u-DROP and DAFD: Tools for Microfluidic Droplet Generator Design

Christopher Rodriguez[1,2], Ali Lashkaripour[3], Douglas Densmore[4]

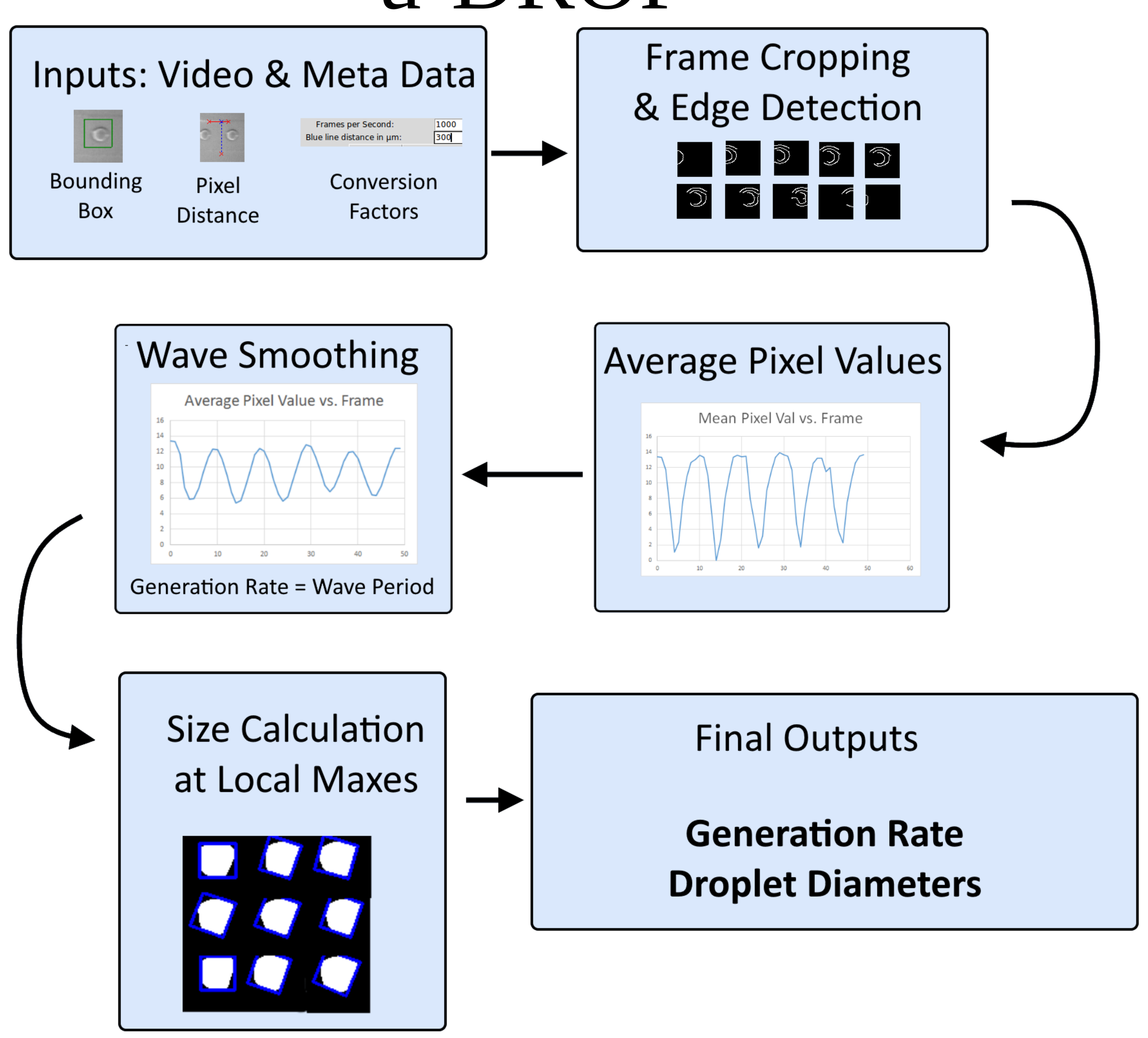
[1]Boston University Bioinformatics BRITE REU Program, Summer 2017; [2]Louisiana Tech University, Cyber Engineering;
[2]Boston University, Department of Biomedical Engineering; [4]Boston University, Department of of Electrical and Computer Engineering

