

Integrated Droplet Routing and Defect Tolerance in the Synthesis of Digital Microfluidic Biochips

TAO XU and KRISHNENDU CHAKRABARTY
Duke University

Microfluidic biochips are revolutionizing high-throughput DNA sequencing, immunoassays, and clinical diagnostics. As high-throughput bioassays are mapped to digital microfluidic platforms, the need for design automation techniques is being increasingly felt. Moreover, as most applications of biochips are safety-critical in nature, defect tolerance is an essential system attribute. Several synthesis tools have recently been proposed for the automated design of biochips from the specifications of laboratory protocols. However, only a few of these tools address the problem of defect tolerance. In addition, most of these methods do not consider the problem of droplet routing in microfluidic arrays. These methods typically rely on postsynthesis droplet routing to implement biochemical protocols. Such an approach is not only time consuming, but also imposes an undue burden on the chip user. Postsynthesis droplet routing does not guarantee that feasible droplet pathways can be found for area-constrained biochip layouts; nonroutable fabricated biochips must be discarded. We present a synthesis tool that integrates defect tolerance and droplet routing in the design flow. Droplet routability, defined as the ease with which droplet pathways can be determined, is estimated and integrated in the synthesis procedure. Presynthesis and postsynthesis defect-tolerance methods are also presented. We use a large-scale protein assay as a case study to evaluate the proposed synthesis method.

Categories and Subject Descriptors: B.7.2 [**Integrated Circuits**]: Design Aids—*Placement and routing*; B.8.1 [**Performance and Reliability**]: Reliability, Testing and Fault-Tolerance; J.3 [**Life and Medical Sciences**]: *Biology and genetics, Health*

General Terms: Algorithms, Design, Performance, Reliability

Additional Key Words and Phrases: Physical design automation, module placement, microfluidics, biochips

ACM Reference Format:

Xu, T. and Chakrabarty, K. 2008. Integrated droplet routing and defect tolerance in the synthesis of digital microfluidic biochips. *ACM J. Emerg. Technol. Comput. Syst.* 4, 3, Article 11 (August 2008), 24 pages. DOI = 10.1145/1389089.1389091 <http://doi.acm.org/10.1145/1389089.1389091>

This research was supported by the National Science Foundation under grants IIS-0312352 and CCF-0541055.

Authors' addresses: T. Xu, K. Chakrabarty, Department of Electrical and Computer Engineering, Duke University, Durham, NC 27708; email: tx@ee.duke.edu; krish@ee.duke.edu.

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or direct commercial advantage and that copies show this notice on the first page or initial screen of a display along with the full citation. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, to republish, to post on servers, to redistribute to lists, or to use any component of this work in other works requires prior specific permission and/or a fee. Permissions may be requested from Publications Dept., ACM, Inc., 2 Penn Plaza, Suite 701, New York, NY 10121-0701 USA, fax +1 (212) 869-0481, or permissions@acm.org. © 2008 ACM 1550-4832/2008/08-ART11 \$5.00. DOI 10.1145/1389089.1389091 <http://doi.acm.org/10.1145/1389089.1389091>

ACM Journal on Emerging Technologies in Computing Systems, Vol. 4, No. 3, Article 11, Pub. date: August 2008.

1. INTRODUCTION

Microfluidics-based biochips combine electronics with biology to open new application areas such as point-of-care medical diagnostics, on-chip DNA analysis, and automated drug discovery [Schulte et al. 2002; Srinivasan et al. 2004; Fair et al. 2006]. Bioassays are mapped to microfluidic arrays using synthesis tools, and are executed through the electronic manipulation of sample and reagent droplets. Prototype biochips have been developed and reported in the literature for enzymatic assays to detect glucose and lactate levels in blood [Guisseppi-Elie et al. 2005], detection of explosives such as TNT [Pamula et al. 2005], real-time monitoring of airborne particulate matter [Fair et al. 2004], synthesis-based gene sequencing [Deamer and Akeson 2000], and protein crystallization [Van der Woerd et al. 2003].

A popular class of microfluidic biochips is based on continuous fluid flow in permanently etched microchannels. Fluid flow in these devices is controlled either using micropumps and microvalves or by electrical methods based on electrokinetics and electroosmosis [Verpoote and Rooij 2003]. An alternative category of microfluidic biochips relies on the principle of electrowetting-on-dielectric. Discrete droplets of nanoliter volume can be manipulated using an array of electrodes. Following the analogy of digital electronics, this technology is referred to as “digital microfluidics” [Fair et al. 2003]. Because each droplet can be controlled independently, these systems also have dynamic reconfigurability, whereby groups of unit cells in a microfluidic array can be reconfigured to change their functionality during the concurrent execution of a set of bioassays.

As more bioassays and biomedical procedures are mapped to a microfluidic platform for concurrent execution, system functionality and design complexity are expected to increase steadily. Next-generation biochips are likely to be multifunctional and adaptive “biochemical processing” devices. For example, inexpensive biochips for clinical diagnostics offer high throughput with low sample volumes, and they integrate hematology, pathology, molecular diagnostics, cytology, microbiology, and serology onto the same platform. The emergence of such integrated and multifunctional platforms provides the electronic design automation community with a new application driver and market for research into new algorithms and design tools.

Over the past few years, several automated synthesis tools have recently been proposed for microfluidic biochips. These design automation methods address operation scheduling and module placement for digital microfluidics [Su and Chakrabarty 2006; 2005; 2004; Hwang et al. 2006; Yuh et al. 2006; Ricketts et al. 2006], as well the design and optimization of routing channels in continuous-flow microfluidics [Pfeiffer et al. 2006]. However, only a few of these tools address the problem of droplet routing in digital microfluidic arrays. These methods typically rely on postsynthesis droplet routing to implement biochemical protocols; such an approach is not only time consuming, but also imposes an undue burden on the chip user. Moreover, postsynthesis droplet routing does not guarantee that feasible droplet pathways can be found for area-constrained biochip layouts; nonroutable fabricated biochips must be discarded. Synthesis tools for digital microfluidics simply assume that the droplet-routing time is

negligible compared to microfluidic operation times, and that the droplet pathways can be trivially determined [Su and Chakrabarty 2004; Su et al. 2006; Yuh et al. 2006; Ricketts et al. 2006].

