A Control Method for Steering Individual Particles Inside Liquid Droplets Actuated by Electrowetting

Shawn Walker[1] and Benjamin Shapiro[2]
University of Maryland at College Park

1. Abstract

An algorithm is developed that allows steering of individual particles inside electrowetting systems by control of actuators already present in these systems. The developed control algorithm permits additional capabilities in electrowetting systems including the ability to precisely place particles such as cells at localized sensors, to precisely move particles from one location to another at rates much faster than those created by diffusion, and to steer and sort particles within a single drop or across splitting drops. Particles are steered by creating time varying flow fields that carry the particles along their desired trajectories. Results are demonstrated using an experimentally validated model that includes two phase liquid fluid dynamics, surface tension and electrowetting interfacial forces, as well as essential loss mechanisms due to contact angle saturation, contact line pinning, and contact angle hysteresis. It is shown that the current UCLA electro-wetting-on-dielectric (EWOD) system contains enough control authority to steer a single particle along arbitrary trajectories and to steer two particles at once along simple paths. Particle steering limits are set by contact angle saturation and by the small number of actuators that are available to actuate the flow in practical electrowetting systems.

[1] Aerospace Engineering, Research Assistant, University of Maryland at College Park, MD 20852. swalker@wam.umd.edu, (301) 405-1998 (author to whom all correspondence should be addressed).
[2] Aerospace Engineering, joint appointment with Bio-Engineering graduate program and Institute of Systems Research, University of Maryland at College Park, MD 20852. benshap@eng.umd.edu, (301) 405-4191.

2. Introduction

We show that electro-wetting-on-dielectric (EWOD) systems contain enough control authority to steer individual particles on trajectories inside the liquid drops. It is possible, for example, to actuate the available electrodes surrounding a single droplet in such a way that the resulting fluid flow inside the drop will carry a particle around a figure 8 path or will carry two particles along separate trajectories.

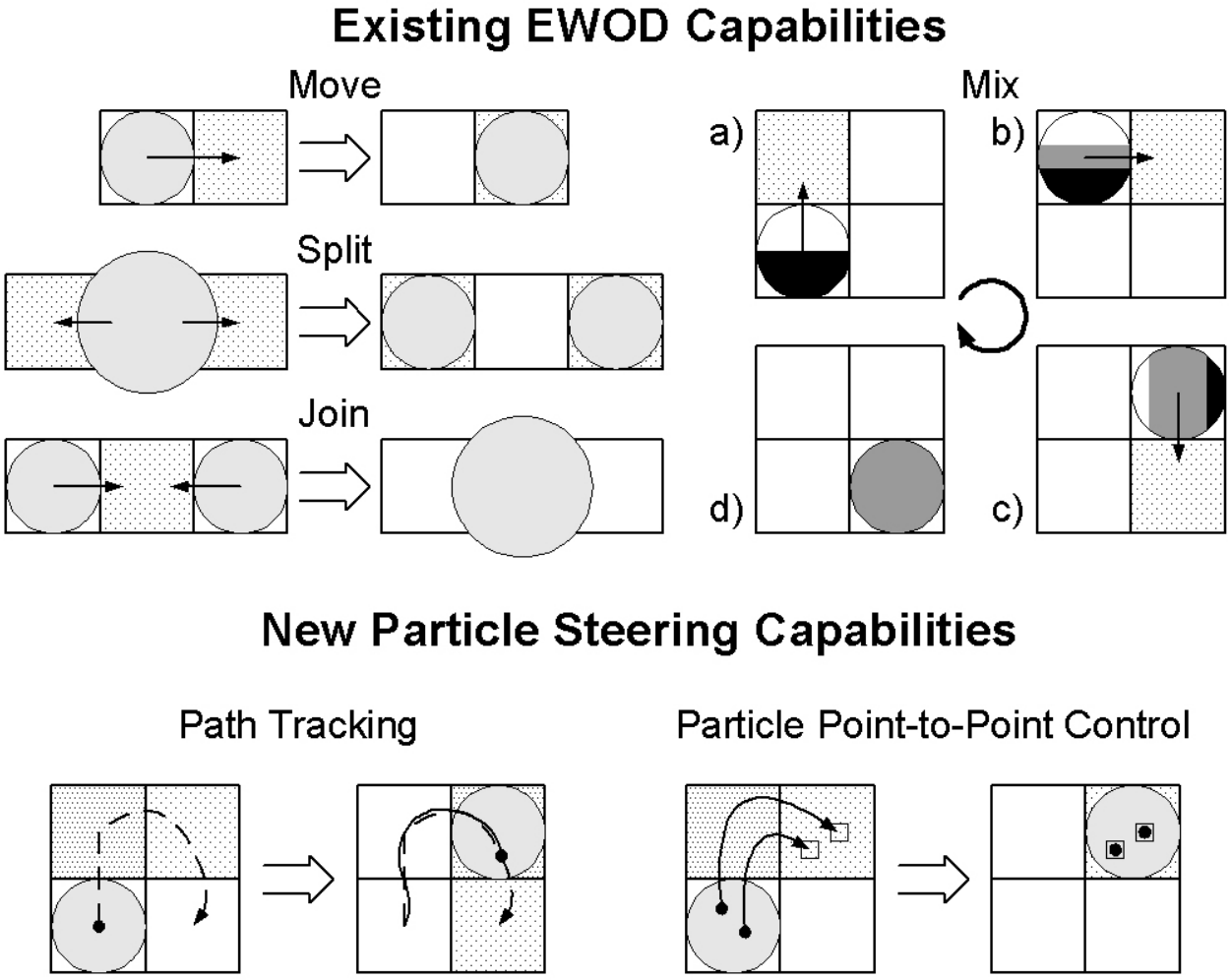


Figure 1: The EWOD system manipulates fluids by charging a dielectric layer underneath the liquid that effectively changes the local surface tension properties of the liquid/gas interface creating liquid motion. Existing (move, split, join, and mix) capabilities of electrowetting devices are shown schematically above (see [1], [2], [3], [4], [5], [6], [7]) alongside the new particle steering capability developed in this paper. The view is from the top. Shaded circles represent droplets of liquid. Squares are electrodes where the dotted hatching indicates the electrode is on. Directed lines specify the direction of motion. The multi-shaded droplet shows the diffusion and mixing of two chemicals, here mixing is enhanced by the fluid dynamics created inside the droplet due to its imposed motion.

Steering particles inside droplets introduces another level of functionality into electro-wetting systems. By being able to steer individual particles inside droplets to 10 micrometer resolution, it will be possible to precisely place cells under or on top of localized sensors, to stretch out DNA strands (by steering beads attached to the two ends of the strand in different directions), to move particles from one location to another at rates much faster than those created by diffusion, to steer and sort particles within a single drop, or to ensure that certain particles remain in one drop while other particles are steered to a drop that is being split away. In essence, our control methods allow electrowetting systems to achieve some of the same capabilities as laser tweezers [8], [9], [10] although admittedly only in two spatial dimensions and with less accuracy.

Steering results are demonstrated using a numerical model [11] of fluid flow in the UCLA electrowetting system [1], [12]. This model of EWOD fluid dynamics includes surface tension and electrowetting interface forces, viscous low Reynolds 2-phase fluid flow, and the essential loss mechanisms due to contact angle saturation, triple point line pinning, and the related mechanism of contact angle hysteresis. The model has no free parameters (all the parameters are either derived from first principles or are taken from experimental data) and we have validated that the model accurately predicts experimental results in the UCLA devices [11].

To experimentally demonstrate particle steering in the UCLA EWOD systems would require real time implementation of the least squares control algorithm, a vision system to find the location of the particles and to track droplet shapes in real time, and integration and experimental validation of the feedback closed loop architecture shown in Figure 8. Of these tasks, we have already demonstrated a particle tracking vision system, a real time control algorithm implementation, and closed loop feedback integration for steering of particles in a micro fluidic system driven by electroosmotic (as opposed to EWOD) actuation (see Figure 2, [13], and [14]) plus we have developed a (millisecond update) image algorithm to track EWOD droplet shapes in real time. It remains to create a real time control implementation for the EWOD system, to implement our vision algorithm on the UCLA devices, and to achieve the feedback control system integration. These tasks are currently being pursued and will be discussed in future publications. We also note that the size of the vision system, which is currently a microscope and a camera with an image algorithm implemented on a digital signal processing chip, can be reduced to an on-chip contact imager as demonstrated in [15]. This permits the feedback EWOD system, with vision feedback, to remain portable and hand-held.