Abstract

Microfluidic chips, devices that transport fluids through channels on the micron scale, help make biological experiments more cost and time efficient. However, designing these chips can be challenging for a biologist entering the field. Specifically, a droplet generator, one of the most basic microfluidic components, requires many iterations of testing and validation to ensure it produces droplets at the correct rate and of the correct size. The goal of this research project was to make droplet generator design faster and simpler through the software we developed. The first program, Micro Droplet Rate/Region Ocular Processing (u-DROP), was used to determine the droplet size and generation rate produced from various experiments in which we changed the chip's orifice size, aspect ratio, width ratio, orifice length, the water and oil input width, the capillary number, and the flow rate. The data gathered from these experiments was used by the second program, Design Automation based on Fluid Dynamics (DAFD) to suggest a droplet generator design that can produce droplets at the specified rate and size. DAFD employs two interpolation models, one to output generation rate and one to output droplet size, to make its predictions. DAFD was validated on a test data set of 2500 randomly generated points.

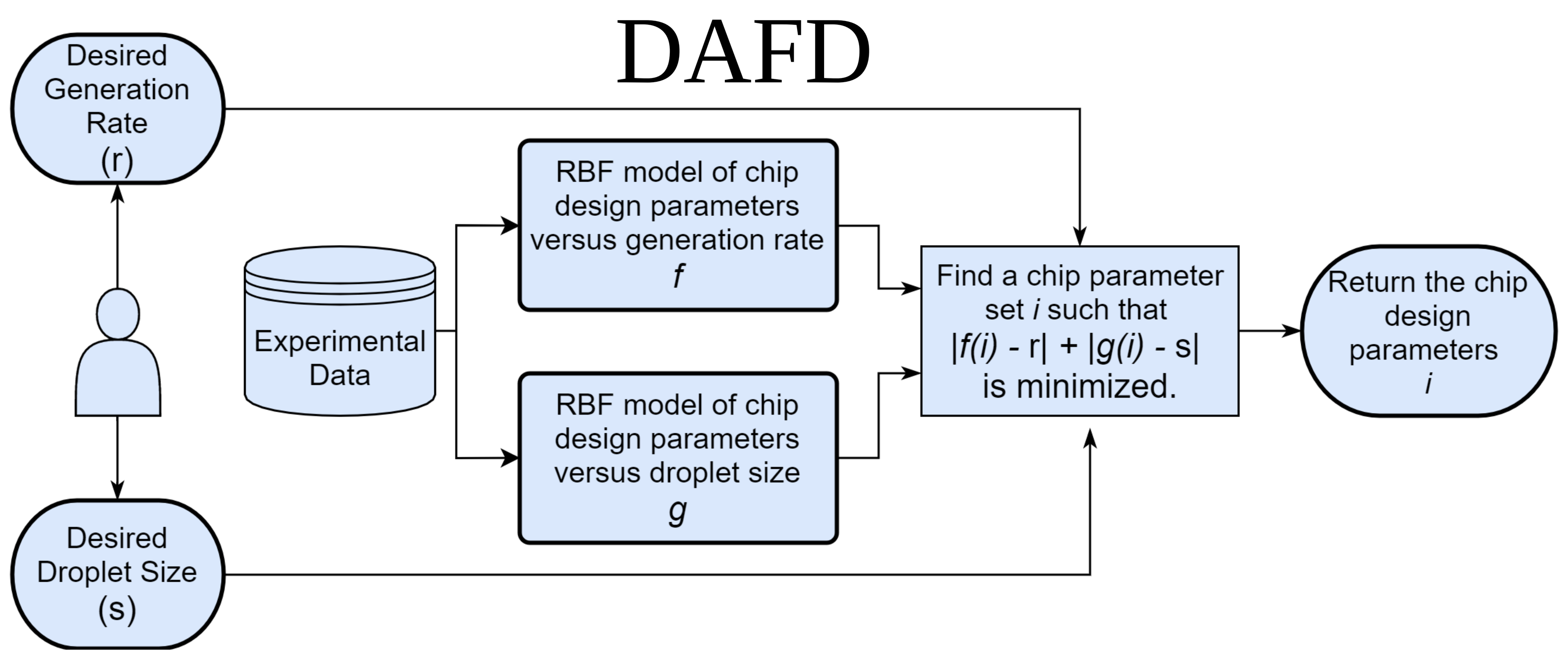
Methods

u-DROP



Measures the droplet generation rate and droplet sizes from a given droplet generation video. Used to collect data for DAFD.

