UNDER REVIEW

RESCIENCE C

Replication / Microfluidics

[Re] Predictive model for the size of bubbles and droplets created in microfluidic T-junctions

Kenneth E. Schackart III[1, ID] and Kattika Kaarj[2, ID]

[1]Department of Biosystems Engineering, The University of Arizona, Tucson, Arizona, USA – [2]School of Biology, Institute of Science, Suranaree Institute of Technology, Nakhon Ratchasima, 30000, Thailand

Edited by
(Editor)

Received
01 November 2018

Published
–

DOI
–

Microfluidic T-junctions are used for creating dispersed bubbles or droplets of a consistent volume. Such platforms have been used for a variety of applications, such as chemical microreactors, biological screening assays, and DNA analysis. It is often critical that the volume of the bubbles or droplets are consistent and predictable. A mathematical model was previously developed by van Steijn *et al.,* for predicting the volume of bubbles and droplets formed in microfluidic T-junctions, however, their code was not made readily available. In this work, we develop Python modules that implement the mathematical model and replicate the figures in the original work. The model could be fully replicated based solely on the contents of the original article. However, one figure provided in the original work was inconsistent with the model equations while another could not be fully replicated.

1 Introduction

1.1 Microfluidic T-junctions

Microfluidic T-junctions are microfluidic devices composed of T-shaped junctions where cross-flowing streams occur. Droplet formation in microfluidic T-junction occurs at the junction where liquid or gas from the side channel (dispersed phase) meets with liquid in the main channel (continuous phase). The dispersed phase forms a plug shape due to the shear and extrusion of the continuous phase and rupture due to the instability of interface[1]. The volume of the resulting droplets heavily relies on the channel geometry, flow rate ratio, and device surface properties[2]. Microfluidic T-junctions have been applied in biological, chemical, and biomedical fields[3]. For example, chemical microreactors that allow very small-scale chemical reactions, high throughput biological screening assays, and DNA analysis[4]. The ability to precisely predict the size of bubbles and droplet generated in microfluidic T-junctions is very important because many of these applications require precise sample volume as determined by droplet size.

1.2 Models for size prediction

Since several parameters can affect droplet formation in microfluidic T-junctions, it can be difficult to precisely tune the droplet size *a priori*. Many models have been reported to solve this issue. For example, a numerical simulation approach was proposed, utilizing a hydrodynamic model to predict droplet size by varying capillary numbers, flow rate ratios, viscosity ratios, and contact angles[5]. In a study by Steegmans *et al.,*, a statistical model based on the data from experiments in other published works was developed

Correspondence should be addressed to Kenneth E. Schackart III (schackartk1@gmail.com)
The authors have declared that no competing interests exist.
Code is available at https://github.com/schackartk/t-junction_model_reporoduction/tree/master..

to model droplet break-up in T-junctions. Their approach divided the process into two phases: (1) droplet growth, and (2) detachment phase[6]. Another study concluded that the effects of surface force and viscous force dominate the droplet stream generated in their microfluidic T-junction system based on numerical results, theoretical analysis, and experimental measurements[7]. A volume-of-fluid model was introduced to investigate the droplet formation process by varying the dispersed phase flow rate under a constant capillary number[8].

In this work, we aim to replicate the mathematical model based on geometrical parameters to predict the size of bubbles and droplets created in microfluidic T-junctions published by van Steijn *et al.,*[9] which was consistent with the model presented by Steegmans *et al.,*.

1.3 van Steijn et al., Model overview

The model presented in the original work by van Steijn *et al.,* divides the droplet formation process into two phases: *filling* and *squeezing*. The final model adds the contributions from the two stages to arrive at the droplet/bubble volume. This is summarized as the dimensionless volume in the following equation:

$$\frac{V}{hw^2} = \frac{V_{fill}}{hw^2} + \alpha\frac{q_d}{q_c} \tag{1}$$

Filling phase — The filling phase begins as the dispersed fluid enters the main channel from the inlet channel. This phase continues until the emerging dispersed fluid reaches the far side of the main channel, at which time the squeezing phase begins.

