The Case for Semi-Automated Design of Microfluidic Very Large Scale Integration (mVLSI) Chips

Jeffrey McDaniel*, William H. Grover†, and Philip Brisk*

*Department of Computer Science and Engineering
University of California, Riverside
Email: jmcda001@cs.ucr.edu, philip@cs.ucr.edu

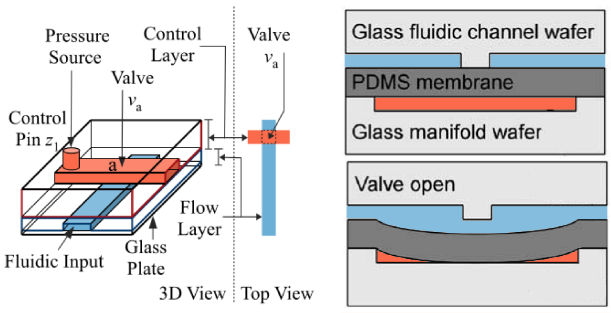
†Department of Bioengineering
University of California, Riverside
Email: wgrover@engr.ucr.edu

***Abstract*—In recent years, significant interest has emerged in the problem of fully automating the design of microfluidic very large scale integration (mVLSI) chips, a popular class of Lab-on-a-Chip (LoC) devices that can automatically execute a wide variety of biological assays. To date, this work has been carried out with little to no input from LoC designers. We conducted interviews with approximately 100 LoC designers, biologists, and chemists from academia and industry; uniformly, they expressed frustration with existing design solutions, primarily commercially available software such as AutoCAD and Solidworks; however, they expressed limited interest and considerable skepticism about the potential for "push-button" end-to-end automation. In response, we have developed a semi-automated mVLSI drawing tool that is designed specifically to address the pain points elucidated by our interviewees. We have used this tool to rapidly reproduce several previously published LoC architectures and generate fabrication ready specifications.**

I. Introduction

Laboratories-on-a-chip (LoCs) based on microfluidic (Very) Large Scale Integration (mVLSI) technology [1]–[5] have received considerable interest from the design automation community in recent years. Motivated by the success of the semiconductor VLSI/CAD ecosystem, many researchers are investigating techniques to automate the design of mVLSI LoCs; however, it is unclear if there actually exists a market for mVLSI CAD software, or whether the proposed automation techniques addresses actual pain points experienced by LoC designers on a day-to-day basis.

This paper attempts to bring clarity to these issues. The authors received funding from the US National Science Foundation (NSF) Innovation-Corps (I-Corps) program to study the commercial potential for mVLSI CAD software. Through the I-Corps program, the authors embarked on an extensive customer discovery process. They interviewed ∼100 LoC designers and other potential customers for mVLSI CAD software. The authors discovered that current LoC designers find existing toolflows ill-suited to their needs; however, they have a strong preference to address specific shortcomings of existing workflows. They do not trust and have no interest in fully-auomated "push-button" end-to-end solutions. Their primary concern is to quickly obtain a working device with the minimum amount of prototyping; they have no interest in optimization metrics derived from the semiconductor industry, and truly do not care whether a solution is optimal or near-optimal, as long as the chip produced works correctly.



(a) (b)

Fig. 1: Microvalves control the motion of fluid through an mVLSI LoC. (a) A multilayer soft lithography microvalve [1]; (b) a monolithic membrane valve [3].

Contribution: This paper proposes semi-automated mVLSI CAD as a solution that addresses the pain points experienced by LoC designers. LoC designers can benefit from interactive software that provides guidance at every step of the design process. In terms of research challenges, this necessites re-thinking the algorithms and optimization criteria that are used. In particular, there is a need for: (1) faster algorithms capable of performing smaller discrete steps in seconds or minutes; (2) algorithms that generate partial solutions which do not alter user-specified design constraints; and (3) algorithms based on the inputs that would provided by an expert LoC designer, rather than the more ideal set of inputs that one would obtain in a fully automated design flow.

This paper also highlights mistaken assumptions that underlie much of the prior work on semi-automated mVLSI CAD. Aside from the fact that LoC designers do not desire solutions that limit their ability to control the design process, many of the optimization criteria that prior work has emphasized are in fact irrelevant to LoC design today. Optimization metrics that derive from semiconductor VLSI are not good proxies for either performance or cost when applied to mVLSI technology. By correcting these mistaken assumptions, mVLSI CAD should pivot to increase its relevance to LoC design practices today.

