

and where $\mathcal{A}_r, \mathcal{A}_z$ denote the peak forces along the radial and vertical directions of the trap, respectively, and $\mathcal{V}_z$, $\mathcal{V}_{xr}$, $\mathcal{V}_{zr}$ are the characteristic frequencies of the sinusoidals describing how the forces evolve around the trap. For more details on this approximation procedure we refer to the original paper (Paneva et al., 2022).

2.3 *Rendering Content via Path Following*

The task at hand is to render, as fast as possible, arbitrary complex objects, formulated as an explicitly parameterised curve

$$Q := \{\xi \in \mathbb{R}^3 \mid \theta \in [\theta_0, \theta_f] \mapsto \mathbf{q}(\theta)\}, \tag{4}$$

where we require $\mathbf{q} \in C^2(\mathbb{R}; \mathbb{R}^3)$. The path parameter θ models the progress on the path from the starting point $\mathbf{q}(\theta_0)$ to the end point $\mathbf{q}(\theta_f)$. To create the PoV effect, we consider periodic paths, for which $\mathbf{q}(\theta_0) = \mathbf{q}(\theta_f)$ and $\dot{\mathbf{q}}(\theta_0) = \dot{\mathbf{q}}(\theta_f)$ hold. For example, consider the shape of the cardioid in Figure 1 (left), which can be described by $\mathbf{q}(\theta) = (0, r\sin(\theta)(1 + \cos(\theta)), -r\cos(\theta)(1 + \cos(\theta)) + r)^\top$, where $\theta \in [0, 2\pi]$ and $r > 0$.

Since Q comes without any preassigned time information, we need to determine the timing $t \mapsto \theta(t)$. Following Faulwasser (2012); Faulwasser et al. (2017), we assume the particle follows the path Q exactly at all times, i.e., $\mathbf{p}(t) - \mathbf{q}(\theta(t)) \equiv 0$. This leads to

$$\mathbf{p}(t) = \mathbf{q}(\theta(t)), \tag{5a}$$

$$\dot{\mathbf{p}}(t) = \dot{\mathbf{q}}(\theta(t)) = \frac{\partial \mathbf{q}}{\partial \theta}\dot{\theta}(t), \tag{5b}$$

$$\ddot{\mathbf{p}}(t) = \ddot{\mathbf{q}}(\theta(t)) = \frac{\partial^2 \mathbf{q}}{\partial \theta^2}\dot{\theta}(t)^2 + \frac{\partial \mathbf{q}}{\partial \theta}\ddot{\theta}(t). \tag{5c}$$

Using a virtual function $v(t) \in \mathbb{R}$ to control the progress of the particle along Q, the timing law is modelled as a double integrator

$$\ddot{\theta}(t) = v(t) \tag{6}$$

to avoid large jumps in the acceleration. To keep the periodic nature, we impose

$$\theta(0) = \theta_0, \quad \theta(T) = \theta_f, \quad \dot{\theta}(0) = \dot{\theta}(T), \tag{7}$$

where the traversal time T will be an optimisation variable in the latter optimal control problem (12). Using

$$\mathbf{z}(t) := (\theta(t), \dot{\theta}(t))^\top$$

we rewrite (6)-(7) as

$$\dot{\mathbf{z}}(t) = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}\mathbf{z}(t) + \begin{pmatrix} 0 \\ 1 \end{pmatrix}v(t), \ \mathbf{z}(0) = \mathbf{z}_0, \ \mathbf{z}(T) = \mathbf{z}_T, \tag{8}$$

solvable with standard Runge-Kutta methods (Butcher, 2016).