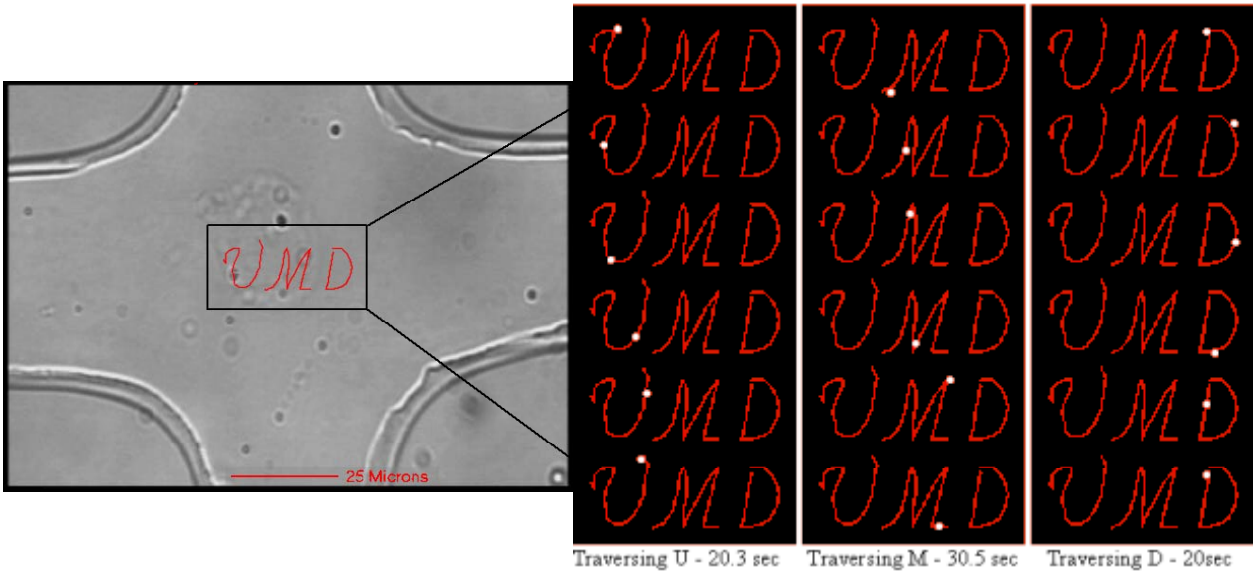


Figure 2: Experimental demonstration of single particle steering by flow control in a PDMS four electrode device driven by *electroosmotic* actuation. Left: Photograph of the micro-fluidic devices with the cursive 'UMD' path super-imposed on the image. Right: The actual path of the target yeast cell (white dot) in the feedback control experiment: the yeast cell tracks the desired path to within one micron. Steering is achieved by creating a fluid flow that carries the cell from one location to the next: besides the four voltages that are used to create this flow there is no other actuation mechanism (such as a laser tweezer or a magnetic bead) that directly addresses the target cell. The purpose of the current paper is to show that electrowetting actuation has a similar capability to steer individual particles by flow control, and to discuss what practical steering tasks can and cannot be achieved (Figure taken from our prior work in [13]).

This paper is organized as follows. In Section 3, we summarize and present validation results for the EWOD model, and we also discuss the dynamics of particles driven by the fluid dynamics. Section 4 outlines the particle steering control task and algorithm development. Finally, Section 5 presents numerical results and discusses which particle trajectories can and cannot be achieved.

3. Summary and Validation of the EWOD Fluid Dynamics Model

A schematic of an EWOD system is shown in Figure 3. The device consists of a sandwich of various layers listed from top to bottom as: top (transparent) electrode, hydrophobic Teflon coating, droplets of water (here only one droplet is shown), another Teflon coating, a layer of solid dielectric silicon dioxide, and an underlying grid of electrodes. There are also spacers to ensure that the channel height is uniform.

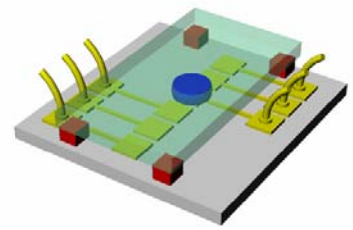


Figure 3: Schematic of an EWOD device. The device consists of multiple layers. The top (transparent) layer is a ground electrode with a hydrophobic coating of Teflon. The bottom layer also has a Teflon coating over a solid dielectric silicon dioxide layer followed by a grid of electrodes with attached wires coming off of the device. The dielectric layer acts as a capacitive electrical energy storage element. The four small cubes are spacers to ensure uniform spacing of the channel. Electrodes are turned on and off, this charges the dielectric layer above each electrode and below the liquid, and the resulting competition between surface tension and dielectric energies causes the droplet to move from one electrode to another [16], [17], [18], [19], [20], [21], [22], [23]. (Figure courtesy of CJ Kim at UCLA.)

Since our particle steering results are demonstrated using a model of electrowetting fluid dynamics, we briefly summarize our prior modeling results and also discuss how particle dynamics, and their actuation, are now included in the model. A complete description of the EWOD fluid model can be found in [11]. We model the fluid flow inside the droplet by making the following assumptions. The liquid is considered to be a continuum because the device length scale is large compared to both the mean free path of air and water molecules. Air flow around the droplet is neglected because it does not significantly affect the droplet motion. Further, since the channel spacing of the device is small compared to its horizontal dimensions, the flow field is modeled in the two planar dimensions and is assumed to be quadratic (thereby satisfying the no slip boundary conditions) in the vertical dimension. Using these assumptions, we derive the fluid equations by starting with the full three-dimensional (3D) Navier-Stokes equations and reducing to a set of 2D Stokes like equations [24]. Essentially, we are able to ignore the nonlinear terms in the Navier-Stokes equations because of the small vertical spacing. This gives the flow field equations inside the droplet with the proper boundary conditions discussed below.

The principle of EWOD operation is that the liquid-gas interface of the droplet can be locally deformed by capacitively charging the silicon dioxide layer underneath it [16], [17], [18], [19], [20], [21], [22], [23]. This changes the mean curvature of the interface and the pressure jump across the interface as quantified by the Young-Laplace relation [25]

(1) $$P = \sigma\left(1/R_1 + 1/R_2\right)$$

where R_1 and R_2 are the principle radii of curvature of the interface, σ is the surface tension coefficient, and P is the pressure. The charged dielectric layer is underneath the liquid and so only the bottom contact angle changes, the top contact angle remains constant. Since the height of the device is very small compared to the width, we approximate the cross-section of the liquid-gas interface as a circle with known contact angles. Quantification of the contact angle dependence on the applied voltage will be discussed below. Once the contact angle is known, equation (1) gives the resulting pressure boundary conditions.

In order to accurately predict experimental results, we include the essential loss mechanisms due to contact angle saturation, contact line pinning, and the related mechanism of contact angle hysteresis. The Young-Lippmann model [18] gives an unrealistic estimate for the contact angle of a droplet on a dielectric plate in terms of the applied voltage. Therefore, we model the contact angle dependence on applied voltage by using data from the UCLA EWOD system [12]. This is necessary to account for contact angle saturation (see Figure 5), which significantly affects the speed of droplet motion. Contact line pinning and contact angle hysteresis are treated by limiting the dynamic range of contact angle variations with a cutoff angle and by weakening the pressure gradient with a fixed multiplier. This is the simplest model of the loss mechanisms that allows us

to accurately predict EWOD fluid dynamic behavior. If these saturation, pinning, and hysteresis effects are not included then our simulation predicts an incorrect droplet splitting shape as well as a split time that is 15 times too fast [11].

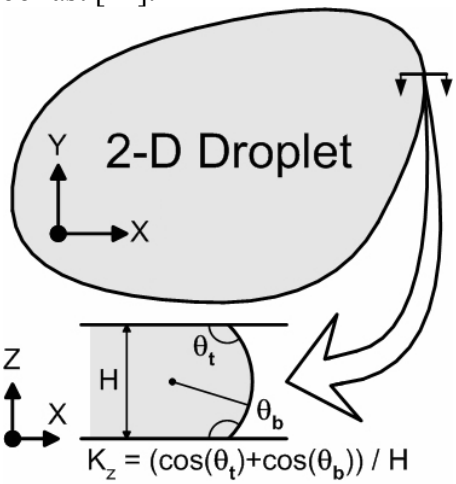


Figure 4: Overhead view of a (2-D) droplet with side view zoom-in of the interface. The liquid-gas interface is assumed to have a circular cross-section. (Figure repeated from [11].)

We now list the final form of the fluid equations in our model. The equations for the pressure field, P, in non-dimensional form, are given by

(2) $\nabla^2 P = 0$, inside the droplet,

(3) $P = \kappa_{xy} - \frac{L}{H}(\cos\theta_t + \cos\theta_b)$, on the boundary of the 2D droplet,

where L is the length scale of the device and H is the spacing between the two plates. The second equation is just the Young-Laplace equation, which says the pressure on the boundary is equal to the mean curvature [26]. The term, κ_{xy}, is the planar curvature of the closed curve that represents the 2D shape of our droplet. The second term is an approximation of the curvature in the z direction assuming the full 3D droplet has a circular cross-section (see Figure 4). The angles, θ_t and θ_b, are the top and bottom contact angles of the droplet. The bottom angle depends on the applied voltage and we use experimental data to model this (see Figure 5).