II. Technology Overview

The basic building block of mVLSI technology is the integrated microvalve [1], [3], which is actuated by external pressure. The microvalve is the fluidic analogue to the transistor; it is the basic building block of larger components such as pumps, mixers, multiplexers, etc. [2], [4]. mVLSI CAD algorithms typically target microvalves fabricated using multi-layer soft lithography [1] (Fig. 1a), or monolithic membrane valves [3] (Fig. 1b). Microvalves of either type are actuated (opened/closed) by external pressure.

Between the two technologies, succesors to the original multi-layer soft lithography microvalves have achieved greater integration densities [5]; however, the multi-layer soft lithography fabrication and assembly processes (details omitted to conserve space) are far more complex and expensive than the simple glass etching required to produce an LoC with monolithic membrane valves. For these reasons, we expect monolithic membrane valves to be a more attractive market solution for startup companies and for product with low or uncertain (at design time) production volumes.

III. Customer Discovery

Customer discovery starts with an idea and vision (in this case, mVLSI CAD software), converts the vision into a series of hypotheses, which can be validated or invalidated through a series of interviews with potential customers [6]. The objective, here is to reduce risk by *objectively* interviewing potential customers. The first step was to understand the pain points experienced by potential customers (LoC designers) and to understand whether or not mVLSI CAD software can address them; if not, it is highly unlikely that a potential customer would be willing to pay for a product.

As participants in the NSF I-Corps program, the authors engaged in customer discovery for mVLSI CAD software, and spoke with $\sim$100 potential customers from academia and industry. This yielded a detailed understand of current workflows as well as a set of common pain points experienced by LoC designers; moreover, the process revealed a number of faulty assumptions that have been implicit in academic mVLSI CAD research up until this point. We summarize these discoveries in the following sections.

IV. mVLSI CAD: Fallacies and Reality

The customer discovery process elucidated a number of mistaken assumptions that underlie prior work on mVLSI CAD; through reptition and citation, these fallacies propagate. This section attempts to correct the record.

Fallacy: LoC designers appreciate analogies between LoCs and semiconductor products.
Reality: It is acknowledged that both transistors and microvalves are building blocks of larger integrated devices; the analogy ends there. LoC designers' primary concern is the correctness of the LoC to perform the desired biological function, and see the design process as being quite distinct from semiconductor VLSI/CAD.

Fallacy: LoC designers desire end-to-end "push-button" design automation software
Reality: First, LoC designers do not trust "push-button" solutions; they want control of all aspects of the design at all times. They are well aware of the shortcomings of their current workflows and are open to improvements. They believe that the shortcomings of a completely new tool would cause more problems than it would solve.

Fallacy: Physical design is a pain point for LoC designers.
Reality: At present, LoC design is performed manually. The proper design of components is a pain point. The lack of a common library, database, or repository of components is a pain point; the present solution appears to be institutional memory, which is quite fragile, especially when employees switch jobs. Given a set of components, an expert designer can place and route the flow layer of a moderately-sized mVLSI chip in tens of minutes.

Fallacy: Semiconductor optimization metrics are relevant.
Reality: An LoC designer wants to produce a chip that works properly; cost, reliability, and testability are concerns. Semiconductor-derived metrics, such as fluid channel length [7], [8] or control skew [9] are low priorities. Fluid channel length affects millisecond fluid transport times for experiments that run for tens of minutes, hours, or days; today's mVLSI chips are not clocked aggressively, so control channel skew is at most a future concern that relies on performance-related assumptions not presently on industry's radar.

Fallacy: Area reduction affects cost
Reality: The number of mVLSI chips that can fit onto a mold is several orders of magnitude smaller than the number of semiconductor VLSI chips that can fit onto a wafer. Small reductions in area will not increase the number of chips per mold, except in corner cases. The cost of glass and PDMS for monolithic membrane valve chips is not a present concern.

Fallacy: LoC designers care about algorithmic optimality
Reality: Current LoC design workflows *interactive*. Designers are not opposed to integrating mVLSI CAD algorithms, as long as they have the opportunity to manually edit and/or undo the result. Fast-running algorithms can boost productivity, which would be appreciated; long-running algorithms (e.g., ILP [8]) would not be used.