2.4 *Coupling Path Following with Trap-Particle Dynamics*

To render content on the levitator, we couple the path-following approach from Section 2.3 with the trap-particle dynamics (1)-(3) from Section 2.2. The usual approach of rewriting (1) as

$$M(\ddot{\mathbf{p}}(t), \mathbf{p}(t), \mathbf{u}(t)) := m\ddot{\mathbf{p}}(t) - F(\mathbf{p}(t), \mathbf{u}(t)) = 0 \tag{9}$$

and finding a local inversion $\mathbf{u}(t) = M^{-1}(\ddot{\mathbf{p}}(t), \mathbf{p}(t))$ is not straightforward for (3). Hence, we tackle this task via numerical optimisation by introducing auxiliary variables

$$\zeta_1 := \sin(\mathcal{V}_{xr}\sqrt{(u_x - q_x(\theta))^2 + (u_y - q_y(\theta))^2}), \tag{10a}$$
$$\zeta_2 := \cos\mathcal{V}_z \cdot (u_z - q_z(\theta)), \tag{10b}$$
$$\zeta_3 := \sin\mathcal{V}_z \cdot (u_z - q_z(\theta)), \tag{10c}$$
$$\zeta_4 := \cos\mathcal{V}_{zr}\sqrt{(u_x - q_x(\theta))^2 + (u_y - q_y(\theta))^2}), \tag{10d}$$
$$\zeta_5 := \sin\phi, \tag{10e}$$
$$\zeta_6 := \cos\phi, \tag{10f}$$

for each trigonometric term in (3), where p_i is replaced by $q_i(\theta)$, $i \in \{x, y, z\}$. This allows us to formally express (2) in terms of $\zeta := (\zeta_1, ..., \zeta_6)$:

$$\tilde{F}(\zeta) := \begin{pmatrix} \mathcal{A}_r\zeta_1\zeta_2\zeta_6 \\ \mathcal{A}_r\zeta_1\zeta_2\zeta_5 \\ \mathcal{A}_z\zeta_4\zeta_3 \end{pmatrix}.$$

Similar to (9), along the path Q we define

$$\widetilde{M}(\mathbf{z}(t), v(t), \zeta(t)) := m\ddot{\mathbf{q}}(\theta(t)) - \tilde{F}(\zeta(t)) = 0. \tag{11}$$

With this approach, we will need to extract the trap positions $\mathbf{u}(t)$ by solving (10) numerically. To counter numerical instabilities that could occur in particular for ζ_i approaching ±1, we introduce additional constraints

$$\mathcal{Z} := \left\{\zeta \in [\varepsilon - 1, 1 - \varepsilon]^6 \mid \zeta_2^2 + \zeta_3^2 = 1,\ \zeta_5^2 + \zeta_6^2 = 1\right\},$$

with a user-chosen back-off parameter $\varepsilon \in \,]0, 1[$ that captures the trade-off between numerically "stable" solutions for $\mathbf{u}(t)$ and exploiting the maximum forces of the device.

The final OCP is then given by

$$\begin{aligned} \min_{v,T,\zeta}\ & T + \gamma \int_0^T v(t)^2 \mathrm{d}t \\ \text{subject to } & (8), \\ & \widetilde{M}(\mathbf{z}(t), v(t), \zeta(t)) = 0, \\ & \zeta(t) \in \mathcal{Z}. \end{aligned} \tag{12}$$

Lastly, we discretised the final OCP and solved the resulting nonlinear programming problem using Ipopt (Andersson et al., 2018). The function evaluations and the computation of the derivatives were performed with CasADi (Wächter and Biegler, 2006).

3. EVALUATION

Next we compare our *OptiTrap* approach against a *Baseline*, where the path parameter is homogeneously sampled and the traps are placed directly on the reference path of the particle. The evaluation was conducted on four test shapes: circle, cardioid, squircle and fish, as shown in Figures 3 and 4. The particle motion shown in these figures comes not from simulations, but from actual experimental results, captured using the OptiTrack tracking system.

In the first part of the evaluation, we investigate the maximum shape size that can be rendered in PoV time (0.1s) with each approach. We see in Figure 3 that the most striking difference was obtained for the squircle, where with *OptiTrap* 1.9 meters of content per second was rendered, and with the *Baseline* only 0.29. The increase in size was similar for both the cardioid (2.80m of content per s with *OptiTrap*, 2.48 with the *Baseline*) and the fish (2.75m of content per s with *OptiTrap*, 2.42 with the *Baseline*), while there was not significant difference for the circle. This is not surprising, as the circle is the simplest and most homogeneous shape in the test parkour.