However, as in the case of integrated circuits (ICs), routing complexity increases with an increase in the complexity and number of bioassay operations that are mapped to a digital microfluidics platform. The reconfigurability inherent in digital microfluidics leads to a large number of candidate droplet pathways, all of which must be evaluated to determine the best synthesis solutions. Moreover, unlike permanently etched microchannels in continuous-flow microfluidic chips, the droplet pathways in digital microfluidics are reconfigurable and can be activated or deactivated dynamically.

More recently, some methods for postsynthesis droplet routing have been proposed [Böhringer 2006; Griffith et al. 2006; Su et al. 2006], but these methods suffer from the need to explicitly evaluate a large number of candidate routes. In particular, Su et al. [2006] use a modified version of the Lee routing algorithm from VLSI design [Sait 1995; Sechen 1998], which requires the enumeration of candidate routes. Moreover, while recent advances in digital microfluidics have significantly reduced microfluidic operation times [Paik et al. 2003], droplet-routing time has not decreased proportionately. Similar to the increased importance of interconnect length in integrated circuits, droplet-pathway lengths are now an important consideration for biochip system synthesis. Therefore, to reduce droplet-routing complexity and to obtain synthesis results that best predict bioassay operation times, droplet routing needs to be integrated in the synthesis flow.

Increase in density and area of microfluidics-based biochips also leads to high defect density, especially for new technology nodes. A comprehensive defect-model library for digital microfluidics has been presented in Xu and Chakrabarty [2007]. Defect types from different energy domains (electrical, fluidic, mechanical) in such mixed-technology devices have been extracted based on experiments using prototype chips; example defects include dielectric breakdown, electrical opens, coating failure, sample residues and protein adsorption, and particle contamination. These defects lead to reduced yield. Low yield will deter large-scale and high-volume production, and will increase production cost. Therefore, defect-tolerant designs are important for the emerging marketplace, especially for low-cost, portable, and disposable devices for clinical diagnostics.

Another reason for the importance of defect tolerance lies in the projected use of microfluidic biochips for safety-critical applications, such as patient health monitoring, neonatal care, and the monitoring of environmental toxins. Therefore, defect tolerance must be integrated into the automated design tools to ensure high levels of system dependability. Despite the recent emergence of automated synthesis methods for biochips [Su and Chakrabarty 2006; 2005; 2004; Hwang et al. 2006; Yuh et al. 2006; Ricketts 2006], defect tolerance has largely been overlooked in the literature.

In this article, we propose a unified synthesis method that combines defect-tolerant architectural synthesis with droplet-routing-aware physical design. Droplet routability, defined as the ease with which droplet pathways can be determined, is estimated and integrated in the synthesis flow. The proposed

approach allows architectural-level design choices and droplet-routing-aware physical design decisions to be made simultaneously. Presynthesis and postsynthesis defect tolerance are also incorporated in the synthesis tool. We use a large-scale protein assay as a case study to evaluate the proposed synthesis method.

The rest of the article is organized as follows. Section 2 provides an overview of digital microfluidic biochips. In Section 3 we discuss related prior work on automated synthesis tools and postsynthesis droplet routing for digital microfluidic biochips. In Section 4, we introduce a new criterion for evaluating droplet routability for a synthesized design, and incorporate it into the overall synthesis flow. Section 5 presents pre- and postsynthesis defect-tolerance methods, and integrates them with the droplet-routing-aware synthesis flow. In Section 6, we use a large-scale protein assay to evaluate the proposed synthesis method. Finally, conclusions are drawn in Section 6.

2. DIGITAL MICROFLUIDIC BIOCHIPS

The microfluidic biochips discussed in this work are based on the manipulation of nanoliter droplets on a two-dimensional electrode array, using the principle of electrowetting [Fair et al. 2003]. A unit cell in the array includes a pair of electrodes that acts as two parallel plates. The bottom plate contains a patterned array of individually controlled electrodes, and the top plate is coated with a continuous-ground electrode. A droplet rests on a hydrophobic surface over an electrode. It is moved by applying a control voltage to an electrode adjacent to the droplet and, at the same time, deactivating the electrode just under the droplet. This electronic method of wettability control creates interfacial tension gradients that move the droplets to the charged electrode. Using the electrowetting phenomenon, droplets can be moved to any location on a two-dimensional array. By varying the patterns of control voltage activation, many fluid-handling operations such as droplet merging, splitting, mixing, and dispensing can be executed in a similar manner. For example, mixing can be performed by routing two droplets to the same location and then turning them about some pivot points [Paik et al. 2003]. The digital microfluidic platform offers the additional advantage of flexibility, referred to as reconfigurability, since fluidic operations can be performed anywhere on the array. Droplet routes and operation-scheduling results are programmed into a microcontroller that drives electrodes in the array. In addition to electrodes, optical detectors such as LEDs and photodiodes are also integrated in microfluidic arrays to monitor colorimetric bioassays [Zeng and Korsmeyer 2004]; see Figure 1.

3. RELATED PRIOR WORK

Recent years have seen growing interest in the automated design and synthesis of microfluidic biochips [Su and Chakrabarty 2006; 2005; 2004; Hwang et al. 2006; Yuh et al. 2006; Ricketts 2006; Pfeiffer et al. 2006]; see Figure 1. One of the first published methods for biochip synthesis decouples high-level synthesis from physical design [Su and Chakrabarty 2004]. It is based on rough estimates for placement costs such as the areas of the microfluidic modules.