Fallacy: The "input" to mVLSI design flow is digital.
Reality: Academic work on architectural synthesis and application mapping assumes that the assay is input to the system using a domain-specific language [10] or sequencing graph [11], [12]. LoC designers are not Computer Scientists and do not think this way. They are resistant to adopting any new tool that imposes a learning curve.

V. Manual mVLSI Design

Fig. 2a depicts the current workflow for manual LoC design. The first step is to verify the target bioassay on traditional benchtop equipment; starting with the validated bioassay, the subsequent (co-)design process is equal parts bioassay miniaturization and LoC design. The premise that an immutable assay specification exists prior to the start of the LoC design process [10]–[12] is unfortunately unrealistic.

The next step two steps of the manual design process involve the creation of a 2D sketch of the LoC; these steps are not a particular bottleneck or pain point, but represent the topics that have received the most attention from the CAD research community: architectural synthesis and physical design. The designer allocates microfluidic components are needed, generates the interconnect architecture, and generates a 2D layout, inserting switches to ensure planarity. This step is typically done using pencil-and-paper, although several LoC designers have mentioned using software, such as MS Paint.

The fourth step is to convert the sketch into a format that can produce masks for the fabrication process. This is typically done using commercial 3D CAD modeling software, such as AutoCAD or Solidworks; the occasional designer reported using simpler software, such as MS Paint, for designs of small to medium complexity. The 3D CAD modeling software has a high learning curve, despite the fact that LoC design uses a small fraction of the software's available features.

The fifth and sixth steps are to fabricate and test the LoC. This can be costly and/or time-consuming, depending on whether fabrication is performed in-house or contracted out to a foundry. In-house fabrication is time-efficient, but requires paying specialists; when contracting with a foundry, the turnaround time may be weeks or months. More often than not, the initial design fails, yielding an iterative refinement process, which is also a pain point. Reducing the time and cost of these iterations is a much higher priority for LoC designers than automating the 2D sketching process.

VI. Semi-Automated mVSLI Design

A semi-automated mVLSI design flow can address the pain points that mVLSI LoC designers regularly experience, without disrupting their existing workflows. As shown in Fig. 2b, mVLSI CAD algorithms can still play an important role, however, they must be interactive (no LoC designer will wait tens of minutes or hours for ILPs to converge [8]), while supporting manual intervention and incremental changes to the design. Specifically, the expectations surrounding the mVLSI CAD algorithms, and how they will be used, must change; their objective should be to provide guidance to the designer, not to replace the existing design process. The designer must maintain manual control of the overall design process, while allowing CAD algorithms to handle some of the more tedious parts of the design.

A use case of a scientist who is designing a new LoC for commercial use will illustrate key aspects of the design flow and its use cases. The use case selected is the LoC device for automated chromatin immunoprecipitation [13]. The original manual placed and routed device is shown in Fig. 3.

A. System Initialization

The core of the software is a collection of mVLSI components which are capable of executing the required operations of biological experiments. These components can be stored in a library of entity files, described using a structural netlist specification language such as MHDL [14] or Mint [15]. Components are sorted based on their corresponding fabrication technologies; once a technology is selected, the user is provided with a list of entries and an open grid workspace (Fig. 4). The interface also presents a bucket of available *containers* that define the device area based on standard dimensions, such as the area of a microscope slide.

B. Placement and Routing

Placement is performed manually by default. The design drags the desired entities into the workspace area, forming components, which can be moved around and copied. If desired, a new component can be added to the library. Many large designs repeatedly make use of smaller components and subsystems, which can be placed either manually or algorithmically. Automated placement algorithms can be used if desired, but this is not expected to be a common feature.

Full and semi-automated routing solutions will be available. The designer will specify connections between two (or more) components by clicking on them. Prior to routing, channels will be visually displayed as straight-line connections between components, as shown in Fig. 5.

The key stipulation, here, is that the routing algorithm needs to be fast; optimality is less of a concern than algorithmic runtime and the aesthetics of the route. The router works behind the scenes on a pre-placed device, with the objective being to display routes to the designer (Fig. 6). The current implementation uses a network flow-based router published previously [16].