The equation describing the filling phase defines the dimensionless fill volume as:

$$\frac{V_{fill}}{hw^2} = \begin{cases} \frac{3\pi}{8} - \frac{\pi}{2}\left(1-\frac{\pi}{4}\right)\frac{h}{w} & \text{for } w_{in} \leq w \\ \left(\frac{\pi}{4} - \frac{1}{2}\arcsin\left(1-\frac{w}{w_{in}}\right)\right)\left(\frac{w_{in}}{w}\right)^2 + & \\ -\frac{1}{2}\left(\frac{w_{in}}{w}-1\right)\left(2\frac{w_{in}}{w}-1\right)^{\frac{1}{2}} + \frac{\pi}{8} + & \\ -\frac{1}{2}\left(1-\frac{\pi}{4}\right)\left(\left(\frac{\pi}{2} - \arcsin\left(1-\frac{w}{w_{in}}\right)\right)\frac{w_{in}}{w} + \frac{\pi}{2}\right)\frac{h}{w} & \text{for } w_{in} > w \end{cases} \tag{2}$$

This equation is derived strictly based on geometries that are present when the emerging dispersed phase reaches the far wall of the main channel.

Squeezing phase — Once the dispersed phase reaches the far wall of the main channel, the squeezing phase begins. The contribution to the total dimensionless volume during the squeezing phase is described by the following equation:

$$\frac{V_{squeeze}}{hw^2} = \alpha\frac{q_d}{q_c} \tag{3}$$

The squeezing coefficient (α) is described by Equation (4):

$$\alpha = \left(1-\frac{\pi}{4}\right)\left(1-\frac{q_{gutter}}{q_c}\right)^{-1}\left(\left(\frac{R_{pinch}}{w}\right)^2 - \left(\frac{R_{fill}}{w}\right)^2 + \frac{\pi}{4}\left(\frac{R_{pinch}}{w} - \frac{R_{fill}}{w}\right)\frac{h}{w}\right) \tag{4}$$

Where the radius denoted R_{fill} is determined by:

$$R_{fill} = \max\left(w, w_{in}\right) \tag{5}$$

And the radius denoted R_{pinch} is described by:

$$R_{pinch} = w + w_{in} - \left(\frac{hw}{h+w} - \varepsilon\right) + \left(2\left(w_{in} - \left(\frac{hw}{h+w} - \varepsilon\right)\right)\left(w - \left(\frac{hw}{h+w} - \varepsilon\right)\right)\right)^{\frac{1}{2}} \tag{6}$$

The original authors determined experimentally that $\frac{q_{gutter}}{q_c} = 0.1$, so this relationship is used during all calculations in the original work.

2 Methods

All code used for replication was written in Python[10] v3.11.0 and using the conda[11] package manager v22.9.0. Code was developed using Windows Subsystem for Linux 2 (WSL2)[12] connected to an Ubuntu[13] v20.04 kernel.

2.1 Python modules

Filling phase — While dimensionless fill volume could be calculated directly using equation (2), we took a more verbose approach in developing the filling module. This is to aid in readability and interpretability of the functions defined in the module.

We begin with the provided definition of fill volume (V_{fill}) as the gross volume (V_{gross}) minus the total gutter volume ($2V_{gutter}$).

V_{gross} is calculated as the cross-sectional area of the emerging droplet ($A_{droplet}$) at the midplane extruded by the channel height (h).

V_{gutter} is calculated as cross sectional area of the gutter (A_{gutter}) extruded along the perimeter of the emerging droplet (l_{gutter}).

This is summarized as:

$$V_{fill} = V_{gross} - 2 * V_{gutter} = hA_{droplet} - 2A_{gutter}l_{gutter} \tag{7}$$

In the module, $A_{droplet}$ is calculated piece-wise using geometry (*e.g.* area of quarter circle with radius w plus area of half circle with radius $\frac{w}{2}$). A simplified version of this is given by:

$$A_{droplet} = \begin{cases} \frac{3\pi}{8}w^2 & \text{for } w_{in} \leq w \\ \frac{\pi}{8}w^2 + \frac{\pi}{4}w_{in}^2 & \\ -w_{in}^2\frac{1}{2}\arcsin\left(\frac{w_{in}-w}{w_{in}}\right) & \\ -\frac{1}{2}\left(w_{in}-w\right)\sqrt{w_{in}^2 - \left(w_{in}-w\right)^2} & \text{for } w_{in} > w \end{cases}$$

Similarly, l_{gutter} is calculated using basic geometric shapes, and a simplified equation is given by:

$$l_{gutter} = \begin{cases} \pi w & \text{for } w_{in} \leq w \\ \frac{\pi}{2}w + w_{in}\left(\frac{\pi}{2} - \arcsin\left(1 - \frac{w}{w_{in}}\right)\right) & \text{for } w_{in} > w \end{cases}$$

The simplified equation for the cross-sectional area of the gutter is:

$$A_{gutter} = \frac{8-\pi}{16}h^2$$

The module calculates the fill volume by individually calculating the variables in equation (7) using the described relationships. This is then divided by hw^2 to make it non-dimensionalized.