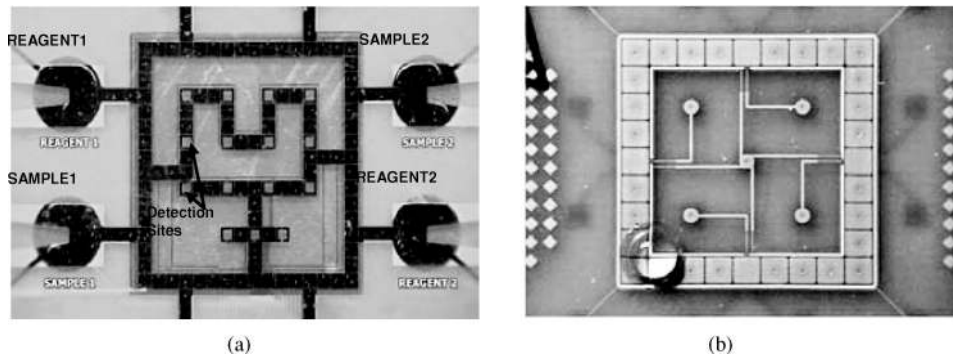


Fig. 1. Fabricated digital microfluidic arrays: (a) glass substrate [Srinivasan et al. 2004]; (b) PCB substrate [Paik et al. 2007].

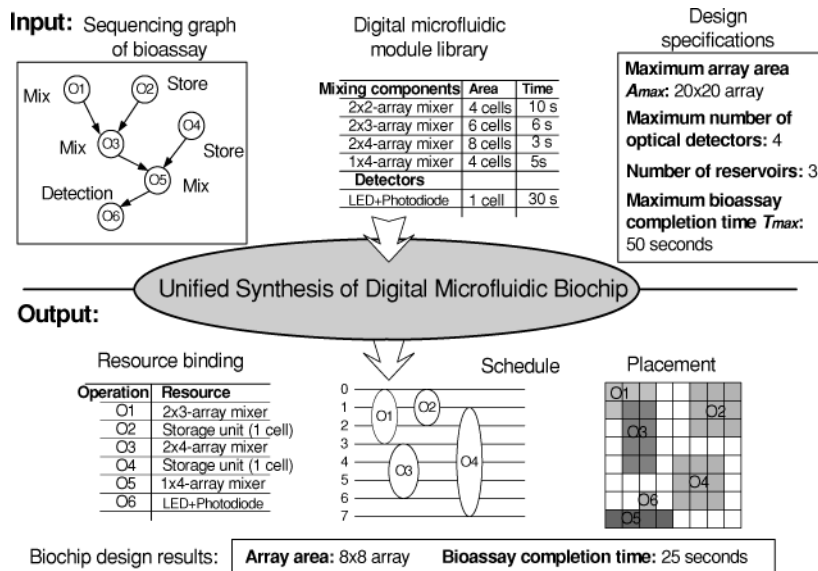


Fig. 2. An example illustrating system-level synthesis [Su and Chakrabarty 2005].

These estimates provide lower bounds on the exact biochip area, since the overheads due to spare cells and cells used for droplet transportation are not known a priori. However, it cannot be accurately predicted if the biochip design meets system specifications, for example, maximum allowable array area and upper limits on assay completion times, until both high-level synthesis and physical design are carried out. Su and Chakrabarty [2005] proposed a unified system-level synthesis method for microfluidic biochips based on parallel recombinative simulated annealing (PRSA), which offers a link between these two steps. This method allows users to describe bioassays at a high level of abstraction, and it automatically maps behavioral descriptions to the underlying microfluidic array.

The design flow is illustrated in Figure 2. First, the different bioassay operations (e.g., mixing and dilution) and their mutual dependences are represented

using a sequencing graph. Next, a combination of simulated annealing and genetic algorithms are used for unified resource binding, operation scheduling, and module placement. A chromosome is used to represent each candidate solution, namely, a design point. In each chromosome, operations are randomly bound to resources. Based on the binding results, list scheduling is used to determine the start times of operations; that is, each operation starts with a random latency after its scheduled time. Finally, a module placement is derived based on the resource binding and the schedule of fluidic operations. A weighted sum of area- and time cost is used to evaluate the quality of the design. The design is improved through a series of genetic evolutions based on PRSA. It generates an optimized schedule of bioassay operations, the binding of assay operations to resources, and a layout of the microfluidic biochip.

The aforesaid top-down synthesis flow unifies architecture-level design with physical-level module placement. However, it suffers from two drawbacks. For operation scheduling, it is assumed that the time cost for droplet routing is negligible, which implies that droplet routing has no influence on the operation completion time. While generating physical layouts, the synthesis tool in Su and Chakrabarty [2005] provides only the layouts of the modules, and leaves droplet-routing pathways unspecified. The assumption of negligible droplet transportation times as valid works for small microfluidic arrays. However, for large arrays and for biochemical protocols that require several concurrent fluidic operations on-chip, the droplet transportation time and routing complexity are nontrivial.

Recent work on automated biochip design has also included postsynthesis droplet routing [Böhringer 2006; Su et al. 2006]. These methods can reduce droplet transportation time by finding optimal routing plans for a synthesized biochip. However, the effectiveness of such methods is limited by the synthesis results; that is, the placement of microfluidic modules often determines those droplet pathways that lead to minimum droplet transportation time. For example, if we need to route a droplet between two modules that are 10 electrodes away from each other, then it is not possible to reduce the droplet transportation time to less than that needed to move a droplet by a distance equal to 10 electrodes. Since droplet pathways are dynamically reconfigurable, the number of feasible droplet pathways can be very high, leading to considerable computation time for a droplet-routing tool.