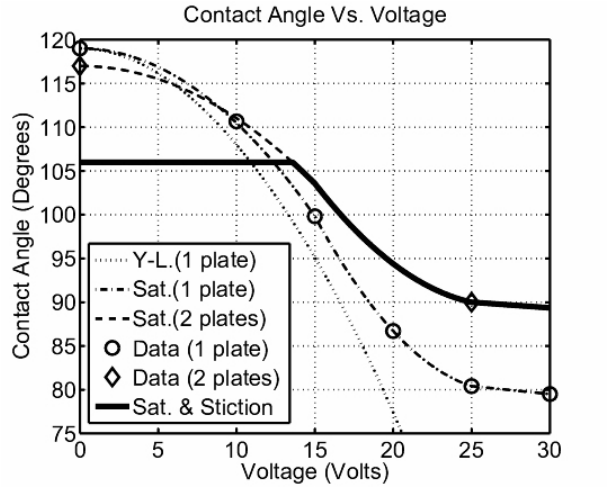


Figure 5: Contact angle versus voltage curves: theoretical and experimental data for contact angle variations under electrowetting actuation. The dotted line denoting the Young-Lippmann (Y-L) curve is theoretical [4], [39], [40]. The single plate saturation curve has six experimental data points (given in [15]) with a piecewise interpolating polynomial (see dot-dashed line and circle data points). The two plate saturation curve has two experimental data points (also from [15]) with an interpolating curve derived from the single plate case using a linear map (see dashed line and diamond data points). The stiction curve is a modification of the two plate saturation curve, which uses a cutoff angle of 106 degrees to emulate contact line pinning (see solid line). This curve is used in our simulation to predict the correct droplet motion and splitting time. (Figure repeated from [11].)

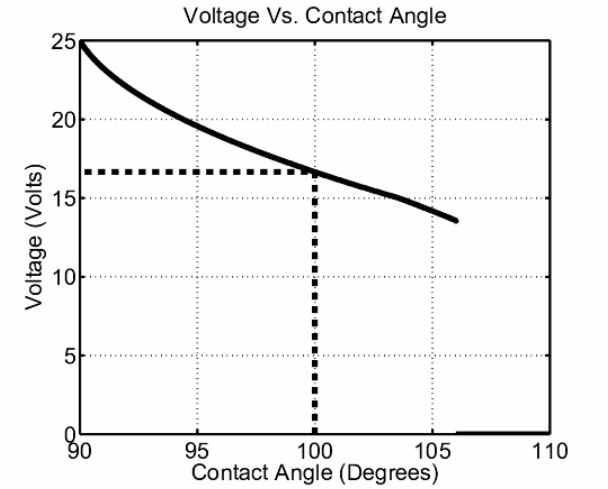


Figure 6: Voltage versus contact angle with all loss effects included. Here we show the inverse mapping of the thick black line in Figure 5 that represents the contact angle variability of the EWOD device when all loss effects are included. The plot shows how to compute the voltage needed to actuate a specific contact angle. The dashed line depicts the mapping from a 100-degree contact angle to a voltage of about 16.5 Volts. We use this in our control algorithm for estimating the necessary actuation voltages (see section 4.4).

The equations for the velocity field, in non-dimensional form, are given by

$$\tau\frac{\partial u}{\partial t}+\beta u=-\frac{K_{hys}}{Ca}\frac{\partial P}{\partial x},\qquad \tau\frac{\partial v}{\partial t}+\beta v=-\frac{K_{hys}}{Ca}\frac{\partial P}{\partial y},\quad \text{inside the droplet,} \tag{4}$$

(5) $\tau = \frac{\rho U_0 L}{\mu}, \quad \beta = 12\left(\frac{L}{H}\right)^2,$

where u and v are the x and y velocity components, ρ = 997 Kg/m^3 and μ = 8.9 x 10^{-4} kg/ms are the density and dynamic viscosity of water, U_0 = 60 mm/s is the nominal velocity scale of the flow, and K_{hys} = 0.3 is a dimensionless constant that models the effects of contact angle hysteresis.

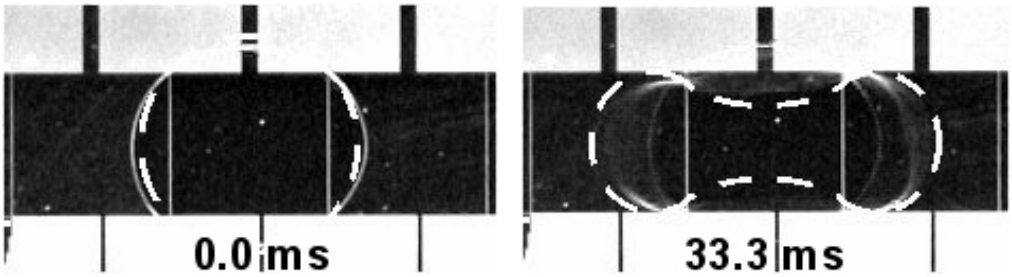
We consider neutral particles that are simply carried along by the (vertically averaged) planar fluid flow. Thus a particle at the location (x,y) will simply follow the velocity of the fluid at its location

(6) $\dot{x} = u(x, y) \qquad \dot{y} = v(x, y),$

where *(u,v)* is the flow field from equation (4) and the dots denote derivatives with respect to time. The pressure gradient in equation (4) can be thought of as a control for manipulating the position of the particle or particles. It is by controlling the pressure gradient field inside the droplet, by changing the pressure on the interface via electrode voltages, that we can achieve particle steering.

The model equations are simulated using finite difference techniques to solve for the pressure and velocity fields as well as a level set method to track the evolution of the liquid-gas interface of the droplet [27], [28], [29]. Our simulation runs in MATLAB and takes approximately ten minutes on a P4 2.66 GHz computer to simulate the splitting droplet discussed below.

Experimental data for a splitting droplet is shown in Figure 7 with snapshots from our simulation overlaid by a white dashed line. The electrode voltages are turned on at time *t = 0* and are, from left to right, 25V, 0V, and 25V. This causes the pressure on the left and right sides of the droplet to decrease, which causes the fluid from the top and bottom to come together and move out towards the left and right. Therefore, the droplet is pulled apart from the sides and pinched together from the top and bottom. The final equilibrium state consists of two satellite drops resting on the active electrodes. As can be seen, the simulation results are in agreement with the experiment. This agreement required that we include all of the loss mechanisms due to contact angle saturation, line pinning, and hysteresis [11]. We have also run simulations for bulk droplet motion, which match the general shape and speed of droplet motion observed in the experiments.



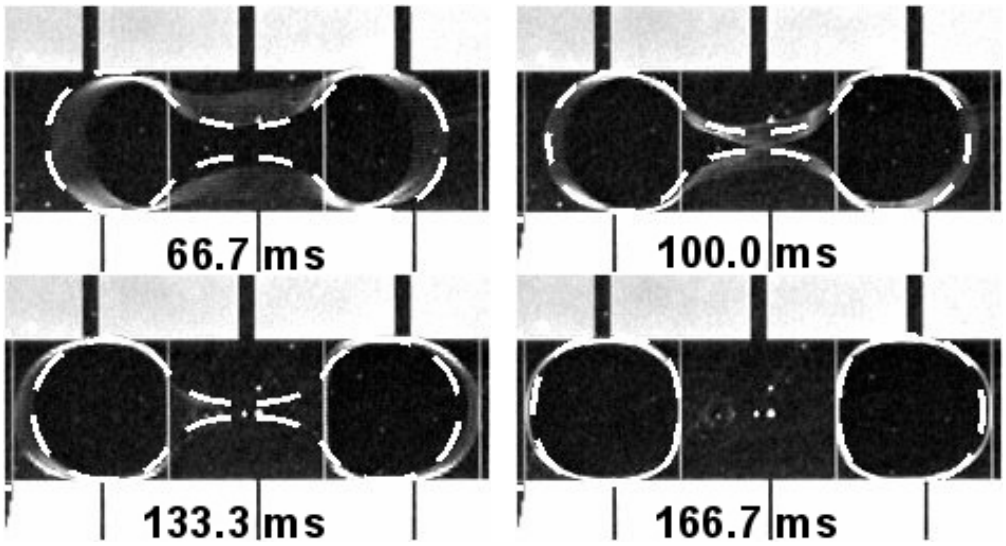


Figure 7: Simulation versus experiment. The voltages, from left to right, are 25V, 0V, and 25V. Each electrode is approximately square with a side length of 1.4 millimeters. The actuation causes the droplet to be pulled apart and pinched in the middle. Eventually, the droplet separates into two satellite drops that come to rest on the active electrodes. (Experimental data courtesy of CJ Kim at UCLA, figure repeated from [11]. The motion blur seen above is caused by the 30Hz frame rate of the camera that was used.)

4. Particle Steering Control Algorithm

Steering of multiple particles inside EWOD driven droplets, using actuators already available in the EWOD devices, requires controlled actuation of the electrode voltages. The voltages directly influence, through the boundary conditions, the pressure gradient field inside the droplet (see Figure 6 and equation (3)). Hence, by manipulating the voltages, we can control the fluid flow fields, and thereby control the velocities and positions of the particles (see equations (4) and (6)). Therefore, the control problem is to find an electrode voltage sequence that creates a temporally and spatially varying flow field that will carry all the particles along their desired trajectories.