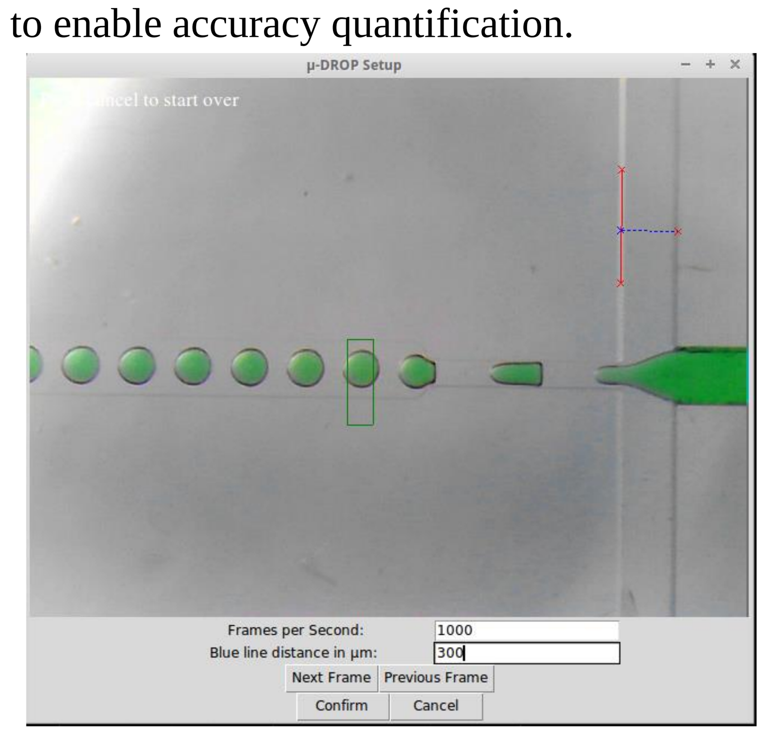
DAFD



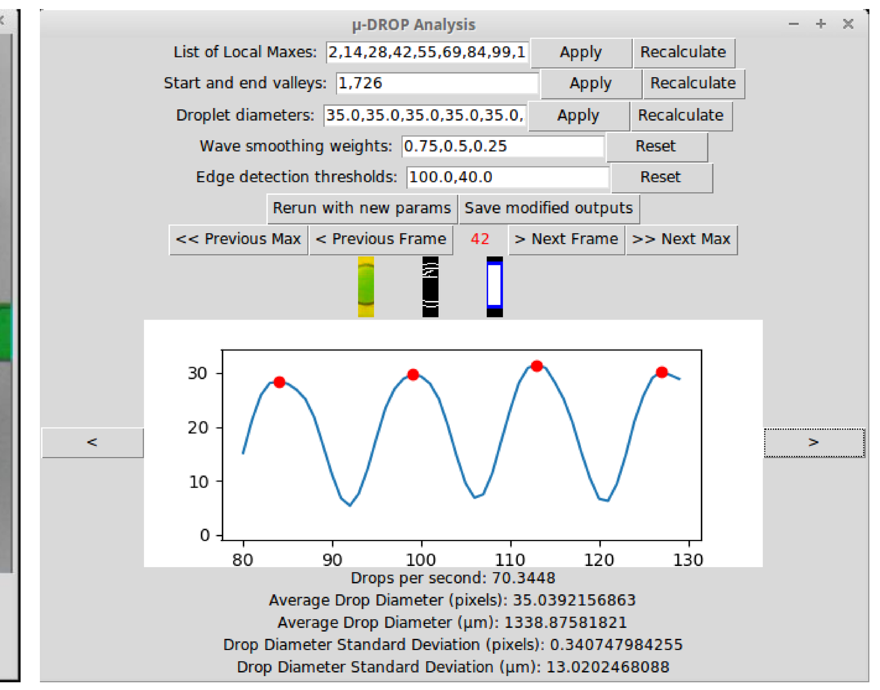
Gives a user the chip design parameters necessary to build a droplet generator that produces the desired generation rate and droplet size.

u-DROP Results

u-DROP works well even in videos with subpar lighting and resolution. Apart from low frame rate videos, u-DROP worked well on all videos provided with almost no adjustments needed. An analysis tool was included in u-DROP to provide detailed information about the results and to enable accuracy quantification.