It is therefore necessary to synthesize biochip designs that make droplet routing easier. Moreover, due to advances in microfluidic module design (smaller feature sizes, improved materials, etc.), the fluidic operation times are decreasing steadily [Paik et al. 2003]. However, the droplet transportation times are not decreasing at the same pace. As a result, routing times must be considered during operation scheduling and in the calculation of assay completion times. For the synthesis results derived from the methods proposed in Su and Chakrabarty [2005], the impact of droplet routing on assay completion time might be significant, and the upper limit on assay completion time might be violated. In such scenarios, the biochip design will no longer correctly implement the desired biochemical procedures. Also, if a synthesized design is not routable, either the chip must be discarded or time-consuming resynthesis must

be carried out. To avoid such occurrences, we have to anticipate the availability of routing paths during synthesis. Therefore, droplet routing must be included in the synthesis flow for digital microfluidics.

The other drawback of the synthesis flow described earlier is that it is defect-oblivious. It can neither guarantee that the design is robust, that is, defect tolerant, nor facilitate reconfiguration techniques that can be used to bypass defects. Therefore, defective chips must be discarded if errors are observed during testing or assay operation. The lack of defect tolerance leads to reduced yield and higher chip cost in an extremely cost-sensitive market. Therefore, defect tolerance needs to be integrated with droplet routing and biochip synthesis.

4. ROUTING-AWARE SYNTHESIS

In this section, we describe how we can incorporate droplet routing in the synthesis flow. Droplet-routing methods can be viewed as being either anticipatory, that is, they anticipate the routability (defined qualitatively as the ease of droplet routing) of the synthesized biochip and design the system to be easily routable, or based on postsynthesis routing to find the efficient droplet pathways.

The focus of this article is on droplet-routing-aware design, whereby we attempt to provide a guaranteed level of routability for every interdependent module pair (we will discuss this aspect in more detail in Section 4.1). Instead of finding efficient droplet pathways after synthesis, we attempt to achieve high-routability mapping of bioassay protocols to the microfluidic array. We next propose a new method to incorporate droplet routing in the PRSA-based synthesis flow for defect-tolerant microfluidic biochips, developed in Su and Chakrabarty [2005].

4.1 Droplet-Routability Estimation

For a synthesized biochip, the droplet-routability of a route between two modules is quantified in terms of the length, measured by number of electrodes, of the droplet transportation path. Droplet routability is evaluated in terms of the average length of all the droplet pathways for the complete chip. Also, we have to control the maximum length of droplet paths. Large values for the maximum path length lead to long routing times, for example, more than 5% of the module operation time, which can have the undesirable consequence of having to halt an assay temporarily until the droplets are routed to their destinations. Moreover, long routing pathways are liable to be blocked by obstacles, namely, intermediate modules. For example, in Figure 3, all routing pathways from M_1 to M_4 are blocked by M_2 and M_3 ; therefore droplet routing is not feasible for this design. Note that guard-ring cells are used to avoid inadvertent mixing and cannot be used for routing. Synthesized designs with large values for the maximum droplet-path length suffer from a high probability of being nonroutable. Based on the aforementioned considerations, we adopt the maximum droplet-path length as a parameter for evaluating routability of a synthesized biochip.

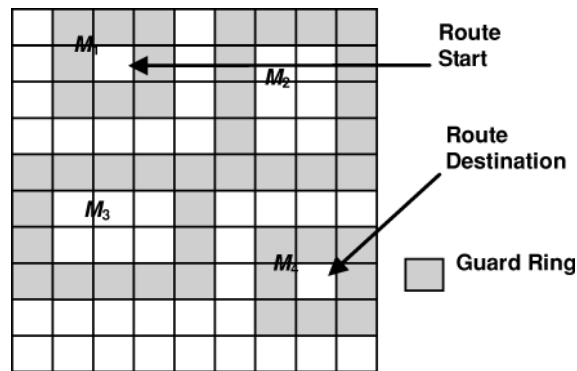


Fig. 3. An example of a nonroutable interdependent pair.

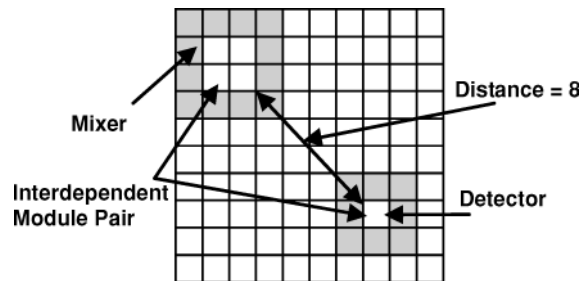


Fig. 4. Illustration of module distance.

A straightforward technique to derive the routability information is to carry out postsynthesis routing to generate an actual routing plan. However, this approach adds to the computational burden of the synthesis tool. In particular, if a routing plan involving all droplets on the array is generated for each chromosome in the PRSA-based unified synthesis method, the overall synthesis time will be overwhelming, due to the large number of chromosomes and evolution steps in the synthesis flow. Moreover, since we only care about the final synthesis result, we need to reduce the effort spent to generate route plans for the intermediate designs. Therefore, we adopt simple estimates of routability, instead of precisely calculating droplet routes at each step. The module distance M_{ij} is defined as the length of the shortest path between two interdependent modules M_i and M_j , assuming no obstacles between them; see Figure 4. By interdependent module, we refer to module pairs where the operation of one module depends on the operation of the other. For example, if optical detection is to be carried out for a mixed droplet, then the optical detector and the mixer are interdependent. Note that in many cases, two interdependent modules may not be able to operate in successive time-steps; for example, a mixed droplet may have to wait for a few cycles, since the detector may be busy processing another detection step when the mixing is finished. In such cases, a storage unit is needed, and we consider the storage unit and the detector as interdependent modules, as are the mixer and the storage unit. Thus droplets are routed only between interdependent modules. The module distance is calculated for each

```

PRSA-Based Droplet-Routing-Aware Synthesis Procedure
1 Set initial population of chromosome and the initial temperature  $T_{\infty}$ ;
2 Implement the synthesis using the information of initial chromosomes:
  {Phase I: Resource binding; Phase II: Scheduling; Phase III: Placement
  Phase IV: Routability estimation}
3 while (Stopping criteria of annealing is not satisfied)
4   for  $i = 1: N$  /* Inner loop of annealing process */
5     Find fitness values of chromosomes through construction procedure;
6     Reproduction; /* Best chromosomes copied to the next generation */
7     Crossover: {Parameterized uniform crossover is to generate the child
      chromosome from two randomly-selected parent chromosomes}
8     if  $\text{Fitness}(\text{child}) < \text{Fitness}(\text{parents})$  or
       $\text{rand}(0,1) < \exp(-[\text{Fitness}(\text{child}) - \text{Fitness}(\text{parents})]/T)$ 
9       Child chromosome is selected;
10      else Parent chromosome (the best one) is selected;
11      end if /* Here a Boltzmann trial is performed */
12      Mutation; /* New chromosomes are generated randomly */
13      New population replaces the old generation; end for
14     $T = \text{rate} \times T$ ; /* update the temperature */ end while
15  Find the best chromosome from the final population;
16  Output the results of resource binding, scheduling and placement.