To ensure that the verbose geometric implementation matches the simplified equation (2), unit tests compare the values obtained using each version.

Squeezing phase — The non-dimensionalized volume during the squeezing period is given by equation (3), with equations for the variables defined in equations (4), (5), and (6).

Functions for calculating each of the intermediate variables and non-dimensionalized squeeze volume were developed, each with accompanying unit tests.

2.2 Replication of figures

To fully replicate the results in the original work, the figures showing model outputs were replicated using the modules that were developed. A Python script imports the necessary functions from the squeezing and filling modules and uses Pandas[14] v1.5.1 and plotnine[15] v0.10.1 to create the figures. This section will describe how each figure was replicated.

For convenience in discussing the figures of the original work, we will prepend figure names with "O" to distinguish them from the figures presented in this manuscript. For example, Figure 2 from the original work will be referred to as Figure O2.

Figure O2 has two panels, which are unlabeled in the original work. We will refer to these as Figure O2a (left panel) and Figure O2b (right panel) for convenience.

Figure O2a — Figure O2a shows the non-dimensionalized volume during the filling phase ($\frac{V_{fill}}{hw^2}$) against the ratio of channel height over channel width ($\frac{h}{w}$) for four width ratios ($\frac{w_{in}}{w}$). To replicate this figure, a series of 1000 data points were generated spanning $\frac{h}{w} \in [0, 0.5]$ for each $\frac{w_{in}}{w}$, arbitrarily setting $w = 1$, and varying h and w_{in} to obtain the necessary ratios.

The dimensionless fill volume was then calculated for each point using the module that was developed. These series of points were then plotted, the axes were scaled identically to the original work, and a color mapping was applied to the width ratios $\frac{w_{in}}{w}$.

Figure O2b — Figure O2b shows the squeezing coefficient (α) against the same range of $\frac{h}{w}$ values as Figure O2a. However, six width ratios ($\frac{w_{in}}{w}$) are plotted.

This figure was again replicated using 1000 data points per width ratio. Channel width (w) was held constant while h and w_{in} were varied. As in the original work the corner roundness $\varepsilon = 0$. The flow rate of the continuous phase was set as $q_c = 1$, and the flow rate of the gutter was $q_{gutter} = 0.1 * q_c$ such that $\frac{q_{gutter}}{q_c} = 0.1$, as described in the original work.

The squeezing coefficient was calculated based on these values using the function in the squeezing module, and plotted using the same axis scales as the original work.

Figure O3 — Figure O3 shows the dimensionless volume of droplets and bubbles against the flow rate ratio of the dispersed phase over the continuous phase ($\frac{q_d}{q_c}$). Six series are plotted, five are air bubbles produced using five width ratios ($\frac{w_{in}}{w}$) and varying heigth over width values ($\frac{h}{w}$); the final series shows liquid-liquid droplets with $\frac{w_{in}}{w} = 1$ and $\frac{h}{w} = 0.48$. For air bubbles $\varepsilon = 0.1w$ and for liquid-liquid droplets $\varepsilon = 0.01w$.

For replication, continuous flow rate (q_c) and channel width (w) were both set equal to 1. Channel height (h), inlet width (w_{in}), and dispersed phase flow rate (q_d) were all varied to achieve the necessary ratios. Corner roundness (ε) was computed based on channel width using the relationship described above. 1000 data points were generated for each of the six series.

The dimensionless volume of the squeezing phase and filling phase were calculated using their respective modules, and the sum of the two was computed. This total dimensionless volume was plotted in the same manner as in the original work.

Figure O6 – Figure O6 shows the shortest distance between the receding interface from the channel wall ($2r$) normalized by channel width ($2r/w$) against time (t) multiplied by $\frac{q_c}{hw^2}$. Where continuous phase flow rate $q_c = 3\mu L\ \text{min}^{-1}$, height $h = 33\mu m$, width $w = 100\mu m$, and corner roundness $\varepsilon = 0.1w$. Three series are shown, corresponding to various width ratios ($\frac{w_{in}}{w}$). Additionally, a horizontal line, $\frac{2r_{pinch}}{w} = \frac{h}{h+w}$, is added corresponding to their experimentally determined threshold at which pinch-off of the droplet occurs.

While the independent axis ($\frac{q_c}{hw^2}t$) can be generated using the provided geometries, and varying time to get values in the required range (*i.e.,* $\frac{q_c}{hw^2}t \in [0, 10]$), the dependent axis, $2r/w$ must computed as a function of $\frac{q_c}{hw^2}t$. Since $\frac{q_c}{hw^2}$ and w are all constant, this amounts to needing $2r$ as a function of t. Although no such equation is explicitly provided in the original work, it can be inferred.