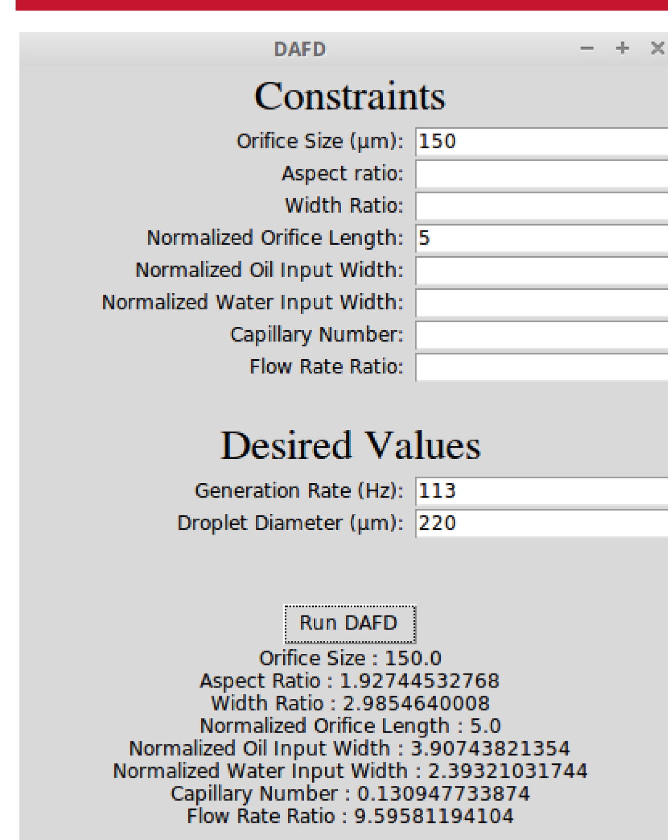


u-DROP inputs



u-DROP outputs and analysis

DAFD Results



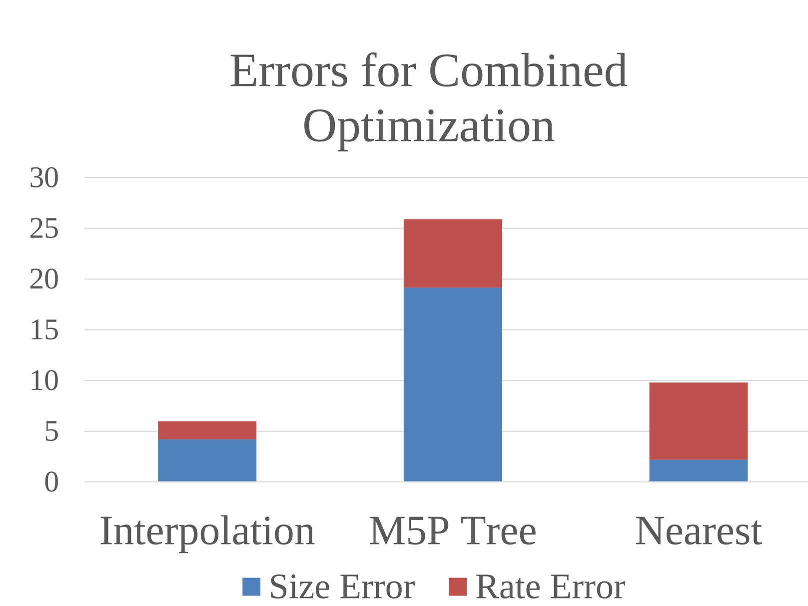
DAFD GUI

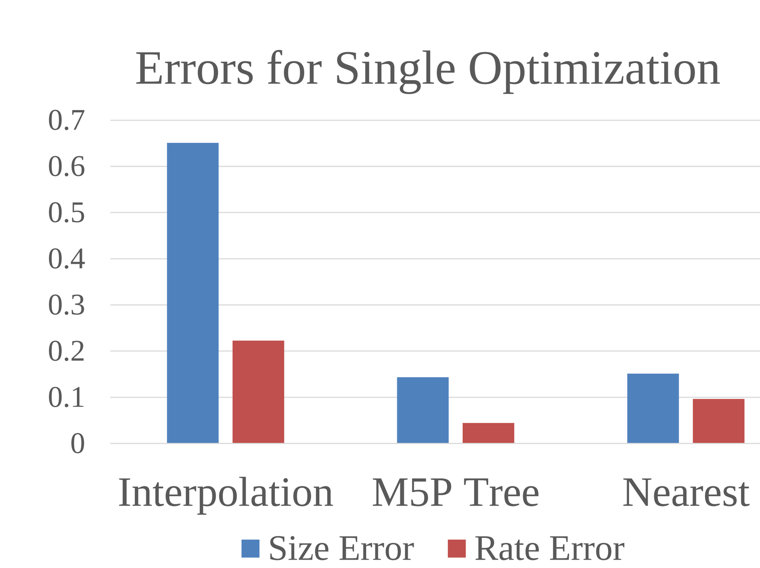
$$f_r(i) = \frac{OS * AR * WR * OL * WIW}{OIW * CN * FRR * 10}$$

$$g_r(i) = \frac{(OIW + AR + WR + OL + WIW + FRR) * CN * 5000}{OS}$$

DAFD was tested with a fake simulated to prove our interpolation parameters are optimal. The simuated dataset is 2500 points created by running 8 inputs (randomly chosen to be within the range of our real data) through function f_r to get generation rate and g_r to get droplet size rate. f_r and g_r are functions we created that are meant to be a simple (but inaccurate) representation of the real world function (which is mathematically difficult to determine) that maps our inputs to outputs.