```

Fig. 5. Pseudocode for the PRSA-based droplet-routing-aware synthesis procedure.

interdependent module pair. Although the module distance M_{ij} may not be exactly the same as the shortest path length (especially if there are obstacles in the form of other modules on the array), M_{ij} is still a good estimate of routability between M_i and M_j .

Note that in some scenarios, the location of two interdependent modules may overlap on the array. In such cases, we set the corresponding module distance to be zero. Since our goal is to guarantee the routability of modules in the synthesized biochip, we adopt the average module distance (over all interdependent modules) as a design metric. Similarly, we adopt maximum module distance to approximate the maximum length of droplet manipulation and use it for routability estimation. For each chromosome considered in the PRSA-based synthesis flow, we calculate the average and maximum module distances.

Next we incorporate routability in the PRSA-based unified synthesis method. Synthesis results with high routability values are more likely to lead to simple and efficient droplet pathways. To find such designs, we incorporate the afore-said two metrics into the fitness function by a factor that can be fine tuned according to different design specifications to control the PRSA-based procedure. The pseudocode for the droplet-routing-aware unified synthesis method is shown in Figure 5. Candidate designs with low routability are discarded during evolution. Thus, the synthesis procedure guarantees that the routing complexity is reduced for the synthesized biochip, while meeting constraints on array size, bioassay processing time, and defect tolerance [Su and Chakrabarty 2005].

4.2 Routing-Time Cost and Assay Completion Time

Next we discuss the impact of routing-time cost on bioassay completion time. Here we use the route-planning method of Su et al. [2006] to find an efficient

route plan for each interdependent pair. The time cost due to the need for droplet transportation is calculated and added to the operation time for the first module in the interdependent module pair. Next the schedule is adjusted based on the modified operation time.

There are two possible scenarios that can arise when the schedule is adjusted. In the first, despite the increased operation time, the fluidic operation can be accommodated in its designated time interval due to the availability of slack or unoccupied time slots in the schedule. In this case, the schedule can simply rely on the available slack or unused time interval for droplet routing. In the second scenario, operations are scheduled so tightly that there is not enough slack available for routing. In this article, we deal with this problem by adding an extra time slot for routing. As a result, the schedule result is “relaxed” and the completion time increased. Note that in relaxing the schedule, the ordering of start times of operations is not changed; therefore, the change in schedule has no impact on other aspects of synthesis, namely, resource binding and module placement. The updated assay completion time includes the routing-time cost and reflects the actual time needed for executing the biochemical protocol on the synthesized biochip.

5. DEFECT-TOLERANT SYNTHESIS

In Section 4, we addressed the problem of integrating droplet routing in the synthesis flow. In this section, we focus on enhancing the robustness of the synthesized design. In order to do this, we incorporate defect tolerance as an objective for routing-aware synthesis. Defect-tolerance methods can be viewed as being either anticipatory, that is, anticipating defect occurrences and designing the system to be defect-resilient, or based on postmanufacture reconfiguration and resynthesis. Here we refer to these two types of defect tolerance as presynthesis and postsynthesis defect tolerance, respectively.

5.1 Postsynthesis Defect Tolerance

We first focus on postsynthesis defect tolerance. Digital microfluidic biochips are fabricated using standard microfabrication techniques [Fair et al. 2003]. Due to the underlying mixed technology and multiple energy domains, they exhibit unique failure mechanisms and defects. A manufactured microfluidic array may contain several defective cells. We have observed defects such as dielectric breakdown, shorts between adjacent electrodes, and electrode degradation [Su et al. 2003].

Reconfiguration techniques can be used to bypass faulty cells or faulty optical detectors to tolerate manufacturing defects. Bioassay operations that are bound to these faulty resources in the original design need to be remapped to other fault-free resources. Due to strict resource constraints in the fabricated biochip, alterations in the resource-binding operation, schedule, and placement must be carried out carefully. Our proposed system-level synthesis tool can be easily modified to deal with this issue. To reconfigure a defective biochip, a PRSA-based algorithm along the lines of that described in Section 4 is used. The following additional considerations must be taken into account.

The objective during reconfiguration is to minimize the bioassay completion time while accommodating all microfluidic modules and optical detectors in the fabricated microfluidic array. As resource constraints, the defect-free parts of the microfluidic array and the number of fabricated fault-free nonreconfigurable resources replace the original design specifications. In the placement phase, the locations of defective cells are no longer available. Note that the locations of nonreconfigurable resources, such as integrated optical detectors and reservoirs/dispensing ports, are fixed in the fabricated biochip.

Using this enhanced synthesis tool, a set of bioassays can be easily mapped to a biochip with a few defective cells; thus we do not need to discard the defective biochip.

5.2 Presynthesis Defect Tolerance

In this section we discuss defect-tolerant design, whereby we attempt to provide guarantees on correct bioassay operation, even if the manufacturing process introduces defects. Instead of handling defects after they are detected, we attempt to achieve defect-tolerant mapping of bioassay protocols to the microfluidic array under broad assumptions of defect occurrences.