To do so, the relationship between $2r$ and R, which is given in the original work, will be used:

$$2r - \varepsilon = R - \sqrt{(R-w)^2 + (R-w_{in})^2} \tag{8}$$

But first, an equation must be found that gives R as a function of time. The equation relating R and t is given by:

$$\left[2\frac{R}{w} + \frac{\pi}{4}\frac{h}{w}\right]\frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{R}{w}\right) = \frac{q_c}{hw^2}\left(1-\frac{\pi}{4}\right)^{-1}\left(1-\frac{q_{gutter}}{q_c}\right) \tag{9}$$

Multiplying both sides by w^2 and integrating with respect to $\mathrm{d}t$:

$$\int_0^t \left[2R + \frac{h\pi}{4}\right]\frac{\mathrm{d}R}{\mathrm{d}t}\mathrm{d}t = \int_0^t \frac{q_c}{h}\left(1-\frac{\pi}{4}\right)^{-1}\left(1-\frac{q_{gutter}}{q_c}\right)\mathrm{d}t$$

Evaluating the integrals:

$$R^2 + \frac{h\pi}{4}R\Big|_{t=0}^{t} = \frac{q_c}{h}\left(1-\frac{\pi}{4}\right)^{-1}\left(1-\frac{q_{gutter}}{q_c}\right)t\Big|_{t=0}^{t}$$

Where R is a function of time *i.e.,* $R = R(t)$ and since t is taken as the time since transition from filling to squeezing, $R(t)|_{t=0} = R_{fill}$. Evaluating yields:

$$R^2 - R_{fill}^2 + \frac{h\pi}{4}(R - R_{fill}) = \frac{q_c}{h}\left(1-\frac{\pi}{4}\right)^{-1}\left(1-\frac{q_{gutter}}{q_c}\right)t$$

Setting equal to zero and rearranging:

$$R^2 + \frac{h\pi}{4}R - R_{fill}^2 - \frac{h\pi}{4}R_{fill} - \frac{q_c}{h}\left(1-\frac{\pi}{4}\right)^{-1}\left(1-\frac{q_{gutter}}{q_c}\right)t = 0$$

R can be solved for using the quadratic equation to yield:

$$R = \frac{-\frac{h\pi}{4} \pm \sqrt{\left(\frac{h\pi}{4}\right)^2 + 4\left(R_{fill}^2 + \frac{h\pi}{4}R_{fill} + \frac{q_c}{h}\left(1-\frac{\pi}{4}\right)^{-1}\left(1-\frac{q_{gutter}}{q_c}\right)t\right)}}{2} \tag{10}$$

Therefore R can be evaluated as a function of time (taking the positive of the square root), and $2r$ can thus be calculated as:

$$2r = R - \sqrt{(R-w)^2 + (R-w_{in})^2} + \varepsilon \tag{11}$$

Functions for both R as a function of time, and $2r$ were added to the squeezing module.

For replication of Figure O6, the provided geometries for w, h, and q_c were used. 1000 points for each inlet width ratio were generated spanning $\frac{q_c}{hw^2}t \in [0, 10]$ by varying

t. Corner roundness (ε) was calculated based on w as described. $2r$ was computed using the squeezing module and divided by w for plotting. As in Figure O6, linetype was mapped to dispersed fluid type (liquid droplets vs. gas bubbles).

2.3 Code quality

To maintain quality of the code that was developed, several precautions were taken.

As previously mentioned, the functions developed in the filling and squeezing modules have associated unit tests. For the filling module, unit tests ensure that the verbose geometric calculations match the simplified equations in the original work. Pytest[16] v7.2.0 testing framework was used for all unit tests. Code coverage of the tests is assessed using coveragepy[17] v7.1.0. Additionally, an integration test was written for the Python script that generates the replicated figures, ensuring that it can be run and generate figures as output.

Black[18] v23.1.0 was used to format all Python files. Several static code checks were employed to automatically assess code quality. Pylint[19] v2.15.5 and Flake8[20] v5.0.4 were used for static code checking and linting, which adhere to PEP8[21] standards. Type annotations were employed in all Python code, and mypy[22] v0.990 was used for static type checking.

A Make[23] v4.2.1 target was added to perform automatic execution of all unit tests and static code checks with a single command.

3 Results

3.1 Python modules

Two Python modules were developed implementing the model presented in the original work.