In the second part of the evaluation, we investigate the maximum possible rendering frequency with *OptiTrap* and the *Baseline*, while keeping the size of the test shape constant. The results are illustrated in Figure 4. As expected,

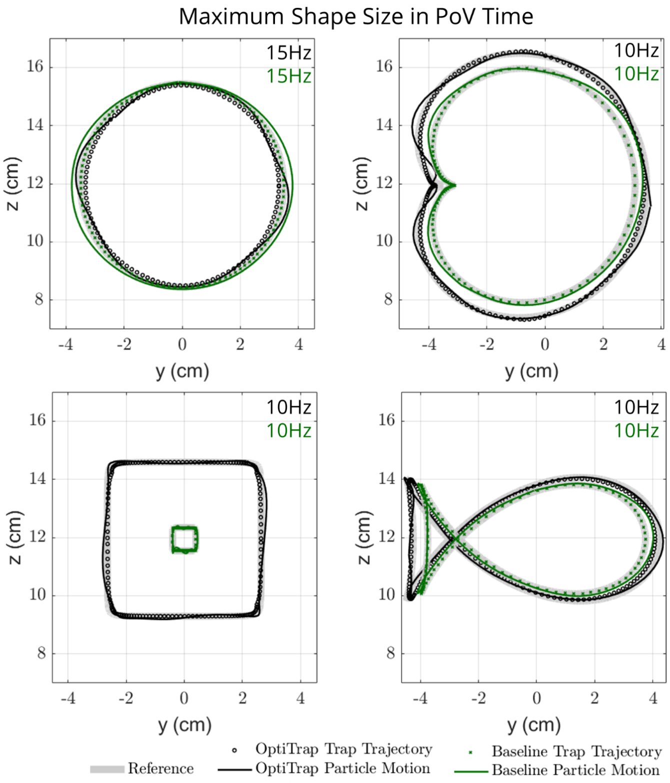

Fig. 3. Maximum shape size of *OptiTrap* vs. the Baseline.

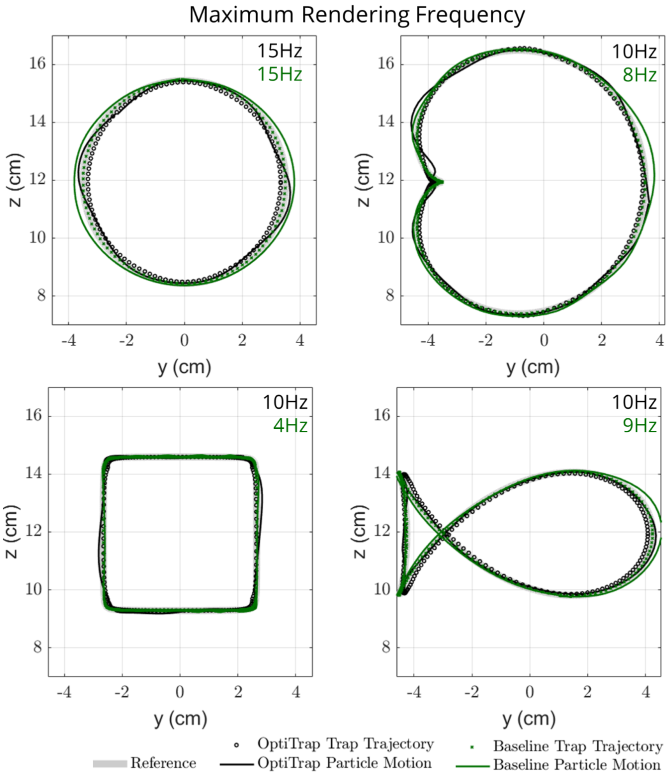

Fig. 4. Maximum rendering frequency of *OptiTrap* vs. the Baseline.

the same rendering frequency was obtained for the circle using both methods. However, the cardioid was rendered at 10Hz using *OptiTrap*, and 8Hz with the *Baseline* (25% increase). The rendering frequency for the fish increased by 11% using *OptiTrap*, i.e., from 9 to 10Hz, and lastly, we obtain an increase of 150% for the squircle.

For a more extensive and detailed evaluation of *OptiTrap*, also including a comparison to a more sophisticated baseline, please refer to Paneva et al. (2022).

