Fault-Tolerant Valve-Based Microfluidic Routing Fabric for Droplet Barcoding in Single-Cell Analysis

Yasamin Moradi[†], Mohamed Ibrahim[‡], Krishnendu Chakrabarty[‡], and Ulf Schlichtmann[†]

[†]Chair for Electronic Design Automation, Technical University of Munich, Arcisstraße 21, 80333 München, Germany

[‡]Department of Electrical and Computer Engineering, Duke University, Durham, NC 27708, USA

***Abstract*—High-throughput single-cell genomics is used to gain insights into diseases such as cancer. Motivated by this important application, microfluidics has emerged as a key technology for developing comprehensive biochemical procedures for studying DNA, RNA, proteins, and many other cellular components. Recently, a hybrid microfluidic platform has been proposed to efficiently automate the analysis of a heterogeneous sequence of cells. In this design, a valve-based routing fabric based on transposers is used to label/barcode the target cells. However, the design proposed in prior work overlooked defects that are likely to occur during chip fabrication and system integration. We address the above limitation by investigating the fault tolerance of the valve-based routing fabric. We develop a theory of failure assessment and introduce a design technique for achieving fault tolerance. Simulation results show that the proposed method leads to a slight increase in the fabric size and decrease in cell-analysis throughput, but this is only a small price to pay for the added assurance of fault tolerance in the new design.**

I. Introduction

Single-cell analysis is used to gain insights into diseases such as cancer. This analysis is done in three steps, namely cell encapsulation and differentiation [1], droplet barcoding [2], and type-driven cell analysis [3], [4]. Recent developments in microfluidic techniques have paved the way for single-cell analysis on biochips [5]. These miniaturized platforms are based on either flow-based microfluidics [6] or digital microfluidic biochips (DMFBs) [7]. Despite the advent of affordable microfluidic technologies, each of these steps in single-cell analysis can only be carried out efficiently in a specific microfluidic technology domain [3], [8]. Therefore, to perform single-cell analysis efficiently on a single chip, a hybrid platform was recently introduced in [2]. This platform consists of components that work in different microfluidic domains. Each stage of the single-cell analysis protocol is carried out on a specific component.

In this work, we are focused on droplet barcoding and a valve-based fabric that is utilized as a crossbar to route a barcoding droplet from the reservoirs to the point where it is mixed with sample/reagents droplets [4]. This routing fabric is composed of transposers [9], which are connected using channels and controlled via pneumatic inputs. Fig. 1 illustrates the layouts of full and half transposers and shows models of these transposers. The straight and crossed lines in the model represent the connections between the input ports and the output ports. Fig. 2(a) presents an example of the routing fabric. It was shown that in this routing fabric, every input can be connected to every output, and it has a smaller footprint compared to the alternative of connecting reservoirs directly to the digital microfluidic part [2].

However, the fault tolerance of the crossbar has not been studied in prior work, and it remains a major concern for the design in [2]. Each transposer in this architecture is composed of channels and valves, which provide connections in the transposer. Recent studies have shown that physical defects can occur during the fabrication process, resulting in faulty channels and valves [10], [11]. Moreover, these basic components may also wear out and fail due to workload [12]. Therefore, connections in a transposer are prone to failures. When a connection along the path from an input to an output of the routing fabric fails, the path becomes disconnected and the barcoding droplet that is directed along this path will not arrive at the output port. Therefore, a sample in the downstream DMFB will not receive any barcoding droplet. As a result, when this sample is manipulated, there will be no information on its identity; hence no meaningful conclusion can be drawn from the analysis.

In this work, we address the fault tolerance of the routing fabric with respect to the failure of connections in transposers. We introduce a method for expanding the routing fabric to guarantee the existence of multiple paths through different transposers between each input/output pair. The main contributions of this paper are as follows:

- We define the concept of a critical transposer and present a method to identify critical transposers.
- We develop a theory of failure assessment and introduce a method for expanding critical transposers.
- We present a comprehensive performance evaluation and assessment of fault tolerance for the expanded crossbar.

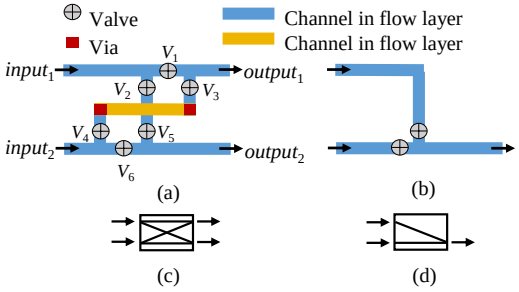

Fig. 1: (a) Layout of a full transposer; (b) layout of a half transposer; (c) model of a full transposer; (d) model of a half transposer.