Three different routing scenarios are envisioned:

Incremental Routing: A new channel is routed when designer clicks; existing routes are not perturbed.

Incremental Routing with Rip-up and Re-route: A new channel is routed when the designer clicks, ripping up and rerouting existing routes as needed.

Batch Routing: All nets are routed at once.

Placement and routing are expected to be incremental and iterative. Given a placed and routed netlist, a designer is equally likely to add a new component, remove an existing component, or move a component, which may affect its routes. When adding a new component, the designer is likely to immediately route all desired connections to existing components; when removing a component incident on multi-terminal routing channels, the channels must be adjusted to maintain connectivity between components that were not removed. And when a component is moved, its incident channels may need to be rerouted as soon as its desired location is selected.

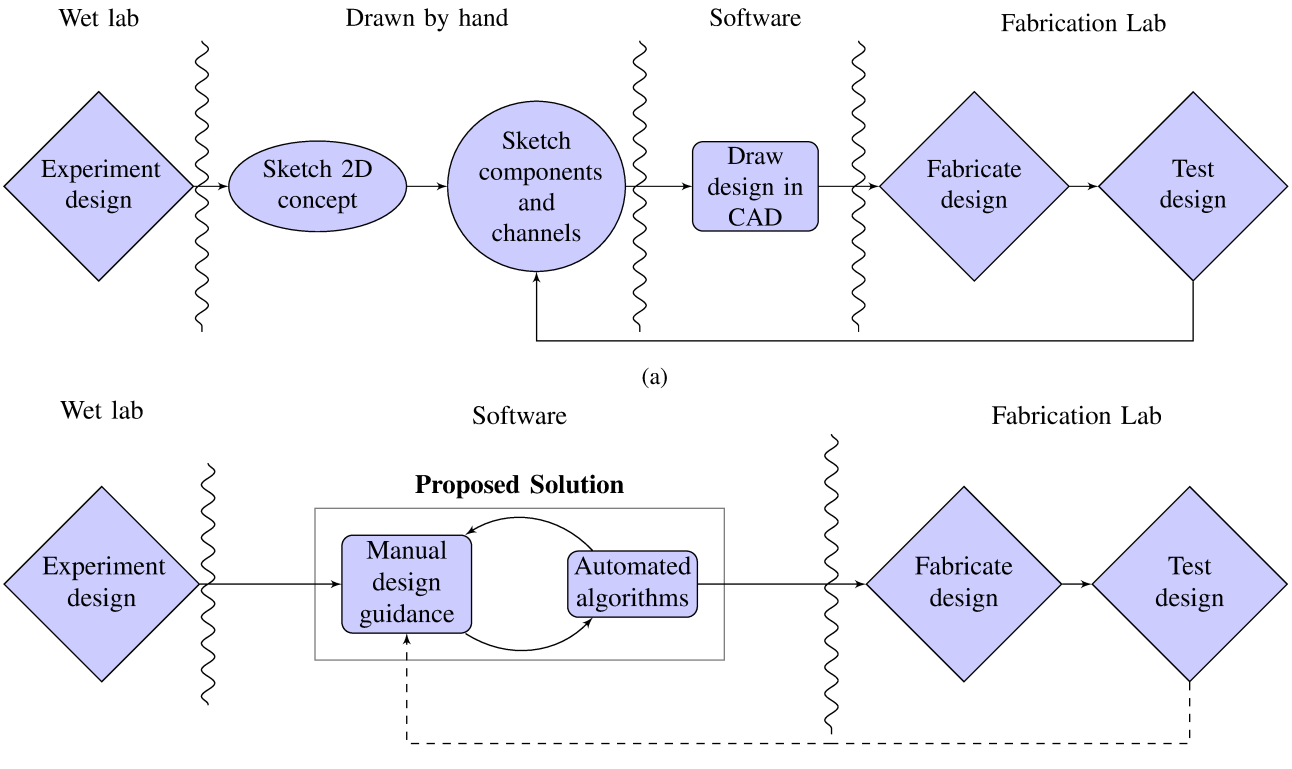


Fig. 2: Current manual (a) and proposed semi-automated (b) workflows for mVLSI LoC design. CAD algorithms can augment the semi-automated workflow, accelerating the design process without replacing it entirely.