The synthesis method described in Section 4 suffers two main drawbacks. First, it does not anticipate defect occurrences and does not consider defect tolerance in the synthesis flow. Instead, it relies on the availability of unused cells in the microfluidic array to avoid defective cells that are located after manufacture. However, such a resynthesis procedure might not be feasible because of lack of availability of spare cells. Moreover, the impact on assay completion time might be significant, and the upper limit on this time might be violated. In such scenarios, the fabricated biochip must be discarded. A second drawback of defect-oblivious synthesis is that after defects are identified, the complete synthesis process must be repeated. Thus, this approach imposes additional computation burden on the design and implementation process.

We next propose a new method to incorporate defect tolerance in the unified synthesis flow for microfluidic biochips. A novel partial reconfiguration method is also presented to enhance defect tolerance after the device is manufactured.

5.2.1 Defect-Tolerance Index. The defect tolerance of a synthesized biochip can be evaluated in terms of survivability, that is, the capability to perform bioassays on a microfluidic array with defects. The defect-tolerance index (DTI) is defined as the probability that defect tolerance can be achieved via successful partial reconfiguration when the array contains defective cells [Su et al. 2005]. Partial reconfiguration refers to the relocation only of those modules that contain defective cells; other modules are not affected. The relocated modules therefore “survive” through the defects (see Figure 6).

Assume that each cell in the microfluidic array has an independent failure probability p . The DTI $D(G)$ value for a layout G can be estimated by multiplying the survival probabilities of all the modules [Su and Chakrabarty 2006],

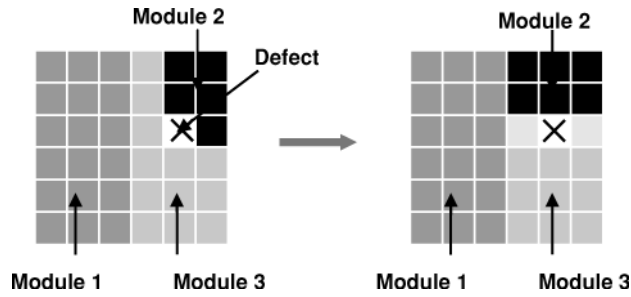


Fig. 6. Example of partial reconfiguration.

as

$$\begin{aligned}
 D(G) &\approx \prod P_s(M_i) \\
 &= \prod (1 - f_1(M_i) + f_1(M_i) \times f_2(M_i)),
 \end{aligned}$$

where $M_i, i = 1 \dots N$ is a microfluidic module (e.g., mixer) contained in a given layout G , and $P_s(M_i)$ is the survival probability of module M . Note that $f_1(M_i)$ is the probability that the module M_i is faulty. It is determined by the equation $f_1(M_i) = 1 - p \bullet A(M_i)$, where $A(M_i)$ is the total number of cells contained in M_i . Finally, $f_2(M_i)$ is the probability that M_i can be successfully reconfigured if it becomes faulty [Su et al. 2005].

Now we incorporate DTI into the PRSA-based unified synthesis method. We first define layout vulnerability by $V = 1 - D$. Layouts with low vulnerability are likely to provide high probability of successful partial reconfiguration. To find such designs, we combine vulnerability with time- and area cost, thus deriving a new fitness function to control the PRSA-based procedure. Candidate designs with low survivability are discarded during evolution. Thus, the synthesis procedure anticipates defect occurrences and selects designs that allow reconfiguration of a large number of modules while meeting constraints on array size and bioassay-processing time.

5.2.2 Partial Reconfiguration and Partial Resynthesis. Next we discuss how defects can be bypassed after manufacture. In the defect-oblivious approach described in Section 4, defect tolerance is achieved by complete resynthesis, which can be very time consuming. Here we propose an efficient method to achieve defect tolerance without need for resynthesis. This method is based on the concept of partial reconfiguration, which was introduced in Section 5.2.1. If the number of defective cells is not excessive, most microfluidic modules on the array are not affected and need not be reconfigured. As discussed in Section 5.2.1, the incorporation of defect tolerance in the design flow ensures a high probability of partial reconfigurability of the modules, that is, it is very likely that the defective biochip can be made usable via partial reconfiguration.

For each affected module, we search the array for available defect-free areas for partial reconfiguration. This can be accomplished fast because the search space is restricted to the layouts in the modules' time duration. Once a module is relocated, the algorithm updates the corresponding layout and starts the

search for the next module. Resource-binding and -scheduling results are not changed. Only the placement of defective modules is modified; therefore, this method is much faster than a complete resynthesis procedure.

In some cases, there may be not enough defect-free cells to carry out partial reconfiguration for some defective modules. We therefore introduce a new method called partial resynthesis. The key idea here is to truncate the bioassay and carry out resynthesis only for those modules that start later than the earliest-in-use defective module. Although this partial resynthesis procedure may take as much time as complete resynthesis in the worst case, that is, if the first in-use module is defective and cannot be relocated, it is faster on average than the complete resynthesis procedure.

Using these two methods, the complexity of performing postmanufacture processing for defect tolerance can be greatly reduced compared to resynthesis. The time needed to complete a set of bioassays is also significantly decreased.

6. EXPERIMENTAL EVALUATION

In this section, we evaluate the defect-tolerant droplet-routing-aware synthesis method by using it to design a biochip for a real-life protein assay.