4. CONCLUSION

We briefly discussed *OptiTrap* – a structured numerical approach to compute trap trajectories for acoustic levitation displays. *OptiTrap* automatically computes physically feasible and nearly time-optimal trap trajectories to reveal generic levitated content in mid-air, assuming only a reference path. This is a particularly important step for the adoption of PoV levitation displays, as it allows the content designers to focus on the shapes to be rendered, with feasible solutions taking into account the capabilities of the specific device, being computed automatically by the algorithm. As such, *OptiTrap* has the potential to become an instrumental tool in helping to further explore and develop these displays. In the future, this method can be extended to include visual content composed of multiple levitated particles, it can be applied to other domains, such as photophoretic displays or containerless matter transportation for applications in pharmacy and biochemistry, or can be used as a base for developing more complex learning-based approaches.

ACKNOWLEDGEMENTS

This project was partially supported by the European Union's Horizon 2020 research and innovation programme under grant agreement #737087 (Levitate)

REFERENCES

Andersson, J.A.E., Gillis, J., Horn, G., Rawlings, J.B., and Diehl, M. (2018). CasADi – A software framework for nonlinear optimization and optimal control. *Mathematical Programming Computation.*

Bachynskyi, M., Paneva, V., and Müller, J. (2018). Levicursor: Dexterous interaction with a levitating object. In *Proceedings of the 2018 ACM International Conference on Interactive Surfaces and Spaces*, ISS '18, 253–262. ACM, New York, NY, USA. doi:10.1145/3279778.3279802. URL `http://doi.acm.org/10.1145/3279778.3279802`.

Bruus, H. (2012). Acoustofluidics 7: The acoustic radiation force on small particles. *Lab Chip*, 12, 1014–1021. doi:10.1039/C2LC21068A.

Butcher, J.C. (2016). *Numerical methods for ordinary differential equations.*

Faulwasser, T. (2012). *Optimization-based solutions to constrained trajectory-tracking and path-following problems.* Ph.D. thesis, Otto-von-Guericke-Universität Magdeburg, Universitaetspl. 2.

Faulwasser, T., Weber, T., Zometa, P., and Findeisen, R. (2017). Implementation of nonlinear model predictive path-following control for an industrial robot. *IEEE Trans. Contr. Syst. Techn.*, 25(4), 1505–1511. doi:10.1109/TCST.2016.2601624.

Fender, A.R., Martinez Plasencia, D., and Subramanian, S. (2021). *ArticuLev: An Integrated Self-Assembly Pipeline for Articulated Multi-Bead Levitation Primitives.* Association for Computing Machinery, New York, NY, USA. URL `https://doi.org/10.1145/3411764.3445342`.

Fushimi, T., Drinkwater, B.W., and Hill, T.L. (2020). What is the ultimate capability of acoustophoretic volumetric displays? *Applied Physics Letters*, 116(24), 244101. doi:10.1063/5.0008351. URL `https://doi.org/10.1063/5.0008351`.

Hirayama, R., Plasencia, D.M., Masuda, N., and Subramanian, S. (2019). A volumetric display for visual, tactile and audio presentation using acoustic trapping. *Nature*, 575(7782), 320–323.

Paneva, V., Bachynskyi, M., and Müller, J. (2020). Levitation simulator: Prototyping for ultrasonic levitation interfaces in virtual reality. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, CHI '20. ACM, New York, NY, USA.

Paneva, V., Fleig, A., Plasencia, D.M., Faulwasser, T., and Müller, J. (2022). Optitrap: Optimal trap trajectories for acoustic levitation displays. *ACM Trans. Graph. (In Press)*. doi:10.1145/3517746. URL `http://dx.doi.org/10.1145/3517746`.

Plasencia, D.M., Hirayama, R., Montano-Murillo, R., and Subramanian, S. (2020). Gs-pat: High-speed multi-point sound-fields for phased arrays of transducers. *ACM Trans. Graph.*, 39(4). doi:10.1145/3386569.3392492.

Wächter, A. and Biegler, L.T. (2006). On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. *Mathematical programming*, 106(1), 25–57.