The filling module consists of 8 functions and 8 unit tests. The squeezing module consists of 8 functions and 7 unit tests. Both modules have 100% test coverage as evaluated by coveragepy. All code passes static checking by Pylint, Flake8, and mypy.

3.2 Replication of figures

All figures showing modeling results in the original work were replicated.

Figure O2a — The figure generated using our modules (Figure 1a) is similar to Figure O2a in the original work. However, for $\frac{w_{in}}{w} > 1$, the lines have a markedly more negative slope than those in Figure O2a.

Interestingly however, a plot exactly matching Figure O2a could be achieved (Figure 1b). To do so, a separate function was written to calculate the dimensionless fill volume. In this function, the volume is not calculated using geometry, but instead used the equation in the original work, but neglecting a pair of parentheses.

The portion of equation (2) that was modified is as follows. The equation presented in the original work contains:

$$\left(\left(\frac{\pi}{2} - \arcsin\left(1 - \frac{w}{w_{in}}\right)\right)\frac{w_{in}}{w} + \frac{\pi}{2}\right)\frac{h}{w}$$

The rest of the equation was unmodified, however, the above portion was replaced with:

$$\left(\frac{\pi}{2} - \arcsin\left(1 - \frac{w}{w_{in}}\right)\frac{w_{in}}{w} + \frac{\pi}{2}\right)\frac{h}{w}$$

Which can be simplified to:

$$\left(\pi - \arcsin\left(1 - \frac{w}{w_{in}}\right)\frac{w_{in}}{w}\right)\frac{h}{w}$$

This is not equivalent to the unmodified version, *i.e.*,:

$$\left(\pi - \arcsin\left(1 - \frac{w}{w_{in}}\right)\frac{w_{in}}{w}\right)\frac{h}{w} \neq \left(\left(\frac{\pi}{2} - \arcsin\left(1 - \frac{w}{w_{in}}\right)\right)\frac{w_{in}}{w} + \frac{\pi}{2}\right)\frac{h}{w}$$

A unit test was also written which asserts that for several $\frac{w_{in}}{w} > 1$, the calculated dimensionless fill volume is not equal using the two different functions. The unit test also asserts that for $\frac{w_{in}}{w} = 1$, the two functions return the same value.

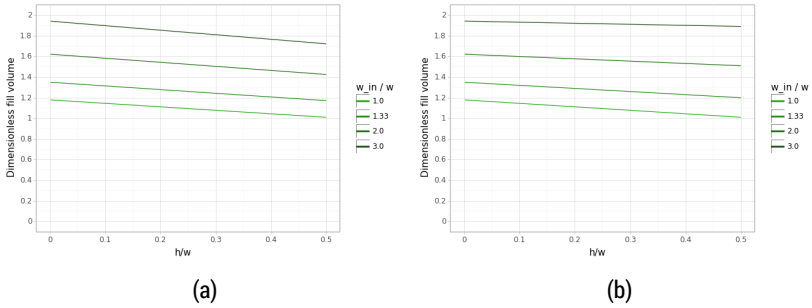

Figure 1. Dimensionless fill volume, $\frac{V_{fill}}{hw^2}$ against $\frac{h}{w}$ for four width ratios using a) the correct model, and b) an incorrect equation to exactly replicate Figure O2a.

Figure O2b — The plot of squeezing coefficient (Figure 2) exactly replicates Figure O2b.

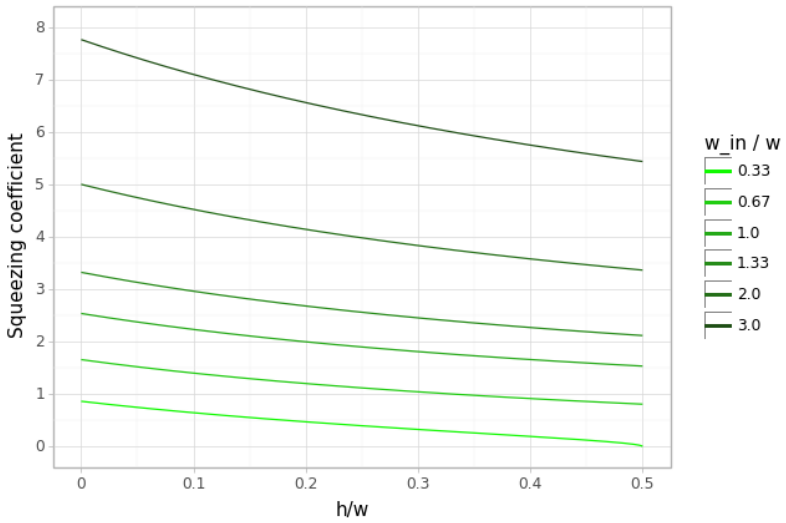

Figure 2. Squeezing coefficient, α against height / channel width for six width ratios. Replication of Figure O2b.