The error for a given input i can be calculated as $\left|\frac{f_r(i) - f(i)}{f(i)}\right|$ for the generation rate and $\left|\frac{g_r(i) - g(i)}{g(i)}\right|$ for the droplet size. We created a dataset of 1000 desired rate/size points to test DAFD. We ran an experiment where DAFD tried to optimize on both rate and size, and then we ran two more where DAFD just optimized on rate or size. We tested RBF interpolation (our current model), M5P linear regression trees, and a simple nearest data point method.





	Size Error	Rate Error	Sum Error
Interpolation	4.20%	1.77%	5.97%
M5P Tree	19.14%	6.76%	25.90%
Nearest Data Point	2.18%	7.62%	9.80%

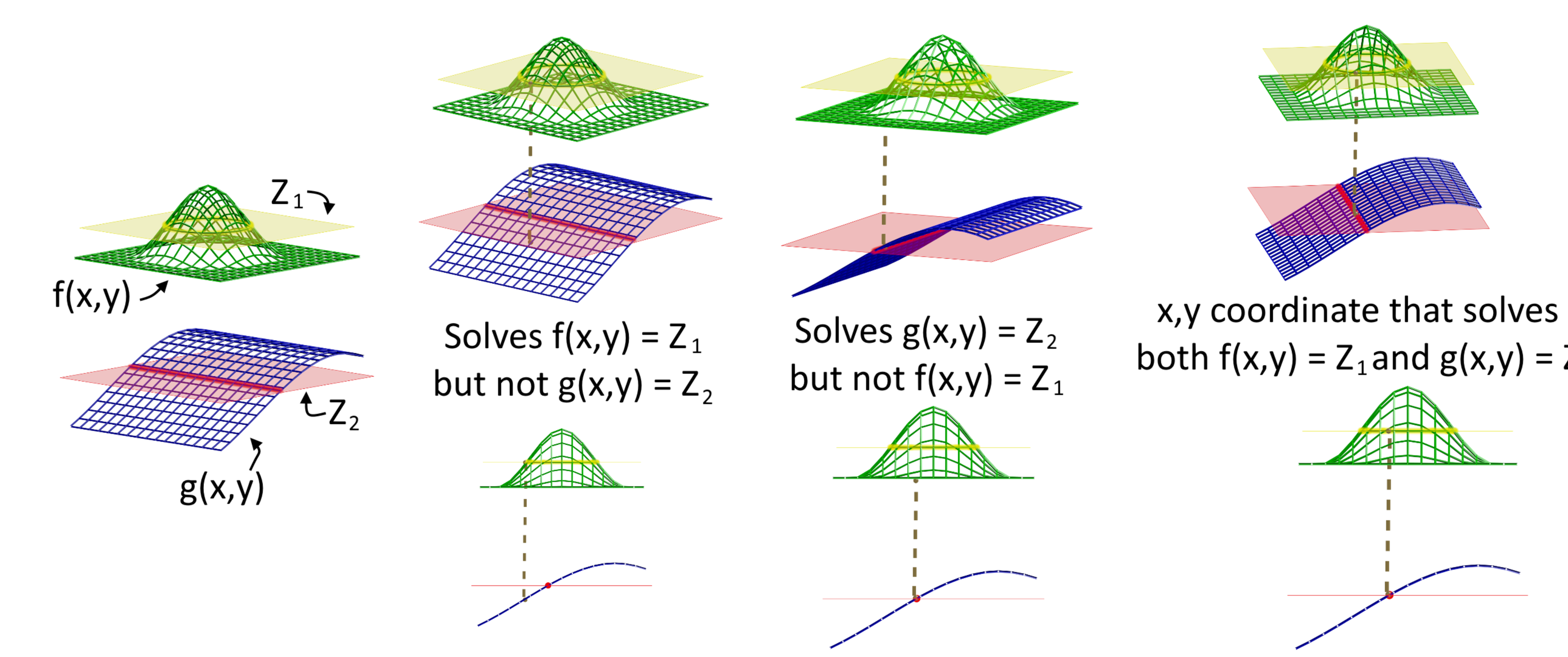
	Size Error	Rate Error
Interpolation	0.65%	0.22%
M5P Tree	0.14%	0.04%
Nearest Data Point	0.15%	0.10%