The control problem described above is a trajectory-tracking problem: we seek to find the control inputs that will cause the system (in this case the particle positions) to follow a desired trajectory. A naïve inspection of the equations of motion, especially equation (6) for the particle dynamics, would suggest that the control problem is standard in linear control theory and that a linear quadratic regulator (LQR) tracking controller [30] could be used. However, the particle motion depends on the droplet shape and on the number of electrodes that the droplet overlays at any given moment. This information is not known a priori, which means that an LQR cannot be used. For this reason, we do local estimation and control at each time step of our simulation using a least squares framework to compute the necessary pressure boundary conditions, and then computing the electrode voltages that will achieve these boundary pressures (see Figure 6 and Figure 9). Any particle deviation from the desired trajectory that may arise from thermal fluctuations, external disturbances, and actuation errors is corrected using feedback of the particle's position. Figure 8 gives a diagram of the closed loop feedback architecture.

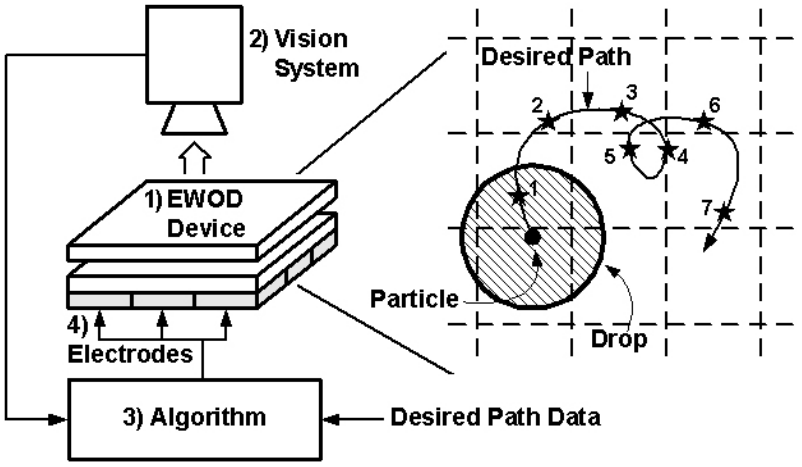


Figure 8: Particle steering closed loop feedback control architecture. 1) The EWOD device is observed by 2) an image system (a microscope/camera or an on-chip contact imager) which transmits information to 3) a computer or chip that contains 3a) an image processing algorithm to identify droplet shapes and the location of the particles and 3b) a control algorithm that computes the actuator voltages that will move the particles from where they are to where they should be, and 4) these actuation voltages are then applied on the EWOD device. The loop repeats at each time step to steer the particles along their desired trajectories. The zoomed top view of the EWOD device shows a single droplet with one particle floating inside. The curvy line indicates the desired path of the particle. In our control algorithm, we sample the trajectory by many points (only seven points are shown here; see the numbered stars 1-7)).

Our single particle steering algorithm proceeds as follows:

1. Initialization: represent trajectory as a set of points connected by straight lines.
2. Find particle position and the location of the droplet boundary.
3. Find the closest trajectory point to the particle. Set the particle's desired direction of motion to be towards a nearby trajectory point (see Section 4.3).
4. Solve a least squares problem for the necessary voltage actuation to induce a pressure gradient field that will move the particle along the desired direction in step 3.
5. Apply control voltages, solve for the resulting pressure and fluid velocity, and update the position of all the particles. Advance to the next time step of the simulation. Go back to step 2 and iterate.

The algorithm details are described below.

4.1. Algorithm Initialization

We represent the desired trajectory curves for each particle as a fine sampling of points connected by straight lines. The points are indexed in the order in which the particles should follow them (i.e. the trajectory is parameterized; see Figure 8). Complicated trajectories are broken up into separate segments for ease of particle tracking (see Section 4.3). For simplicity, only one particle and trajectory is considered in the following sections. Multiple particle steering is discussed in Section 4.5.

4.2. Particle Position and Droplet Boundary Sensing

We need to know the shape and position of the droplet as well as the position of each particle in order to apply our control algorithm. At the beginning of each time step, we obtain the position of the particle and the location of the droplet boundary using feedback through a vision system (see Figure 8). The issues of integrating a vision system with an EWOD device are not considered here. For the purposes of this paper, the particle positions and droplet shape information are taken directly from the simulation.

4.3. Compute the Desired Direction of Particle Motion

Next, the desired direction of motion for the particle is chosen to be a unit vector that points from the particle's current coordinates towards one of the trajectory points. Since maximum forcing of the pressure gradient is used to drive the particle in the desired direction (see Figure 10 in Section 4.4), it is necessary to choose a trajectory point that is just out of reach of the particle for the current time step. Otherwise, it is possible that the particle could overshoot trajectory points and trace out an unwanted zigzag path around the trajectory.

Hence, we find the target trajectory point by first finding the closest trajectory point to the particle. Then, using the trajectory parameterization (i.e. the index list; see Figure 8), we look ahead after the closest point and choose the target to be the first trajectory point that is at least one grid spacing away. This ensures the particle will move forward along the trajectory. It also guarantees that the target point is out of reach because the time steps of our simulation are chosen by the CFL criterion [31], which says that no particle can move more than a grid step at each time step. If the closest trajectory point is the last point of the trajectory, then the particle aims for the last point.

For a self-intersecting or extremely curvy trajectory, it is possible that the particle could become stuck in a loop and not travel the entire trajectory. We resolve this issue by breaking the trajectory into smooth segments that do not intersect and only allow the particle to see one segment at a time. As a result, the particle follows one piece of the trajectory until it reaches the end, where our algorithm switches to the next segment. Therefore, without loss of generality, we assume in the following subsections that the trajectory consists of just one segment.

The forcing of the particle is created by the pressure gradient. And the desired unit vector discussed above determines the direction of forcing. This unit vector is used in the next section to calculate the pressure boundary conditions needed to realize the pressure gradient that will move the particle in the desired direction.

4.4. Least Squares Solution of the Required Pressure Boundary Conditions

Figure 9 shows a top view of a sample droplet in the EWOD device containing a single particle. The current drop shape overlaps four electrodes hence four actuators are available to move the single particle. In each of the four cases, only one electrode is on; the rest are off. The arrows inside the droplet show the fluid flow for each of the four voltage actuations. The black dot

represents the particle with a thick arrow indicating the negative direction of the pressure gradient at the particle location (note that the fluid flows opposite to the pressure gradient).

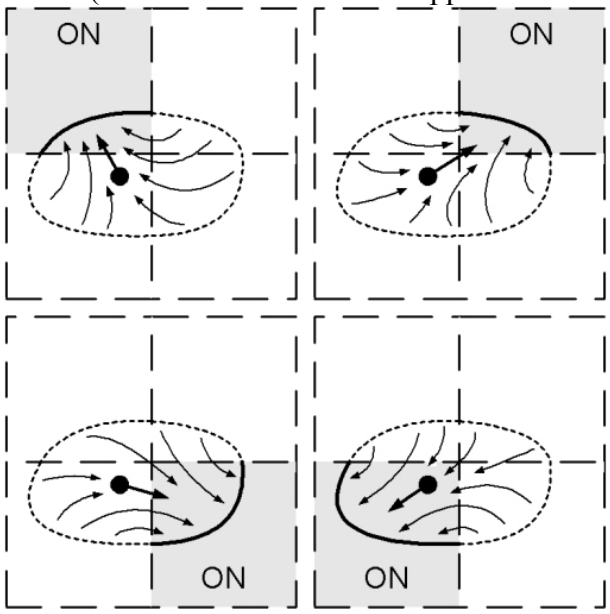


Figure 9: Linear combination of pressure gradients for a single droplet overlaying four electrodes (small dashed squares). The diagram above shows a droplet in an EWOD system with four different instances of voltage actuation. In each instance, only one of the four electrodes is on. The particle floating inside the droplet (black dot) has a thick arrow indicating its direction of motion for each single electrode actuation. These arrows actually represent the opposite direction of the pressure gradient when a unit pressure boundary condition is set on the thick curve that overlays the shaded electrode, with zero pressure boundary conditions everywhere else. The thin curvy arrows show the fluid flow inside the droplet. Since the pressure equation (2) is linear, we can make the particle move in any desired direction by taking an appropriate linear combination of the four possible boundary conditions given above.

Our algorithm centers on the idea of taking the right linear combination of pressure gradients in Figure 9 to make the particle (or particles) move in the direction(s) we want at a particular time step. This will directly correspond to finding the right combination of electrode voltages at every time step to realize the desired particle motion (or motions).

First, given the current droplet configuration, we solve equation (2) for the pressure field inside the droplet for a single active electrode. The pressure boundary conditions are defined to be one on the droplet boundary that lies over the active electrode and zero everywhere else (see Figure 9). From the pressure solution, the pressure gradient at each particle's position is computed. After repeating this for each electrode, we obtain a matrix of pressure gradients

$$(7) \qquad G = -\begin{bmatrix} \nabla P_1(x_1, y_1) & \cdots & \nabla P_N(x_1, y_1) \\ \vdots & \ddots & \vdots \\ \nabla P_1(x_m, y_m) & \cdots & \nabla P_N(x_m, y_m) \end{bmatrix},$$

where (x_j, y_j) are the coordinates for the jth particle. Each column of pressure gradients $\nabla P_k(x_j, y_j)$ in the matrix corresponds to a single active electrode; each row to a single particle.