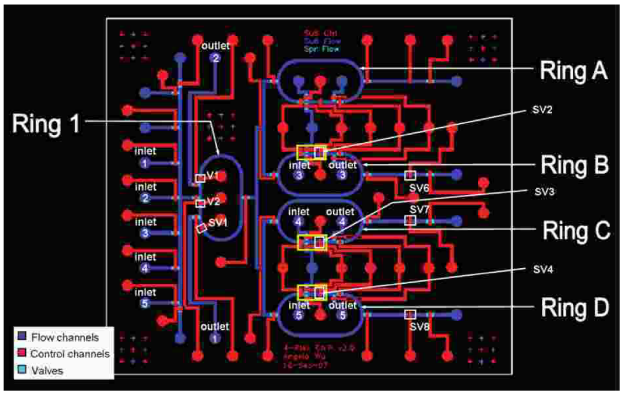


Fig. 3: TA manually placed and routed chromatin immunoprecipitation device [13]

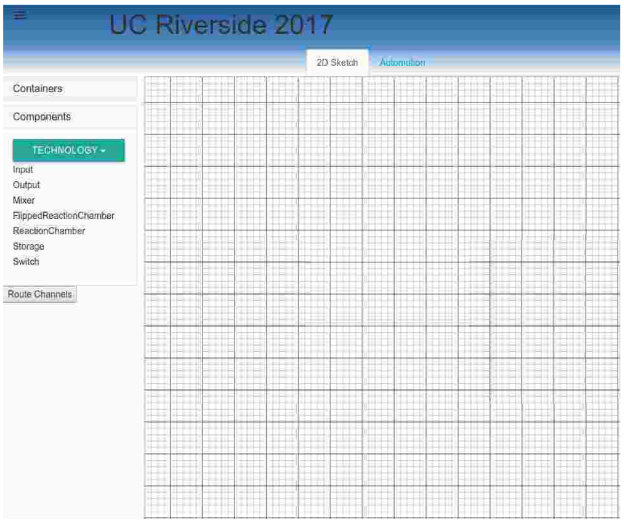


Fig. 4: Semi-automated design software initialized with available entities on the left hand side and an open drawing grid.

C. Control Layer Generation

Control layer design should also be semi-automated. As LoC designers do not start with a sequencing graph specification of a bioassay, existing control sharing techniques based on microvalve actuation sequences [9], [17]–[19] are impractical. Designers expect to make control sharing decisions manually. Using the graphical interface, the designer will click on a set of microvalves that will share a common control line. The designer may click on the location to place the control input, or she may allow the control input to be placed algorithmically. Control channels will then routed automatically, similar in principle to fluid channels. It is expected that the flow layer will be designed and validated before generating the control layer, but incremental approaches that design both layers at the same time can also be supported.

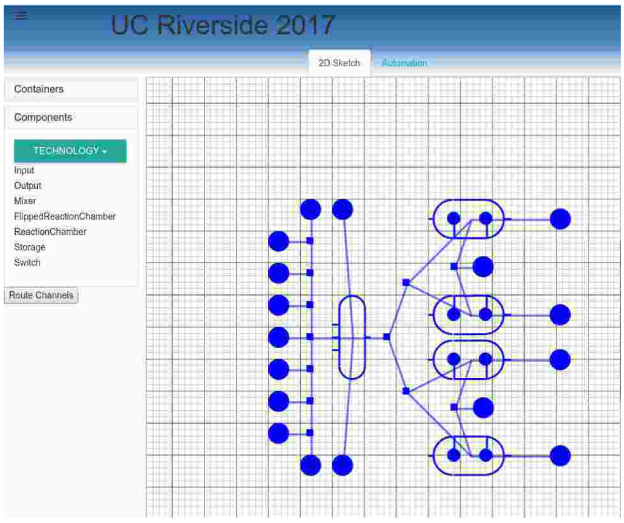


Fig. 5: The scientist has manually placed the components onto the drawing grid and specified their interconnections.