Discussion and Future Works

u-DROP's high accuracy and "glass box" design philosophy make it a great tool for the specific problem of droplet generation analysis. Droplet generators, however, are not the only components in microfluidics. For the next step of u-DROP, we are looking to expand the program to analyze videos of droplet fission. The preliminary tests have shown that u-DROP can handle these videos with very little modifications. We hope that u-DROP can be expanded to analyze all basic microfluidic primitives. Our future goals for u-DROP are not only to make the program a data gathering tool for DAFD but also to have the program be useful as a standalone tool to speed up analysis of already milled chips.

We are currently working on finishing the experiments to gather the real data for DAFD. There is no way to computationally check DAFD's accuracy with the real data, so we must do it experimentally.

Errors in DAFD can come from either incorrect fitting of the interpolation models to the real data or an inability of the program to find a solution that satisfies both the models. The second type, combined model errors, is equal to the difference between combined and single optimization error. This error will become more prevalent when expanding the number of output variables and constraining inputs. The figure below explains the error and the general premise behind reverse predictive modelling.



u-DROP and DAFD are part of a larger family of tools at CIDAR lab that attempt to make microfluidics cheaper, faster, and more accessible for all researchers. We are working to integrate DAFD into the existing pipeline. Specifically, we want to combine DAFD with our automatic chip design tool Fluigi so that users do not have to provide as many constraints as they did before. When DAFD and u-DROP are expanded to cover more microfluidic primitives, chip design can be abstracted to setting the constraints for the experiments and letting the CIDAR tools output a chip that can perform that specific experiment accurately and efficiently.

Additional Reading and Acknowledgements

- Wei-Yin Loh (2011). Classification and Regression Trees. WIREs Data Mining and Knowledge Discovery, Volume 1, 14-23
- Link, Anna, Weitz, Stone (2004). Geometrically Mediated Breakup of Drops in Microfluidic Devices. Physical Review Letters, Volume 92, 4 pages
- John Canny (1986). A Computational Approach to Edge Detection. IEEE Transactions on Pattern Analysis and Machine Intelligence, Volume 8, 679-698
- Ronald L. Graham (1983). Finding the Convex Hull of a Simple Polygon. Journal of Algorithms, Volume 4, 324-331
- M. Powell (1978). A fast algorithm for nonlinearly constrained optimization calculations. Lecture Notes in Mathematics, Volume 630, 144-157
- Amar Basu (2012). Droplet Tracking Velocimetry (DTV) Automated Measurement of Droplet Motion and Shape Using Digital Image Processing. Miniaturized Systems for Chemistry and Life Sciences, Volume 16, 1894-1896

This work was funded, in part, by NSF grant DBI-1559829, awarded to the Boston University Bioinformatics BRITE REU program