The total number of particles is m and the number of available electrodes is N. The minus sign accounts for the direction of particle motion.

Next, given the desired pressure gradient at each particle's location in the droplet, we wish to find the appropriate boundary conditions to realize it. Since Laplace's equation for the pressure (2) is linear regardless of the droplet shape, solutions for single active electrodes can be combined linearly to obtain the pressure gradient field due to many active electrodes. This reduces our problem to solving a linear system

$$(8) \qquad G\alpha = b\,,\ \alpha = \begin{bmatrix} \alpha_1 \\ \vdots \\ \alpha_N \end{bmatrix},\ b = \begin{bmatrix} \nabla P_D(x_1, y_1) \\ \vdots \\ \nabla P_D(x_m, y_m) \end{bmatrix},$$

where $\nabla P_D(x_j, y_j)$ is a 2x1 vector representing the desired pressure gradient at the jth particle and α is the vector of boundary values that will achieve *b*. We set $\nabla P_D(x_j, y_j)$ equal to the unit vector in section 4.3 that represents the desired direction of motion for the jth particle. If $2m \geq N$, the number of particle degrees of freedom is greater than the available actuators and (in general) (8) cannot be solved exactly. A least squares solution is needed to obtain the best fit of actuations α. Otherwise, it is a pseudo-inverse problem, which has a solution as long as the matrix G has full row rank [32].

We solve equation (8) for α using the singular value decomposition (SVD) [32]. In addition, each component of the solution vector must be made to satisfy an inequality constraint

$$(9) \qquad \alpha_{\min} \leq \alpha_j \leq \alpha_{\max}\,, \quad 1 \leq j \leq N$$

where $\alpha_{\min}$ and $\alpha_{\max}$ are the minimum and maximum values that the pressure boundary condition can be for any electrode. These constraints come from the limitations of varying the contact angle (i.e. contact angle saturation). Hence, $\alpha_{\min}$ and $\alpha_{\max}$ are related to the maximum and minimum contact angles achievable in the EWOD device. In order to satisfy equation (9), we take the solution α to equation (8) and transform each of its components so that the full dynamic range of boundary forcing is utilized (see Figure 10).

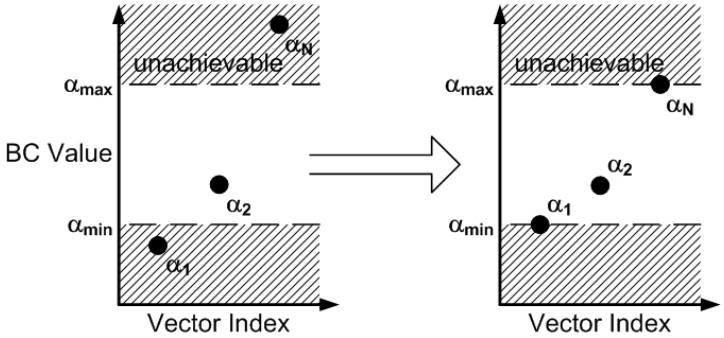


Figure 10: Linear transformation of boundary conditions. An example of satisfying the boundary condition constraints is shown above. On the left, the components of the solution to equation (8) are plotted with the maximum and minimum constraint bounds denoted by dashed lines (see equation (9)). On the right, the components have been linearly mapped to enforce the constraints. This introduces a scaling factor into equation (8), which affects the magnitude of the pressure gradient b vector (i.e. the magnitude of the force

acting on the particles). In effect, this causes the particle to be forced as much as possible in the desired direction.

With this new transformed α, we know what the pressure boundary values should be to realize the desired pressure gradient field. But it is not possible to exactly enforce α because we cannot directly control the planar curvature term κ_{xy} in equation (3). For a circular droplet, the planar curvature term is constant and has no effect on the pressure gradient field [33]; hence, it can be ignored. Using equation (3), it is straightforward to compute the contact angles needed to implement α. For non-circular droplets, we use the same procedure. It is not reasonable to use the planar curvature term in our control algorithm because it involves 2nd derivatives of data that cannot be accurately measured in experiments [34]. Instead, we view it as a small error to the desired directional forcing of the particles. This error grows as the droplet deviates from being a circle. This is not a problem for particle steering for two reasons. First, the linear transformation of the boundary conditions in Figure 10 ensures maximum forcing of the particle. Thus, the relative magnitude of the error due to the xy planar curvature is minimized. Second, any particle trajectory tracking errors that may occur are corrected through our feedback system (see numerical simulations in section 5.1). However, the planar curvature does limit the type of trajectories that the particles can follow and this is discussed in Section 5.2.

Finally, the electrode voltages needed to actuate the contact angles corresponding to the pressure boundary condition vector α are computed using the inverse contact angle saturation/line pinning curve given in Figure 6.

4.5. Apply Voltages, Update Particle Position, Advance to the Next Time Step

Our simulation advances to the next time step after using the voltages computed in section 4.4 to solve for the induced pressure and velocity fields. The velocity field is then used to update the position of the particle (see Figure 11). The scaling described in Figure 10 ensures the particle will be forced as fast as possible along the desired direction. Our algorithm runs by going back to section 4.2 and repeating for each time step.

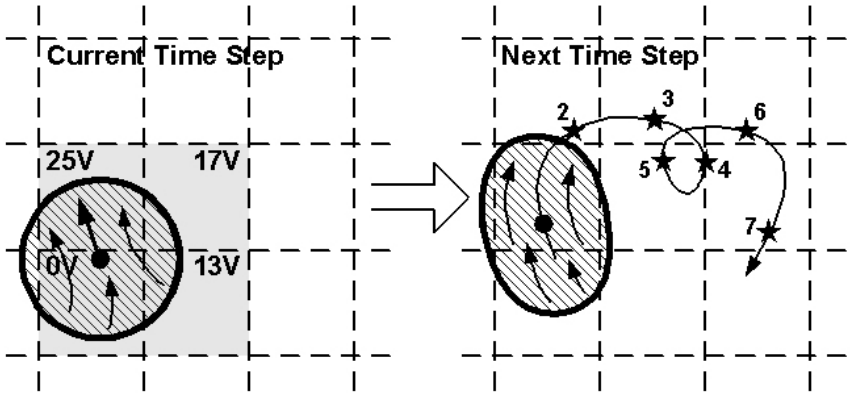


Figure 11: EWOD particle steering control algorithm update. The droplet configuration from Figure 8 is shown in the diagram above. The direction of motion for the particle is towards the trajectory point that is just out of reach for the current time step. This control strategy ensures the particle will move as fast as

possible and stays close to its desired trajectory. On the left, the shaded electrodes contain the voltages needed to move the particle in the desired direction. These are computed by the least squares solution in section 4.4 and by the voltage versus contact angle curve in Figure 6. The varying voltage grid induces a pressure gradient field inside the droplet such that the pressure gradient at the particle is pointing along the desired direction of motion. This moves the droplet and particle along the trajectory to the next time step.

Multiple particle steering is easily handled by applying the above discussion to each particle and its respective trajectory. The only change is that the linear system in section 4.4 has more rows to accommodate the extra particles. If the number of electrodes is limited, then this can adversely affect the controllability we have. A single particle can be made to track interesting trajectories with enough electrodes (see Figure 12 and Figure 13). Two particles can be controlled for simple trajectories as shown in Figure 14. For more than two particles, all but the simplest trajectories (i.e. straight lines) cannot be tracked. Section 5.2 discusses the merits and limitations of our method.

5. Simulation Results and Discussion

In this section, we present some results that demonstrate basic particle steering control using our experimentally validated simulation. A 3x3 electrode grid is used to actuate and control the droplet where each square electrode is 1.4 mm on a side. We present four cases that are controllable and three cases that are uncontrollable and then discuss the possibilities and limits of our method. The voltages generated by our algorithm are reasonable and are within the limits of the UCLA device discussed in [12].

5.1. Controllable Cases

Figure 12 shows a droplet moving in a way that makes a particle floating inside follow a figure ‘8’ path. A circular droplet starts on the center electrode with a particle resting in the center of the droplet. The blue dashed curve represents the desired trajectory, which is made up of a fine sampling of points. Two segments are used to represent the trajectory because of the self-intersection (see section 4.3). The voltages on the electrode grid are actuated using the algorithm in Section 4, which causes the particle to move forward along the trajectory. For this case, the droplet always overlaps enough electrodes to allow it to be controlled in a way that keeps the particle moving on the figure ‘8’ path. The particle never deviates more than 20 micrometers from the desired trajectory.