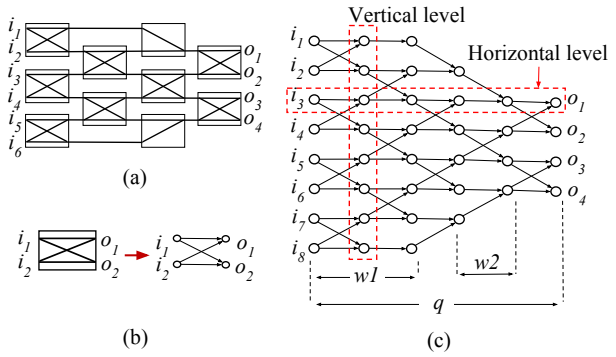

Fig. 2: (a) A 6-to-4 crossbar; (b) a 2-to-2 crossbar and its corresponding graph $\mathcal{F}_{2\times2}$; (c) graph model $\mathcal{F}_{8\times4}$ for an 8-to-4 crossbar.

II. Fault Model and Transposer Expansion

As illustrated in Fig. 1, each input is connected to both outputs of the full transposer through valves and channels. A half transposer has only one output but it can be connected to either input. We consider a fault model in which a combination of defective valves and channels block the flow of liquid in one path. Such a fault leads to the disconnection of an input/output pair inside a transposer. To guarantee the connectivity between inputs and outputs of a transposer with a faulty connection, the path for every input/output pair in the transposer must be replicated. By connecting a full transposer to the inputs of the target transposer, each input of the new transposer will be connected to each output of the target transposer via two paths. As shown in Fig. 3(a), one path goes through $output_{1,1}$ and the other one goes through $output_{1,2}$. For instance, two parallel paths from $input_{1,1}$ to $output_{2,1}$ are shown in different colors in Fig. 3(a). Fig. 3(b) shows a similar expansion of a half transposer.

III. Problem Formulation

When a droplet enters a transposer, it can be routed straight or crossed. This entry port of a transposer is referred to as a *decision point* [2]. A valve-based crossbar with n inputs and m outputs can be mapped to a directed acyclic graph (DAG) $\mathcal{F}_{n\times m} = (\mathcal{V}_{n\times m}, \mathcal{E}_{n\times m})$, where each decision point is mapped to a vertex $v \in \mathcal{V}_{n\times m}$, and each edge $e \in \mathcal{E}_{n\times m}$ represents the channel connecting two decision points. As an example, $\mathcal{F}_{2\times2}$ is a full transposer, as shown in Fig. 2(b). Fig. 2(c) shows the graph model corresponding to an 8-to-4 crossbar. It is shown in [2] that for $\mathcal{F}_{n\times m}$ to represent a fully connected crossbar, it is sufficient that n and m are even integers, and $\mathcal{F}_{n\times m}$ has q vertical levels, where $q = \frac{m+n}{2}$. Let w_1 and w_2 be defined as $w_1 = m - 1$ and $w_2 = (n - m)/2$. As shown in Fig. 2(c), n nodes in this design reside at each of the first w_1 vertical levels. This number is reduced in the subsequent w_2 vertical levels until it reaches m at the last level, which represents the output ports. We address each decision point in the crossbar with its coordinates. In other words, a node $p \in \mathcal{V}_{n\times m}$ is represented as (vertical level of p, horizontal level of p). An input i_k refers to the decision point $(1, k)$ in the crossbar, where $1 \le k \le n$. Similarly, an output o_j refers to the decision point $(q, j + \frac{n-m}{2})$, where $1 \le j \le m$.

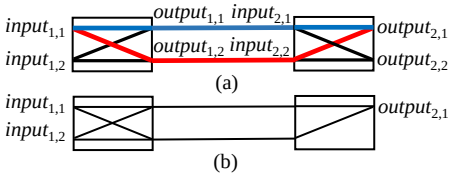

Fig. 3: (a) Expanded full transposer; (b) expanded half transposer.

The existence of a path from an input i_k to an output o_j depends on the functionality of the edges that connect the intermediate decision points. An edge whose failure (i.e. a connection failure in the transposer, also referred to in this paper as *edge failure*) results in the disconnection of an input/output pair is referred to as a *critical edge*. An edge can only be critical if it is a part of all the paths between an input/output pair. A transposer with a critical edge is referred to as a *critical transposer*. To maintain the full connectivity in the case of an edge failure, the crossbar should not have any critical edges.