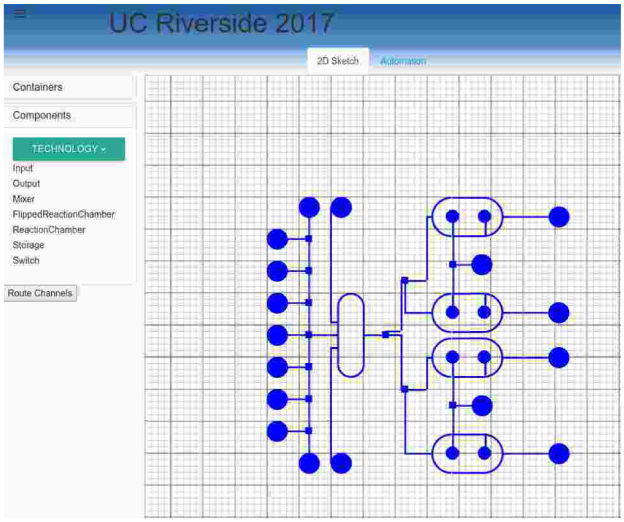


Fig. 6: The design automation algorithms have completed the routing of the channels and displayed the resultant device specification to the scientist.

D. Crossing the Bridge Between LoC Design and Manufacturing

Foundries typically employ computer-aided manufacturing (CAM) software as part of the LoC fabrication process. CAM software can accept a variety of well-established file formats, such as .svg, .stl, and file formats used by AutoCAD (.dwg and .dxf) and Solidworks (.slddrw). Among these options, the preferred choice is .svg, an open vector graphics format: .svg files can produce the masks required to fabricate mVLSI LoCs using multi-layer soft lithography (Fig. 1a); CAM software can convert .svg files to .gcode, which is used for the machined etching step required to produce monolithic membrane valves (Fig. 1b).

A secondary concern is design rule checking. Each foundry and mVLSI technology have their own design rules; in principle, design rule checking can be automated, but it is burdensome to support so many sets of seemingly ad-hoc design rules. At present, there is no standard file format for LoCs; however, this is an open issue presently being discussed by the Microfluidics Consortium[1]. If a standard device layout emerges that can be commonly used across foundries, then a standard design rule checker can easily be applied to all designs. Moreover, standardization can help with issues such as instrumentation and FDA approval in the USA.

VII. Related Work

To the best of our knowledge, the first paper on mVLSI CAD described an semi-automated AutoCAD plug-in that automates several tedious mVLSI design tasks [17]. The designer lays out the flow layer manually, and provides a device-specific instruction set which describes the fluidic transfer opertaions that will need to be performed. The software then automatically determines microvalve locations and generates an optimized control layer. This tool has two key drawbacks: (1) instruction set specification is tedious and subject to human error for large designs; and (2) the design methodology is completely flat and eschews the notion of reusable components. The semi-automated design flow that we propose here eliminates these drawbacks.

The Neptune project[2] at Boston University is an interactive tool for mVLSI design, which meets many of the criteria that we have described in this paper. A user designs the LoC using MINT, a microfluidic hardware design language [20]; MINT provides a very high degree of control over physical parameters, such as channel widths and microvalve sizes. The system automatically lays out the chip, generates a design schematic, and provides a warning list of design rules that may have been violated during layout. Neptune also provides assistance with building and controlling the device. For example, it provides a list of servo-syringe combinations to control the device along with .svg and 3D printing files that are needed for device fabrication. A graphical interface provides user control over the LoC; the user clicks on a graphical microvalve to actuate it, and the host PC sends commands to a microcontroller which, in turn, directly controls the servos. Neptune provides significant value to LoC designers and users while eschewing superfluous and unnecessary optimization.

VIII. Conclusion

In 2012, ∼85% of publications on microfluidics were published in engineering journals, ∼9% in biology and medicine journals, and ∼6% in interdisciplinary journals [21]; presumably, this survey did not include computer science/engineering conferences. On the one hand, this suggests that microfluidics and LoC technology may be a proverbial "solution in search of a problem." On the other hand, the survey favorably observed that many of the microfluidics papers published in biology and

[1] http://www.cfbi.com/microfluidics.htm
[2] Project URL: http://2016.igem.org/Team:BostonU_HW

medicine journals had multidisciplinary coauthors, including engineers, and recommended that "...microfluidic researchers should court collaborators from biology and clinical laboratories (and vice versa)." The survey concludes by noting that significant impact can truly be achieved when "the use of microfluidics introduces truly enabling functionality compared to current methods."