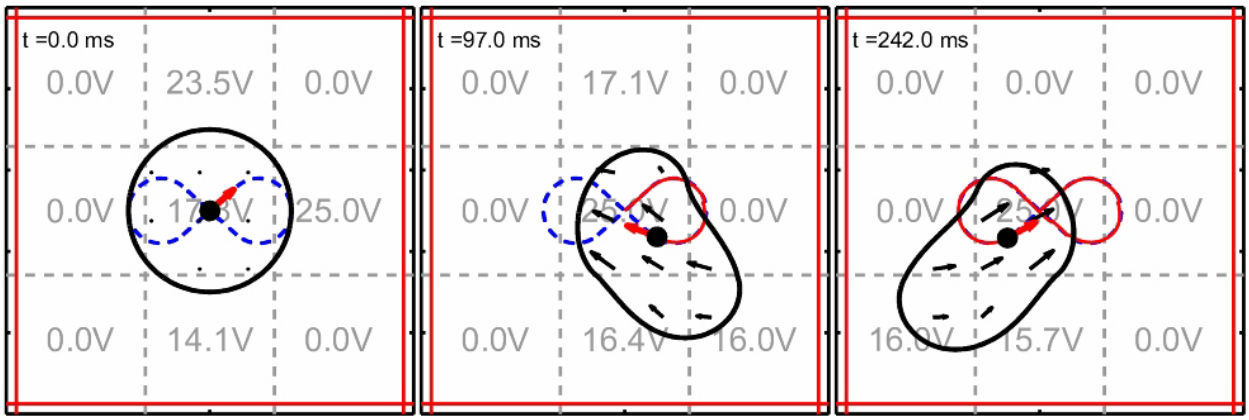


Figure 12: Particle following a figure '8' path. In the simulation results above, we have a droplet (denoted by the thick black curve) lying on a 3x3 grid of electrodes (denoted by the dashed lines). The blue dashed curve is the desired figure '8' path and a black dot represents the particle with a thick red arrow pointing in the desired direction of travel. The red curve is the actual path of the particle. The black arrows inside the droplet denote the fluid velocity field inside the droplet. The voltages on the grid are time varying in such a way as to keep the particle moving along the path and are computed using the method of section 4, equations (7)-(9).

In Figure 13, a particle is shown following an angular path that is represented by five separate straight-line segments. This is to prevent the particle from rounding off the corners as it travels along the trajectory. Just as in Figure 12, the droplet always overlaps enough electrodes to keep the particle on the path, with a maximum deviation error of 25 micrometers.

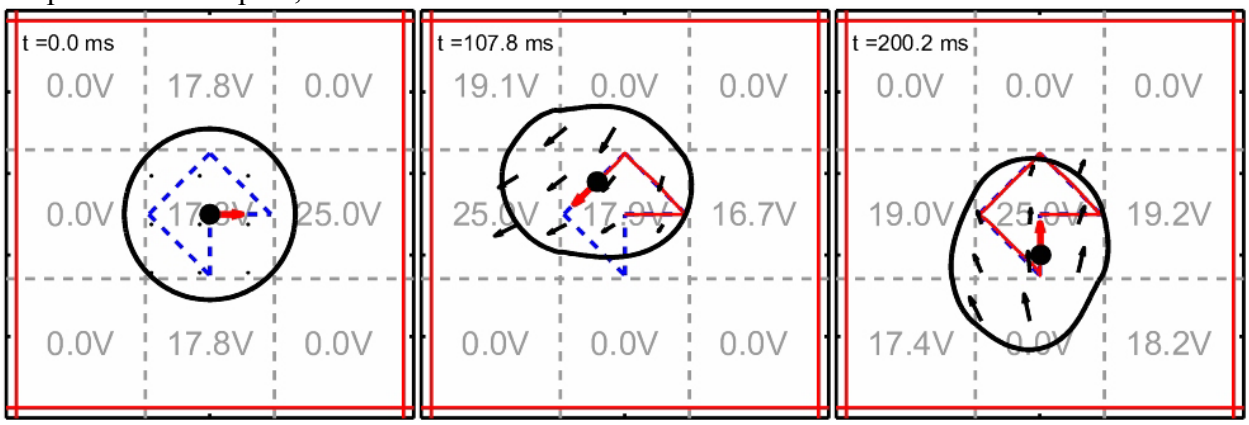


Figure 13: Particle following an angular path (same format as in Figure 12). The particle is able to track the trajectory very well, especially at the corners.

An example of two-particle control is shown in Figure 14. One particle is held stationary while the other moves along a circular arc. The trajectory for the stationary particle consists of a single point, which ensures that it stays close to that point. As the particle on the right follows the circular arc trajectory, the stationary particle oscillates around its desired position to within 10 micrometers. The droplet itself becomes deformed because of the limited actuators and the restrictive task of moving one particle and holding another still. This also prevents the particle on the circular arc from moving past the point shown in the last frame of Figure 14 and completing the circle.

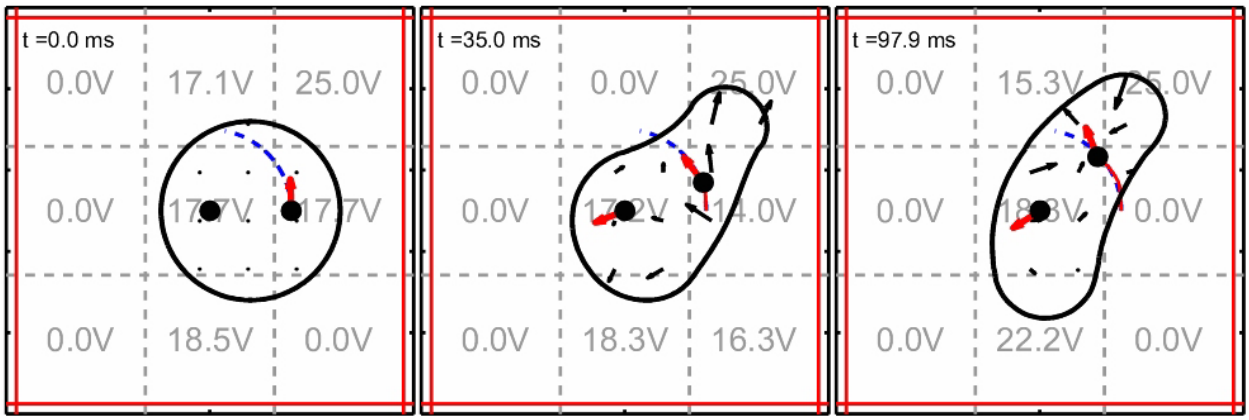
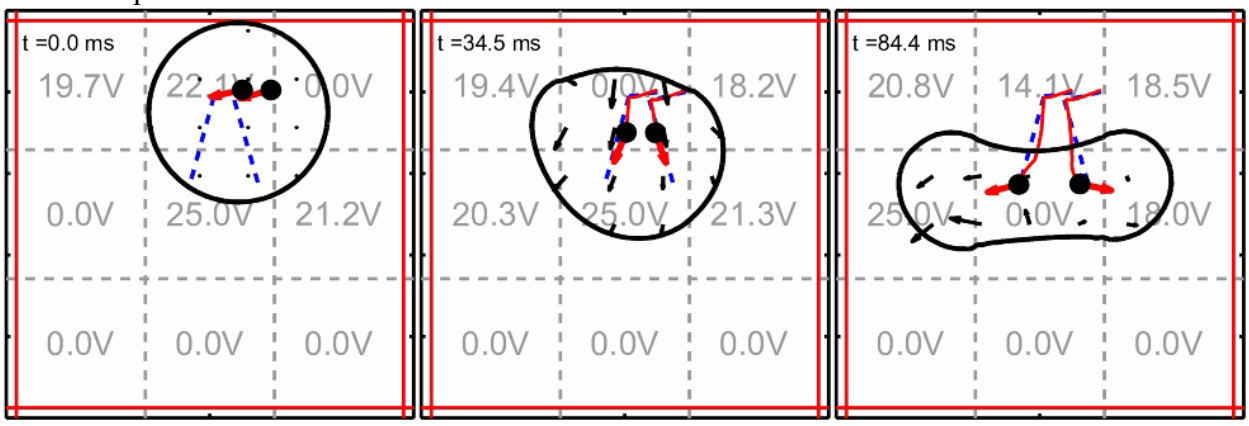


Figure 14: Two-particle control: one particle moves on a quarter circle, the other is stationary (same format as in Figure 12). The stationary particle's trajectory is a single point. As the particle on the right follows the circular arc, the droplet distorts to accommodate both particle motions.

In Figure 15, we demonstrate particle separation. A droplet starts in the first panel with two particles spaced 0.31 millimeters apart. Both particles follow separate diverging trajectories designed to stretch the droplet and separate the particles. Once the particles near the ends of their trajectories (see the third frame), our control algorithm turns off and we command an open loop voltage of 25V on the middle left and right electrodes and zero volts everywhere else. This causes the droplet to split into two smaller drops, each of which contains a single particle. The reason for not using our control algorithm to complete the split is because of numerical instability. When both particles are in the lobes of the dumbbell shape of the pinching droplet, the available forcing at the particles' positions is fairly weak. This would cause the condition number of the G matrix in equation (8) to degenerate and produce errors in the least squares solution. Therefore, we avoid this by commanding open loop voltages that we know will split the droplet (see Figure 7). Also, see Figure 17 for an example of how this numerical instability can affect particle control.