Recently, the feasibility of performing a colorimetric protein assay on a digital microfluidic biochip has been successfully demonstrated [Srinivasan et al. 2004]. Based on the Bradford reaction [Srinivasan et al. 2004], the protocol for a generic droplet-based colorimetric protein assay is as follows. First, a droplet of the sample, such as serum or some other physiological fluid containing protein, is generated and dispensed into the biochip. Buffer droplets such as from a 1M NaOH solution are then introduced to dilute the sample to obtain a desired dilution factor (DF). This on-chip dilution is performed using multiple hierarchies of binary mixing/splitting phases, referred to as the interpolating serial dilution method [Fair et al. 2003]. The mixing of a sample droplet of protein concentration C and a unit buffer droplet results in a droplet with twice the unit volume, and concentration $C/2$. The splitting of this large droplet results in two unit-volume droplets of concentration $C/2$ each. Continuing this step in a recursive manner using diluted droplets as samples, an exponential dilution factor of $DF = 2^N$ can be obtained in N steps. After dilution, droplets of reagents, such as Coomassie brilliant blue G-250 dye, are dispensed into the chip, and they mix with the diluted sample droplets. Next, the mixed droplet is transported to a transparent electrode, where an optical detector (e.g., a LED-photodiode setup) is integrated. The protein concentration can be measured from the absorbance of the products of this colorimetric reaction, using a rate kinetic method [Srinivasan et al. 2004]. Finally, after the assay is completed, all droplets are transported from the array to the waste reservoir. A sequencing graph model can be developed from the aforementioned protocol for a protein assay ($DF = 128$), as shown in Figure 7. There are a total of 103 nodes in one-to-one correspondence with the set of operations in a protein assay, where DsS , DsB_i ($i = 1, \dots, 39$), and DsR_i ($i = 1, \dots, 8$) represents the generation and dispensing of sample, buffer, and reagent droplets, respectively.

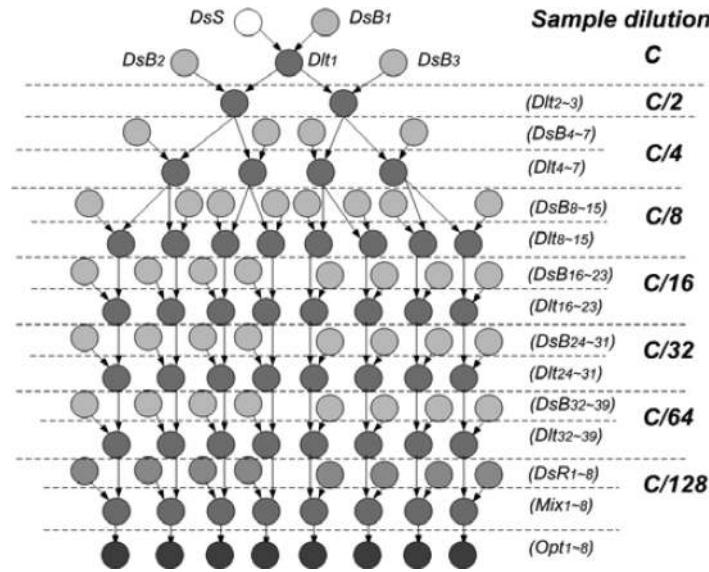


Fig. 7. Sequencing graph for a protein assay.

In addition, Dlt_i ($i = 1, \dots, 39$) denotes the binary dilution (including mixing/splitting) operations, Mix_i ($i = 1, \dots, 8$) represents the mixing of diluted sample droplets and reagent droplets, and Opt_i ($i = 1, \dots, 8$) denotes the optical detection of the droplets. Until the fourth step of a serial dilution, all diluted sample droplets are retained in the microfluidic array. After that stage, for each binary dilution step, only one diluted sample droplet is retained after splitting, while the other droplet is moved to the waste reservoir.

The basic operations for the protein assay have been implemented on a digital microfluidic biochip [Fair et al. 2003; Su et al. 2006]. Experiments indicate that the dispensing operation takes 7 sec. [Fair et al. 2003]. The operation times of various mixers have been found to differ [Fair et al. 2003]. A binary dilution operation can also be easily implemented by mixing of the sample droplet, followed by droplet splitting. Absorbance of the assay product can be measured using an integrated LED-photodiode setup. Experiments indicate this absorbance measurement takes 30 sec. [Su et al. 2005]. The microfluidic module library for a protein assay is shown in Table I. A total of 122 interdependent module pairs must be routed for this protocol. Clearly, the large number of droplet transportation operations in this protocol makes it difficult for a biochemist user or a postsynthesis design tool to determine transportation paths. We also need to specify some design parameters for the biochip to be synthesized. Different design specifications can be determined based on user needs and manufacturing constraints.

6.1 Results for Routing-Aware Synthesis

We first evaluate the proposed routing-aware synthesis method described in Section 4. We apply it to an example in which we set the maximum

Table I. Experimentally Characterized Module Library for Synthesis

Operation	Resource	Time (s)
DsS;DsB; DsR	On-chip reservoir/dispensing port	7
Dlt	2×2 -array dilutor	12
	2×3 -array dilutor	8
	2×4 -array dilutor	5
	4-electrode linear array dilutor	7
Mix	2×2 -array mixer	10
	2×3 -array mixer	6
	2×4 -array mixer	3
	4-electrode linear array mixer	5
Opt	LED + Photodiode	30
Storage	Single cell	N/A

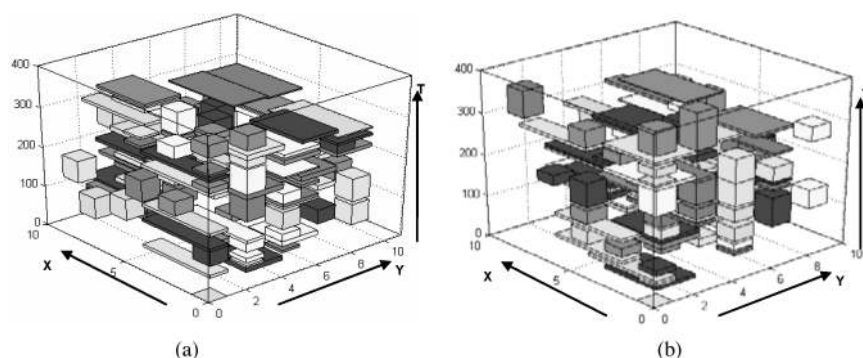


Fig. 8. A 3D model illustrating the synthesis results: (a) routing-oblivious method of Su and Chakrabarty [2005]; (b) the proposed method.

microfluidic-array size to be 100 cells, and the maximum allowable completion time for the protein assay to be 400 sec. We assume only one on-chip reservoir/dispensing port available for sample fluids, but two such ports for buffer fluids, two for reagent fluids, and one for waste fluids. Finally, we assume that at most four optical detectors can be integrated into this biochip.