Within this context, mVLSI CAD software, fully or semi-automated, offers the potential to rapidly explore microfluidic solutions to new problems in biology or medicine. Optimization that incrementally improves performance or tries to reduce cost misses the point: microfluidic technologies have not yet been commoditized, and are highly distinguishable. The selling point of a given technology or LoC is its ability to reliably execute assays of importance to potential customers; cost is not (yet) a driving factor. To be impactful, research on mVLSI CAD should emphasize its utility to LoC designers and users, as opposed to optimization for its own sake.

The case for semi-automated mVLSI is simple: LoC designers experience pain points that optimization strategies derived from semiconductor VLSI do not address. The "design science" for mVLSI LoCs is immature, and professional designers are unwilling to cede control to algorithms. This may change in the future; however, the most appropriate short-term strategy is to integrate with existing mVLSI design practices and tools, rather than proposing fully automated replacements which potential customers view as being prohibitively risky.

Acknowledgments

This material is based upon work supported by the National Science Foundation under Grant Nos. 1351115, 1560596, and 1640757.

References

[1] M. A. Unger, H.-P. Chou, T. Thorsen, A. Scherer, and S. R. Quake, "Monolithic microfabricated valves and pumps by multilayer soft lithography," *Science*, vol. 288, no. 5463, pp. 113–116, 2000. [Online]. Available: http://dx.doi.org/10.1126/science.288.5463.113

[2] T. Thorsen, S. J. Maerkl, and S. R. Quake, "Microfluidic large-scale integration," *Science*, vol. 298, no. 5593, pp. 580–584, 2002. [Online]. Available: http://dx.doi.org/10.1126/science.1076996

[3] W. H. Grover, A. M. Skelley, C. N. Liu, E. T. Lagally, and R. A. Mathies, "Monolithic membrane valves and diaphragm pumps for practical large-scale integration into glass microfluidic devices," *Sensors and Actuators B: Chemical*, vol. 89, no. 3, pp. 315 – 323, 2003. [Online]. Available: http://dx.doi.org/10.1016/S0925-4005(02)00468-9

[4] W. H. Grover, R. H. C. Ivester, E. C. Jensen, and R. A. Mathies, "Development and multiplexed control of latching pneumatic valves using microfluidic logical structures." *Lab on a chip*, vol. 6, no. 5, pp. 623–31, may 2006. [Online]. Available: http://dx.doi.org/10.1039/B518362F

[5] I. E. Araci and S. R. Quake, "Microfluidic very large scale integration (mvlsi) with integrated micromechanical valves," *Lab Chip*, vol. 12, pp. 2803–2806, 2012. [Online]. Available: http://dx.doi.org/10.1039/C2LC40258K

[6] S. Blank and B. Dorf, *The Startup Owner's Manual: The Step-By-Step Guide for Building a Great Company*, 1st ed. K&S Ranch, 2012.

[7] Q. Wang, Y. Ru, H. Yao, T. Ho, and Y. Cai, "Sequence-pair-based placement and routing for flow-based microfluidic biochips," in *21st Asia and South Pacific Design Automation Conference (ASP-DAC)*, Macao, January 25-28, 2016, pp. 587–592. [Online]. Available: http://dx.doi.org/10.1109/ASPDAC.2016.7428075

[8] T. Tseng, M. Li, B. Li, T. Ho, and U. Schlichtmann, "Columba: co-layout synthesis for continuous-flow microfluidic biochips," in *Proceedings of the 53rd Annual Design Automation Conference (DAC)*, Austin, TX, USA, June 5-9, 2016, pp. 147:1–147:6. [Online]. Available: http://dx.doi.org/10.1145/2897937.2897997

[9] H. Yao, T. Ho, and Y. Cai, "PACOR: practical control-layer routing flow with length-matching constraint for flow-based microfluidic biochips," in *Proceedings of the 52nd Annual Design Automation Conference (DAC)*, San Francisco, CA, USA, June 7-11, 2015, pp. 142:1–142:6. [Online]. Available: http://dx.doi.org/10.1145/2744769.2744887

[10] J. McDaniel, C. Curtis, and P. Brisk, "Automatic synthesis of microfluidic large scale integration chips from a domain-specific language," in *Proceedings of the IEEE Biomedical Circuits and Systems Conference (BioCAS)*, Rotterdam, The Netherlands, October 31 - November 2, 2013, pp. 101–104. [Online]. Available: http://dx.doi.org/10.1109/BioCAS.2013.6679649