Therefore, our goal is to first find input/output pairs that are connected through only one path or through multiple overlapping paths in a given crossbar. Following this, our objective is to identify critical transposers in the path between these input/output pairs. By expanding these critical transposers, an additional path for connecting the pair can be created that excludes the critical edges. Hence, in the redesigned crossbar, failure of any single edge can be tolerated.

IV. Identifying Critical Edges

To investigate the connectivity between an input/output pair, we divide the path between them into two sections: from the input to a node in vertical level w_1, and from the node in vertical level w_1 to the output. We are interested in vertical level w_1 since the number of nodes in the vertical levels is reduced from this level forward; see Fig. 2(c). Note that each node in $\mathcal{F}_{n\times m}$ is only connected to the nodes in the previous and next vertical level. In order to reach an output port from an input port, none of the intermediate vertical levels can be skipped. Therefore, any path between an input/output pair goes through a node at vertical level w_1.

If an input i of a transposer T is connected to a node p further downstream in the crossbar, any path from i to p goes through one of the outputs (o) of T. Because both inputs of any full transposer are connected to both of its outputs, the other input of T is also connected to p through o. A similar argument holds for the connection between the outputs of a transposer and nodes further upstream in the crossbar. Note that i_{k-1} and i_k are inputs of the same transposer if $k = 2K$ where $2 \le 2K \le n$. Similarly, o_{j-1} and o_j are outputs of the same transposer if $j = 2J$ where $2 \le 2J \le m$.

Let R_k be the set of nodes at vertical level w_1 that are connected to i_{k-1} and i_k, and let S_j be the set of nodes in this vertical level that are connected to o_{j-1} and o_j. Let us also define $P_{k,j}$ as the intersection of these two sets, i.e., $P_{k,j} = R_k \cap S_j$. For example, R_4, S_4, and their intersection in an 8-to-4 crossbar are shown in Fig. 4.

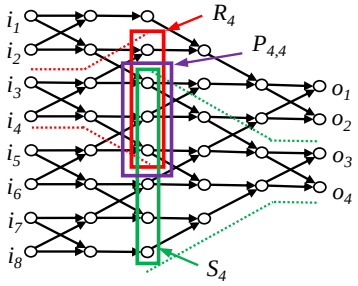

Fig. 4: The sets R_4, S_4, and $P_{4,4}$ in an 8-to-4 crossbar.

All the paths from i_{k-1} and i_k to o_{j-1} and o_j go through a node in $P_{k,j}$ while any other node at vertical level w_1 is either not connected to i_{k-1} and i_k or not connected to o_{j-1} and o_j. We present two key lemmas for determining the number of paths between an input/output pair based on the size of $P_{k,j}$. The proofs are not included in this short paper.

Lemma 1. *The set $P_{k,j}$ has only one element if and only if either $k = 2$ and $j = m$, or $k = n$ and $j = 2$. For other values of k and j, $P_{k,j}$ has three or more elements.*

Lemma 2. *If $P_{k,j}$ has only one element, then there is only one path for connecting each of the inputs i_{k-1} or i_k to each of the outputs o_{j-1} or o_j. If $P_{k,j}$ has more than one element, then there exist two non-overlapping paths from each of the inputs i_{k-1} or i_k to each of the outputs o_{j-1} or o_j.*

We refer to i_1 and i_2 as the first two inputs, i_{n-1} and i_n as the last two inputs, o_1 and o_2 as the first two outputs, and o_{m-1} and o_m as the last two outputs. Using Lemmas 1-2, we obtain the following theorem to identify the critical edges, and hence the critical transposers in a given crossbar.

Theorem 1. *The only critical edges in a given crossbar are those that are used to connect each of the first two inputs to each of the last two outputs, or each of the last two inputs to each of the first two outputs.*

Using Theorem 1, the critical edges (and hence the critical transposers), can be easily identified in a given crossbar. As shown in Fig 5, these transposers are positioned in two diagonal paths in the crossbar. By expanding the critical transposers, we create alternative paths for connecting the input/output pairs that previously were connected only through one path. For example, Fig. 5 shows the corresponding graph for an 8-to-4 fault-tolerant crossbar. The edges that represent additional transposers are shown in red.