Figure O3 — The figure of total fill volume (Figure 3) very closely replicates Figure O3, with a small difference. For all lines corresponding to gas bubbles, the replicated figure appears to be an exact match. However, for the single line corresponding to the liquid-liquid droplets, the slope is steeper than in our figure.

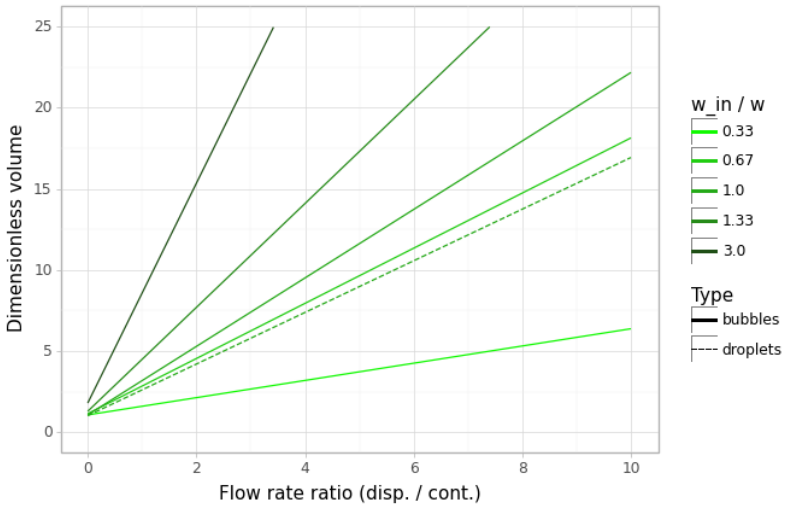

Figure 3. Total dimensionless volume, $\frac{V}{hw^2}$ against flow rate ratio $\frac{q_d}{q_c}$ for various width ratios and dispersed fluid types. Replication of Figure O3.

Figure O6 — The figure depicting the evolution of the receding interface (Figure 4) replicates the corresponding Figure O6 well.

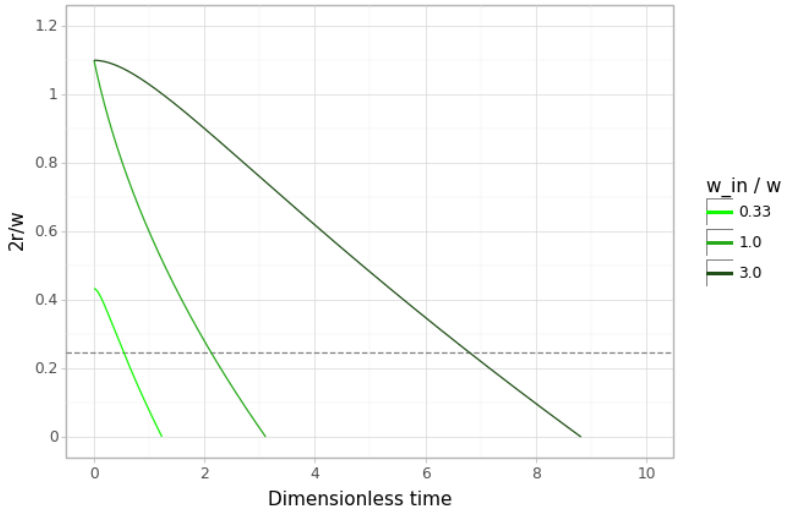

Figure 4. Evolution of the receding interface as $\frac{2r}{w}$ against dimensionless time, $\frac{q_c}{hw^2}t$ for various width ratios. Dashed line shows $\frac{2r_{pinch}}{w} = \frac{h}{h+w}$, the threshold at which the model predicts pinch off to occur. Replication of Figure O6.

4 Discussion

4.1 Model replication

Python modules were developed that successfully replicate the model in the original work. These modules have excellent code testing coverage, and adhere to Python community guidelines. Model logic was organized into modules to keep it separate from the specific geometries used to replicate the figures. By doing so, it is easier to test the func-

tions constituting the model and makes it easy to reuse the code for generalization. All module functions can be imported into any Python program and used for prediction.

4.2 Replication of figures

All figures could be replicated, although with one discrepancy in our figure corresponding to Figure O3.