[11] W. H. Minhass, P. Pop, and J. Madsen, "System-level modeling and synthesis of flow-based microfluidic biochips," in *Proceedings of the 14th International Conference on Compilers, Architecture, and Synthesis for Embedded Systems (CASES)*, Taipei, Taiwan, October 9-14, 2011, pp. 225–234. [Online]. Available: http://dx.doi.org/10.1145/2038698.2038733

[12] W. H. Minhass, P. Pop, J. Madsen, and F. S. Blaga, "Architectural synthesis of flow-based microfluidic large-scale integration biochips," in *Proceedings of the 15th International Conference on Compilers, Architecture, and Synthesis for Embedded Systems (CASES)*, Tampere, Finland, October 7-12, 2012, pp. 181–190. [Online]. Available: http://dx.doi.org/10.1145/2380403.2380437

[13] A. R. Wu, J. B. Hiatt, R. Lu, J. L. Attema, N. a. Lobo, I. L. Weissman, M. F. Clarke, and S. R. Quake, "Automated microfluidic chromatin immunoprecipitation from 2,000 cells." *Lab on a chip*, vol. 9, no. 10, pp. 1365–70, may 2009. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/19417902

[14] J. McDaniel, A. Baez, B. Crites, A. Tammewar, and P. Brisk, "Design and verification tools for continuous fluid flow-based microfluidic devices," in *Proceedings of the 18th Asia and South Pacific Design Automation Conference (ASP-DAC)*, Yokohama, Japan, January 22-25, 2013, pp. 219–224. [Online]. Available: http://dx.doi.org/10.1109/ASPDAC.2013.6509599

[15] R. Sanka, H. Huang, R. Silva, and D. Densmore, "Mint - microfluidic netlist," in *Proceedings of the International Workshop on Bio-Design Automation (IWBDA), Poster Presentation*, Newcastle upon Tyne, UK, August 16-18, 2016. [Online]. Available: http://cidarlab.org/wp-content/uploads/2016/09/MINT-IWBDA-2016-Poster-Template-copy.pdf

[16] J. McDaniel, B. Crites, P. Brisk, and W. H. Grover, "Flow-layer physical design for microchips based on monolithic membrane valves," *IEEE Design & Test*, vol. 32, no. 6, pp. 51–59, 2015. [Online]. Available: http://dx.doi.org/10.1109/MDAT.2015.2459699

[17] N. Amin, W. Thies, and S. P. Amarasinghe, "Computer-aided design for microfluidic chips based on multilayer soft lithography," in *Proceedings of the 27th International Conference on Computer Design (ICCD)*, Lake Tahoe, CA, USA, October 4-7, 2009, pp. 2–9. [Online]. Available: http://dx.doi.org/10.1109/ICCD.2009.5413185

[18] W. H. Minhass, P. Pop, J. Madsen, and T. Ho, "Control synthesis for the flow-based microfluidic large-scale integration biochips," in *Proceedings of the 18th Asia and South Pacific Design Automation Conference (ASP-DAC)*, Yokohama, Japan, January 22-25, 2013, pp. 205–212. [Online]. Available: http://dx.doi.org/10.1109/ASPDAC.2013.6509597

[19] K. Hu, T. A. Dinh, T. Ho, and K. Chakrabarty, "Control-layer optimization for flow-based mvlsi microfluidic biochips," in *Proceedings of the 17th International Conference on Compilers, Architecture and S Synthesis for Embedded Systems (CASES)*, Uttar Pradesh, India, October 12-17, 2014, pp. 16:1–16:10. [Online]. Available: http://dx.doi.org/10.1145/2656106.2656118

[20] R. Sanka, H. Huang, R. Silva, and D. Densmore, "MINT - Microfluidic Netlist Instantiate the Device Defining the Control Layer," in *International Workshop on Biological Design Automation*, 2015, p. 13.

[21] E. K. Sackmann, A. L. Fulton, and D. J. Beebe, "The present and future role of microfluidics in biomedical research," *Nature*, vol. 507, no. 7491, pp. 181–189, 03 2014. [Online]. Available: http://dx.doi.org/10.1038/nature13118