V. Evaluation and Simulation Results

Our goal in this section is to evaluate the expanded routing fabric in terms of fault tolerance and transposer overhead. We also compare the quality of the fault-tolerant routing fabric with the original design reported in [2]. We use an enhancement of the CoSyn platform (We refer to it as CoSyn+) for our simulations, and we apply CoSyn+ to the routing fabric of [2] and the proposed fault-tolerant routing fabric. The enhancements to CoSyn were implemented using C++. The set of bioassays constituting the single-cell analysis protocol [2] were used as a benchmark, and the cells were classified using a uniform distribution function.

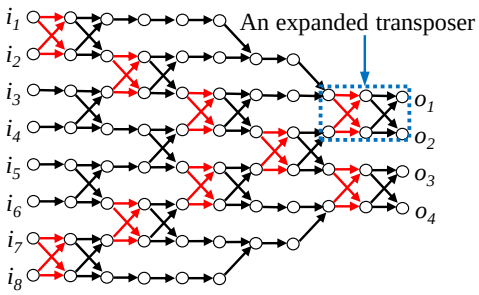

Fig. 5: The graph model of an expanded 8-to-4 crossbar.

A. Fault-tolerance Assessment

We first investigate the reliability of the expanded crossbar. Recall that the routing fabric designed in this work is only guaranteed to tolerate a single fault. We show however that this design is also resilient in most cases to multiple faults. To investigate the probability of a disconnected input/output pair in the case of more than one edge failure, we simulate multiple edge failures by removing random edges in the crossbar. The crossbar is reliable if none of the input/output pairs get disconnected in the case of multiple faults.

First, we inject 10000 random double faults in five crossbars of different sizes and record the number e of instances of double faults for which the crossbar gets disconnected. The percentage of injected faults that are benign (i.e., they do not cause any disconnections) is computed as $\frac{(10000-e)\times 100}{10000}$. We repeat the experiment 20 times for each crossbar and record the mean percentage of benign multiple faults and the standard deviation among these experiments, which are shown in Fig. 6. We then carry out the same experiment by injecting more faults at a time. As shown in Fig. 6, the redesigned crossbar is more than 97% reliable in the case of double faults, more than 96% reliable in the case of triple faults, and more than 95% reliable in the case of quadruple faults. As expected, the reliability of the crossbar decreases when more faults occur, but our results show that the fault tolerance remains high for a large multiplicity of faults. Note also that the reliability is higher for larger crossbars. This is expected since the number of edges is higher in a larger crossbar, and therefore, the probability that the faults happen on a group of edges that can disconnect an input/output pair is lower.

B. Transposer Overhead

Next we calculate the cost of fault tolerance in terms of the transposer overhead in the redesigned crossbar. The number of additional transposers is equal to the number of transposers in the two critical paths. Since one transposer is common to the two paths (where diagonal paths cross each other), the total number of additional transposers is $2q-3$. Table I shows the transposer overhead for various values of n and m. The overhead is less for larger routing fabrics since at each vertical level the number of transposers increases with the number of inputs and outputs, but the overhead stays constant.

TABLE I: Overhead introduced by the fault-tolerant routing fabric

$n \times m$	20×4	20×8	20×16	40×4	40×8	40×16	40×20	40×36
# Transposers in the original routing fabric	73	106	160	248	321	455	516	720
Transposer overhead	21 (28%)	25 (23%)	33 (20%)	41 (16%)	45 (14%)	53 (12%)	57 (11%)	73 (10%)

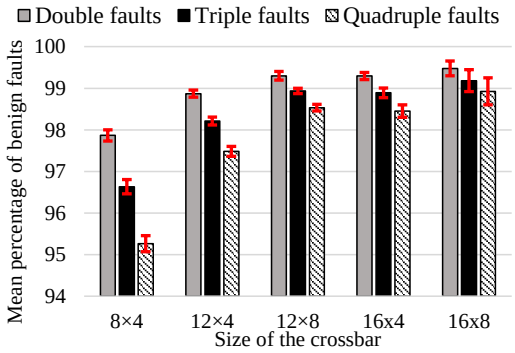

Fig. 6: Mean percentage of benign multiple faults.

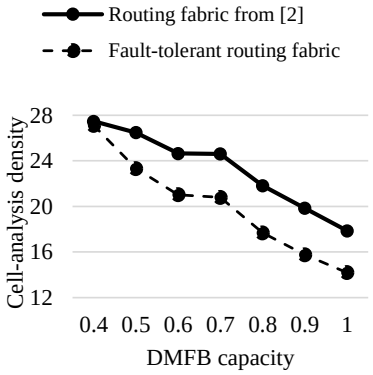

Fig. 7: Cell-analysis density using 20 barcoding outputs.