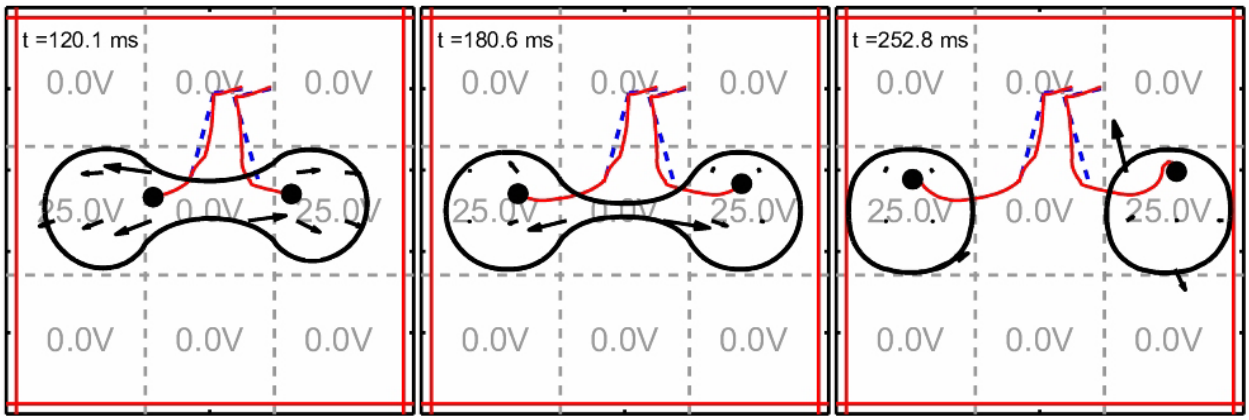


Figure 15: Two-particle separation into two satellite drops (same format as in Figure 12). Each particle first follows a trajectory that takes them away from each other. When there is sufficient distance between the two particles, our control algorithm turns off and the separation is completed in the usual way by applying the open loop voltages used in the experimental splitting example of Section 3 (see Figure 7).

5.2. Uncontrollable Cases

We now present some cases that cannot be achieved. In Figure 16, a particle is shown traveling along a sine wave path. The particle is able to track the trajectory very well until near the end where there is a kink in the particle's path. The loss of tracking is because the droplet's shape and position is such that the number of available electrodes is very limited. It becomes impossible to create a pressure gradient field that will continue moving the particle in the tangential direction of the desired trajectory. Hence, the particle drifts away from the trajectory by more than 100 micrometers. This situation corresponds to equation (8) having no exact solution, which means only a least squares best fit of the desired pressure gradient can be computed. Eventually, however, the particle is able to reacquire the trajectory.

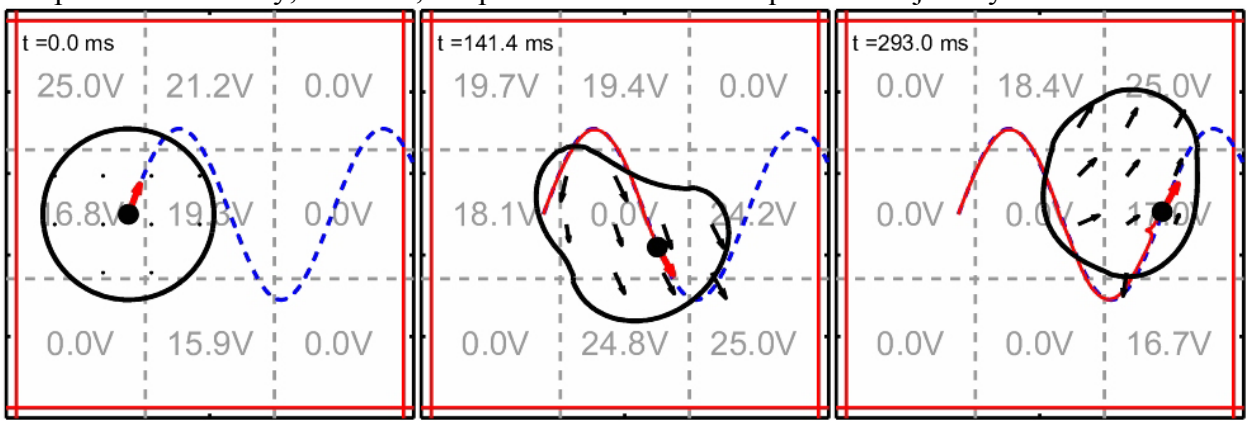


Figure 16: Particle traveling on a sine wave (same format as in Figure 12). The particle is able to track the sine wave path until the last time frame where the particle drifts away from the desired trajectory (see the kink in the particle's path).

Figure 17 shows two, initially separate, particles trying to come together and touch. The desired motion of the particles induces the droplet to try and pinch together in an effort to have the

particles touch. However, when the particles begin to near each other, the droplet ceases its splitting action. Instead, the droplet holds the necking region and begins to oscillate up and down. This is because we are trying to specify two opposite directions of motion at points that are very close together, which leads to a numerical instability in solving equation (8). As the particle positions get closer together, the condition number of the matrix G degenerates causing spurious oscillations in the control voltages. The droplet is unable to bring the particles together, much less pinch, because of the randomly varying electrode voltages.

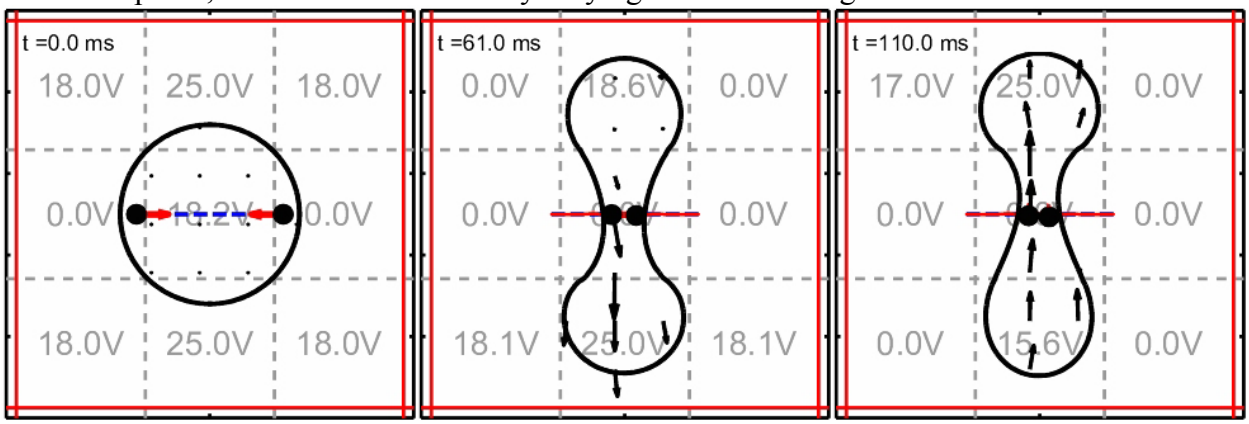


Figure 17: Two particles trying to come together and pinch a droplet (same format as in Figure 12). The particles travel on two separate trajectories that would, ideally, bring them together. However, as they come together, numerical instabilities in equation (8) cause random variations in the control voltages. This causes the droplet to hold its shape and move up and down in an undesirable way.

Figure 18 shows two particles trying to follow diverging paths. At first the droplet is able to deform enough to keep the two particles on their respective trajectories. But this quickly fails. The droplet is unable to continue deforming in a way that keeps both particles on track and moving forward. Since the trajectories are just straight lines represented by two points each, the control algorithm keeps the particles moving forward while trying to force them towards the endpoints of the trajectories. The end result is both particles stay roughly parallel with each other and are unable to recover their trajectories. This stems from a lack of available electrodes and the limitations imposed by contact angle saturation.

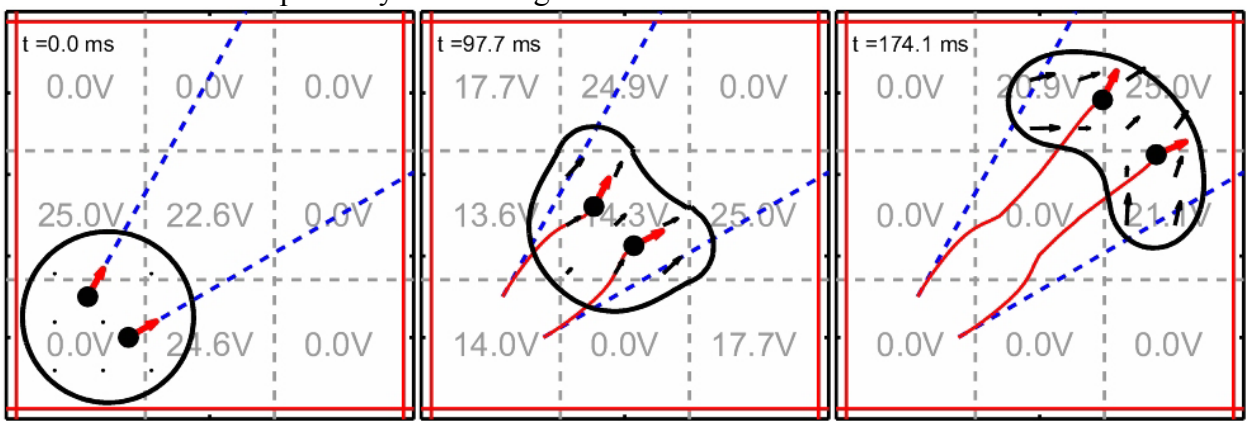


Figure 18: Two particles on diverging paths (same format as in Figure 12). Each particle is attempting to follow separate trajectories, both of which lead away from each other. Due to limitations of the pressure

boundary actuation, and a lack of electrodes, the control algorithm is unable to keep both particles moving on their respective paths.