Given that the same functions from the modules are used to calculate dimensionless volume for both liquid-liquid droplets and air bubbles, and that the air bubble results match exactly, it is unlikely that the modules are incorrect. This means that one or more of the provided geometries are different in our implementation and theirs.

Since the liquid-liquid droplet line appears to be labeled as $\frac{w_{in}}{w} = 0.67$ though the figure caption states $\frac{w_{in}}{w} = 1.0$, we considered that perhaps the true ratio was 0.67. However, reducing this ratio only reduces the slope, exacerbating the discrepancy.

Remarkably, it appears that the experimental results plotted in Figure O3 more closely match the line for liquid-liquid droplets in our Figure 3 than theirs. This may be an indication that when generating model results, they incorrectly entered one or more geometric parameter.

4.3 Challenges

Developing verbose geometry-based functions in the filling module was slightly difficult to do. The original work gives some explanation of how the equations were derived, but many steps were omitted. This is understandable since describing in full detail would result in extremely long exposition. Indeed, the granularity of the geometric calculations performed in the modules is greater than described here for the same reason.

While replication could have been done by simply using the simplified equation, having the geometry-based code provided assurance that our model implementation was correct since we could test the results of the verbose code against the simplified equation. This became very useful when exact replication of Figure O2a was elusive.

Exact replication of Figure O2a was a matter of luck and the product of typing the simplified equation into several programming languages (assuming we had made some mistake), despite the verbose code matching our implementation of the simplified equation. It is fortunate that we happened to eventually make the same mistake as the original authors one of those times. This allowed us to exactly replicate their figure.

The fact that the original authors, and eventually we, made mistakes in entering that equation demonstrates how easy it is to make mistakes that can greatly affect results when writing code. This is precisely why tests are so important. Unit testing makes it far more likely that such bugs will be caught in development.

Perhaps the greatest challenge to replicating the original results came from their omission of an explicit equation providing $2r$ as a function of time t, which must have been used to generate Figure O6. Further, the authors did not indicate which equations could be used to derive such a relationship. It was with a fair amount of effort that the relationship was derived for this replication.

5 Conclusion

The model in the original work could be fully replicated based solely on the contents of that article. However, two discrepancies in the figures were encountered upon replication. The first was likely due to a mistake in the original code; the original figure was successfully replicated in this work by deliberately using an equation that was incorrect. Although the second discrepancy remains, it is likely due to differences in the geometries provided to generate the original figure and those presented in the figure caption.

This notion is supported by the closer alignment of their experimental results to our modeling results than theirs.

6 Code Availability

All source code is available at https://github.com/schackartk/t-junction_model_replication

7 Author Contributions

We report author contributions according to the CRediT taxonomy[24].
KS: Conceptualization, Formal Analysis, Investigation, Methodology, Project Administration, Software, Validation, Visualization, Writing — original draft, Writing — review & editing;
KK: Conceptualization, Investigation, Validation, Writing — original draft, Writing — review & editing

8 Acknowledgements

The authors would like to thank Dr. Qiudong Wang at the University of Arizona for challenging KS to replicate the results of a math modeling paper as a final project, and for providing feedback on the first iteration of this project (originally written in MATLAB, available at https://github.com/schackartk/MATH585).