C. Design-Quality Assessment

We also investigate the quality of a platform using the fault-tolerant routing fabric by comparing it to the quality of a platform using the original routing fabric. For this purpose, we evaluate the number of single-cell experiments that can be carried out on the two platforms in a given window of time and by using a given number of resources in the other parts of the chips, i.e., DMFB resources. We use the terms *cell-analysis density* and *DMFB capacity* as they were defined in [2].The cell-analysis density refers to the number of cells that can be analyzed in a specific window of time when a given set of DMFB resources is available. We calculate the cell-analysis density for a time window of one minute and a set of DMFB resources of size 100 electrodes. The DMFB capacity represents the fraction of input cells that can be processed simultaneously on a given set of DMFB resources. Therefore, DMFB capacity changes when the number of DMFB resources are varied.

Our objective here is to compare the cell-analysis density of two platforms using the original routing fabric and the fault-tolerant routing fabric, respectively. We simulate the analysis of 50 input cells of 40 different types using 20 barcoding outputs for various numbers of DMFB resources. As shown in Fig. 7, cell-analysis density decreases when more DMFB resources are available, i.e., more resources are used to process the same number of input cells. In general, cell-analysis density is less for the fault-tolerant routing fabric since the total completion time for the same number of input cells is higher and fewer cells finish analysis in a given window of time. Nevertheless, the slight decrease in the cell-analysis density is only a small price to pay for the added assurance of fault tolerance in the new design.

VI. Conclusion

We have introduced an analysis method for identifying critical transposers in a routing fabric and a design technique for expanding critical transposers to ensure single-fault tolerance. We have also shown that the redesigned routing fabric provides a high degree of fault tolerance for multiple faults. The design was evaluated in terms of transposer overhead and the cell throughput for limited DMFB resources. As part of future work, we will investigate the possibility of reducing the overhead by using other architectures for the fault-tolerant crossbar.

Acknowledgment

This research was supported partly by Technical University of Munich – Institute for Advanced Study, funded by the German Excellence Initiative and the European Union Seventh Framework Programme under grant agreement N^{o} 291763, as well as by the IGSSE Project FLUIDA of Technical University of Munich.

References

[1] J. Chen *et al.*, "Microfluidic impedance flow cytometry enabling high-throughput single-cell electrical property characterization," *International Journal of Molecular Sciences*, vol. 16, no. 5, pp. 9804–9830, 2015.

[2] M. Ibrahim *et al.*, "CoSyn: Efficient single-cell analysis using a hybrid microfluidic platform," in *Proc. IEEE/ACM Design, Automation, and Test in Europe Conference (DATE)*, 2017, pp. 1673–1678.

[3] S. Hosic *et al.*, "Microfluidic sample preparation for single cell analysis," *Analytical Chemistry*, vol. 88, no. 1, pp. 354–380, 2015.

[4] A. M. Klein *et al.*, "Droplet barcoding for single-cell transcriptomics applied to embryonic stem cells," *Cell*, pp. 1187–1201, 2015.

[5] A. Saadatpour *et al.*, "Single-cell analysis in cancer genomics," *Trends in Genetics*, vol. 31, no. 10, pp. 576–586, 2015.

[6] T. Thorsen *et al.*, "Microfluidic large-scale integration," *Science*, vol. 298, no. 5593, pp. 580–584, 2002.

[7] R. B. Fair, "Digital microfluidics: is a true lab-on-a-chip possible?" *Microfluidics and Nanofluidics*, vol. 3, no. 3, pp. 245–281, 2007.

[8] A. Rival *et al.*, "An EWOD-based microfluidic chip for single-cell isolation, mRNA purification and subsequent multiplex qPCR," *Lab on a Chip*, vol. 14, no. 19, pp. 3739–3749, 2014.

[9] R. Silva *et al.*, "A reconfigurable continuous-flow fluidic routing fabric using a modular, scalable primitive," *Lab on a Chip*, 2016.

[10] K. Hu *et al.*, "Fault modeling, testing, and design for testability," in *Computer-Aided Design of Microfluidic Very Large Scale Integration (mVLSI) Biochips*. Springer, 2017, pp. 81–115.

[11] P. Pop *et al.*, "Continuous-flow biochips: Technology, physical-design methods, and testing," *IEEE Design & Test*, vol. 32, no. 6, pp. 8–19, 2015.

[12] W. H. Minhass *et al.*, "System-level modeling and simulation of the cell culture microfluidic biochip procell," in *IEEE Symposium on Design Test Integration and Packaging of MEMS/MOEMS (DTIP)*, 2010, pp. 91–98.