The limitations of achievable particle control arise from having a small number of electrodes available for actuation and from contact angle saturation. Moving several particles in different directions requires many degrees of freedom in adjusting the pressure boundary conditions. As the droplet moves, it must overlap enough electrodes to allow the realization of the pressure gradient field needed to push the individual particles along their trajectories. Hence, a finer electrode grid would allow more precise control of more particles simultaneously. Also, some trajectories will require the droplet to become extremely distorted and may require it to split into several pieces. To do this, one needs enough dynamic range in the boundary forcing to overcome the droplets natural tendency to remain in a circular shape (see the xy planar curvature term in equation (3)). Contact angle saturation limits the boundary forcing and the degree of droplet deformation, which can cause controllability to be lost and particles to drift off of their desired trajectories (see Figure 16 and Figure 18). In addition, if two particles are very close together, it is not possible to force them in arbitrary directions. The limits of boundary forcing and the numerical instability that enters into solving equation (8) inhibit close particle control no matter how many actuators are present (see Figure 17).

As of today, it is only feasible to fabricate devices with a few actuators that can control one or two particles. It is interesting that existing EWOD systems have enough control authority to steer a single particle along complex trajectories and to steer two particles along simple paths.

6. Conclusion

A control method has been developed to steer individual particles inside electrowetting systems. Using only existing EWOD electrode actuators, the algorithm can steer particles to specific locations, hold one particle stationary while another particle is moved along a path, and steer and sort particles within and across individual droplets. The particle steering is achieved by creating fluid flows that carry all the particles from where they are to where they should be at each time step. It is possible to steer a single particle along interesting trajectories with a small (3x3) number of electrodes. Steering two particles independently inside a single drop is also possible for simple particle motions. A finer grid of electrodes, or the use of larger liquid drops, would allow simultaneous steering of more particles along more complicated trajectories inside a single drop of liquid. The particle steering results outlined in this paper are based on a model of electrowetting fluid dynamics that has been validated against experiments. An experimental demonstration of particle steering in the UCLA devices will require the implementation of a real time vision system, a real time control algorithm realization, along with image sensing, control computation, and EWOD device system integration: these items have been demonstrated in our electroosmotically driven systems and will be reported in the context of electrowetting devices in future publications.

7. References

1. Lee, J., et al., *Electrowetting and electrowetting-on-dielectric for microscale liquid handling.* Sensor Actuat. A-Phys, 2002. **95**: p. 269.
2. Kim, C.J. *Micropumping by Electrowetting*. in *Int. Mechanical Engineering Congress and Exposition, New York, NY, IMECE2001/HTD-24200*. 2001.
3. Srinivasan, V., V. Pamula, M. Pollack, and R. Fair. *A digital microfluidic biosensor for multianalyte detection*. in *Proceedings of the IEEE 16th Annual International Conference on Micro Electro Mechanical Systems*. 2003.
4. Paik, P., V. K. Pamula, and R. B. Fair, *Rapid droplet mixers for digital microfluidic systems.* Lab on a Chip, 2003. **3**: p. 253-259.
5. Pollack, M.G., R.B. Fair, and A.D. Shenderov, *Electrowetting-based actuation of liquid droplets for microfluidic applications.* Applied Physics Letters, 2000. **77**(11).
6. Cho, S.K., H. Moon, J. Fowler, S.-K. Fan, and C.-J. Kim. *Splitting a Liquid Droplet for Electrowetting-Based Microfluidics*. in *Int. Mechanical Engineering Congress and Exposition, New York, NY, IMECE2001/MEMS-23831*. 2001.
7. Fowler, J., H. Moon, and C.-J. Kim, *Enhancement of Mixing by Droplet-Based Microfluidics.* IEEE Conf. MEMS, Las Vegas, NV, 2002: p. 97-100.
8. Ashkin, A., *History of Optical Trapping and Manipulation of Small-Neutral Particles, Atoms, and Molecules.* IEEE Journal on Selected Topics in Quantum Electronics, 2000. **6**(6): p. 841-856.
9. Hoogenboom, J.P., et al., *Patterning surfaces with colloidal particles using optical tweezers.* Applied Physics Letters, 2002. **80**(25): p. 4828-4830.
10. Curtis, J.E., B.A. Koss, and D.G. Grier, *Dynamic holographic optical tweezers.* Optics Communications, 2002. **207**(1-6): p. 169-75.
11. Walker, S. and B. Shapiro, *Modeling the Fluid Dynamics of Electro-Wetting on Dielectric (EWOD).* Journal of Micro-Electro-Mechanical Systems, 2004: p. submitted.
12. Cho, S.K., H. Moon, and C.-J. Kim, *Creating, Transporting, Cutting, and Merging Liquid Droplets by Electrowetting-Based Actuation for Digital Microfluidic Circuits.* Journal of Microelectromechanical Systems, 2003. **VOL. 12**(NO. 1): p. 70-80.
13. Armani, M., et al. *Using Feedback Control and Micro-Fluidics to Steer Individual Particles*. in *18th IEEE International Conference on Micro Electro Mechanical Systems*. 2005. Miami, Florida.
14. Armani, M., S. Chaudhary, R. Probst, and B. Shapiro, *Using Feedback Control and Micro-Fluidics to Steer Individual Particles.* Journal of Micro-Electro-Mechanical Systems: p. submitted.
15. Fossum, E.R., *CMOS Image Sensors: Electronic Camera-On-A-Chip.* IEEE Trans. On Electron Devices, 1997. **44**(10): p. 1689-1698.
16. Berge, B., *Electrocapillarity and wetting of insulator films by water.* Comptes Rendus de l Academie des Sciences Series II, 1993. **317**(2): p. 157-163.
17. Vallet, M., B. Berge, and L. Vovelle, *Electrowetting of water and aqueous solutions on poly(ethyleneterephthalate) insulating films.* The European Physical Journal, 1996. **11**(B): p. 583-591.
18. Lippmann, G., *Relation entre les phenomenes electriques et capillaires.* Ann. Chim. Phys., 1875. **5**: p. 494-549.

19. Verheijen, H.J.J. and W.J. Prins, *Reversible electrowetting and trapping of charge: Model and experiments.* Langmuir, 1999. **15**: p. 6616-6620.
20. Darhuber, A.A., and S.M. Troian, *Principles of microfluidic actuation by modulation of surface stresses.* Annu. Rev. Fluid Mech., 2005. **37**: p. 425-455.
21. Zhang, T., K. Chakrabarty, R. B. Fair, and S. E. Lyshevsky, *Microelectrofluidic Systems: Modeling and Simulation*. 2002, Boca Raton, FL: CRC Press. 288 pages.
22. Blake, T.D., A. Clarke, and E. H. Stattersfield, *An investigation of electrostatic assist in dynamic wetting.* Langmuir, 2000. **16**(6): p. 2928–2935.
23. Shapiro, B., et al., *Equilibrium Behavior of Sessile Drops under Surface Tension, Applied External Fields, and Material Variations.* Journal of Applied Physics, 2003. **93**(9).
24. Panton, R.L., *Incompressible Flow*. 2 ed. 1996, New York, NY: John Wiley & Sons, Inc.
25. Probstein, R.F., *Physicochemical Hydrodynamics: An Introduction*. 2 ed. 1994, New York: John Wiley and Sons, Inc.
26. Batchelor, G.K., *An Introduction to Fluid Dynamics.* 1967: Cambridge University Press.
27. Sethian, S.A., *Level Set Methods & Fast Marching Methods*. 2 ed. 1999: Cambridge University Press.
28. Osher, S. and R. Fedkiw, *Level Set Methods and Dynamic Implicit Surfaces.* 2003: Springer-Verlag New York.
29. Caiden, R., R. Fedkiw, and C. Anderson, *A Numerical Method for Two Phase Flow Consisting of Separate Compressible and Incompressible Regions.* J. Comput. Phys., 2001. **166**: p. 1-27.
30. Lewis, F.L., and V. L. Syrmos, *Optimal Control*. 2nd ed. 1995, New York, NY: Wiley-Interscience. 560 pages.
31. Morton, K.W., and D. F. Mayers, *Numerical Solution of Partial Differential Equations*. 1994: Cambridge University Press. 239 pages.
32. Strang, G., *Linear Algebra and Its Applications*. 3 ed. 1988, New York, NY: Brooks Cole. 520 pages.
33. Zachmanoglou, E.C. and D.W. Thoe, *Introduction to Partial Differential Equations with Applications*. 1986, New York, NY: Dover Publications, Inc.
34. Ralston, A., and P. Rabinowitz, *A First Course in Numerical Analysis*. 2nd ed. 2001, Mineola, NY: Dover.