References

1. D. Huang, K. Wang, Y. Wang, H. Sun, X. Liang, and T. Meng. "Precise control for the size of droplet in T-junction microfluidic based on iterative learning method." en. In: **Journal of the Franklin Institute** 357.9 (June 2020), pp. 5302–5316. DOI: 10.1016/j.jfranklin.2020.02.046. URL: https://www.sciencedirect.com/science/article/pii/S0016003220301307 (visited on 02/20/2023).
2. R. Dreyfus, P. Tabeling, and H. Willaime. "Ordered and disordered patterns in two-phase flows in microchannels." eng. In: **Physical Review Letters** 90.14 (Apr. 2003), p. 144505. DOI: 10.1103/PhysRevLett.90.144505.
3. X. Casadevall i Solvas and A. deMello. "Droplet microfluidics: recent developments and future applications." eng. In: **Chemical Communications (Cambridge, England)** 47.7 (Feb. 2011), pp. 1936–1942. DOI: 10.1039/c0cc02474k.
4. A. M. Ibrahim, J. I. Padovani, R. T. Howe, and Y. H. Anis. "Modeling of Droplet Generation in a Microfluidic Flow-Focusing Junction for Droplet Size Control." en. In: **Micromachines** 12.6 (June 2021). Number: 6 Publisher: Multidisciplinary Digital Publishing Institute, p. 590. DOI: 10.3390/mi12060590. URL: https://www.mdpi.com/2072-666X/12/6/590 (visited on 02/20/2023).
5. H. Liu and Y. Zhang. "Droplet formation in a T-shaped microfluidic junction." In: **Journal of Applied Physics** 106.3 (Aug. 2009). Publisher: American Institute of Physics, p. 034906. DOI: 10.1063/1.3187831. URL: https://aip.scitation.org/doi/10.1063/1.3187831 (visited on 02/22/2023).
6. M. L. J. Steegmans, C. G. P. H. Schroën, and R. M. Boom. "Generalised insights in droplet formation at T-junctions through statistical analysis." en. In: **Chemical Engineering Science** 64.13 (July 2009), pp. 3042–3050. DOI: 10.1016/j.ces.2009.03.010. URL: https : / / www . sciencedirect . com / science / article / pii / S000925090900181X (visited on 02/22/2023).
7. X.-B. Li, F.-C. Li, J.-C. Yang, H. Kinoshita, M. Oishi, and M. Oshima. "Study on the mechanism of droplet formation in T-junction microchannel." en. In: **Chemical Engineering Science** 69.1 (Feb. 2012), pp. 340–351. DOI: 10.1016/j.ces.2011.10.048. URL: https://www.sciencedirect.com/science/article/pii/S0009250911007627 (visited on 02/22/2023).
8. G. Y. Soh, G. H. Yeoh, and V. Timchenko. "Numerical investigation on the velocity fields during droplet formation in a microfluidic T-junction." en. In: **Chemical Engineering Science** 139 (Jan. 2016), pp. 99–108. DOI: 10.1016/j.ces.2015.09.025. URL: https://www.sciencedirect.com/science/article/pii/S0009250915006478 (visited on 02/22/2023).

9. V. van Steijn, C. R. Kleijn, and M. T. Kreutzer. "Predictive model for the size of bubbles and droplets created in microfluidic T-junctions." In: **Lab Chip** 10.19 (2010). Publisher: The Royal Society of Chemistry, pp. 2513–2518. DOI: 10.1039/C002625E. URL: http://dx.doi.org/10.1039/C002625E.
10. pypi. **Python Package Index - PyPI**. URL: https://pypi.org/.
11. **Conda**. 2017. URL: https://www.anaconda.com.
12. Microsoft. **Windows Subsystem for Linux (WSL)**. URL: https://learn.microsoft.com/en-us/windows/wsl/install.
13. C. Ltd. **Ubuntu**. Apr. 2020. URL: https://releases.ubuntu.com/focal/.
14. T. p. d. team. **Pandas**. Feb. 2020. DOI: 10.5281/zenodo.3509134. URL: https://doi.org/10.5281/zenodo.3509134.
15. H. Kibirige et al. **plotnine: A grammar of graphics for Python**. Sept. 2022. DOI: 10.5281/zenodo.7124918. URL: https://zenodo.org/record/7124918 (visited on 02/02/2023).
16. H. Krekel. **pytest: The pytest framework makes it easy to write small tests, yet scales to support complex functional testing.** 2004. URL: https://github.com/pytest-dev/pytest.
17. N. Batchelder and G. Rees. **coveragepy: The code coverage tool for Python**. URL: https://github.com/nedbat/coveragepy/tree/master/coverage.
18. Ł. Langa. **Black: The uncompromising Python code formatter**. 2018. URL: https://github.com/psf/black.
19. S. Thénault. **Pylint: It's not just a linter that annoys you!** 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA, 2001. URL: https://github.com/PyCQA/pylint.
20. T. Ziade and I. Cordasco. **Flake8: Your Tool For Style Guide Enforcement**. 2011. URL: https://github.com/PyCQA/flake8.
21. G. Van Rossum, B. Warsaw, and N. Coghlan. "PEP 8–style guide for python code." In: **Python. org** 1565 (2001).
22. J. Lehtosalo. **mypy: Optional static typing for Python**. 2012. URL: https://github.com/python/mypy.
23. S. Feldman. **GNU Make**. 1988. URL: https://www.gnu.org/software/make/.
24. L. Allen, A. O'Connell, and V. Kiermer. "How can we ensure visibility and diversity in research contributions? How the Contributor Role Taxonomy (CRediT) is helping the shift from authorship to contributorship." en. In: **Learned Publishing** 32.1 (2019). _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/leap.1210, pp. 71–74. DOI: 10.1002/leap.1210. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/leap.1210 (visited on 10/14/2022).