We first use the routability-oblivious PRSA-based algorithm from Su and Chakrabarty [2005] to find a desirable solution for the protein assay that satisfies design specifications. The solution thus obtained yields a biochip design with a 10×10 microfluidic array, an assay completion time of 377 sec., a maximum module distance of 14 electrodes, and an average distance of 3 electrodes. Next we use the droplet-routing-aware synthesis method using the procedure of Figure 5. The procedure yields a biochip design with a 10×10 microfluidic array, a completion time of 378 sec., a maximum module distance of 7 electrodes, and an average distance of 1 electrode. The computation time for the routability-oblivious and routing-aware methods for the protein assay are 4 min. and 5 min., respectively, on a 3.00 GHz dual-core Intel Xeon server with 4G of RAM.

We illustrate the synthesis results, namely, assay operation schedule and module placement, using the 3D box model shown in Figure 8 (a color version of this figure can be found online through the ACM Digital Library). Each

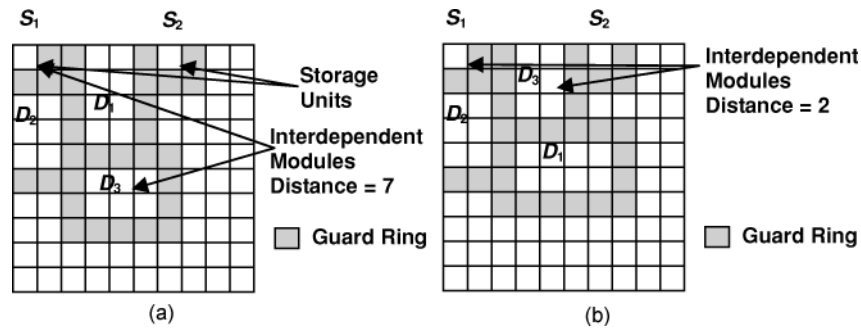


Fig. 9. (a) A snapshot of a nonroutable layout from routing-oblivious synthesis (time instant 297s); (b) corresponding layout in routing-aware synthesis (time instant 299s).

microfluidic module is represented as a 3D box, the base of which denotes the rectangular area of the module and the height denoting the time span of the corresponding assay operation. The projection of a 3D box on the X-Y plane represents the placement of this module on the microfluidic array, while the projection on the Z-axis (time axis) represents the schedule of the assay operation. Note that the synthesis results determine the locations of integrated optical detectors. Transparent electrodes for optical detection are used in the microfluidic array.

Although the two designs have comparable area- and time cost, the routing-aware synthesis method leads to a 50% reduction in the average and maximum module distances. This indicates a significant improvement of routability and reduction of the time cost for carrying out droplet routing. To verify this improvement, we applied the postsynthesis routing method of Su et al. [2006] to find efficient droplet-pathway routing for both layouts. We find that while routing-aware synthesis easily leads to a feasible routing plan, the layout for the routing-oblivious result is not routable, that is, no pathway is available for certain droplet manipulations. Figure 9(a) shows a snapshot of the layout for the routing-oblivious result taken at time instant 297s. In this snapshot, a droplet is to be routed from the storage unit S1 to dilutor D3, which is located 7 electrodes away in the routing-oblivious layout. However, as shown in Figure 9(a), there is no pathway available for routing due to the compact layout and large module distance. By contrast, in the layout derived from the routing-aware synthesis procedure, since the average module distance is minimized, D3 is placed next to S1 and the droplet pathway can be trivially determined.

Thus we can see that without violating constraints on time- and area cost, the routing-aware method carefully arranges interdependent modules to be close proximity. Therefore, it ensures that droplet pathways can be determined with a high probability. On the other hand, the routing-oblivious method only aims at satisfying constraints on time- and area cost. As a result, the interdependent modules are likely to be separated by other modules when routing-oblivious synthesis is employed; a consequence is that routing solutions cannot be obtained. Without a careful arrangement of modules, routing-oblivious synthesis can find feasible routes only if the area constraint is sufficiently loose whereby

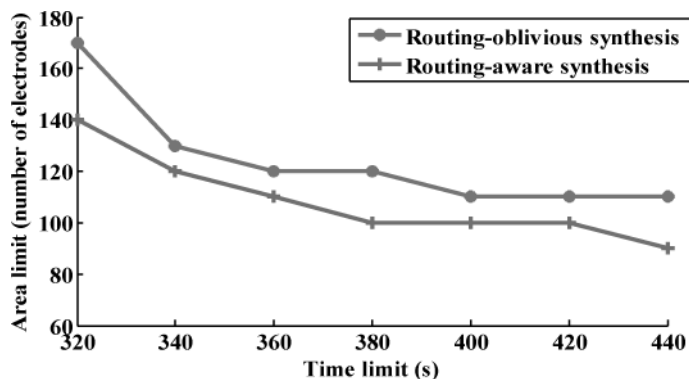


Fig. 10. Feasibility frontier and feasible design region for Su and Chakrabarty [2005] and the routing-aware synthesis method.

enough chip area is available to create droplet pathways. As a result, time- and area cost are compromised and the design specifications might not be met.

We examine this issue as follows. We first synthesize the protein assay under a set of design specifications using both the routing-oblivious and routing-aware synthesis methods. The pool of design specifications is defined by the Cartesian product of a set of time-cost limits ($T = \{T_1, T_2, T_3, \dots, T_n\}$) and a set of area-cost limits ($A = \{A_1, A_2, A_3, \dots, A_n\}$). Therefore, each synthesized chip G_{ij} corresponds to a point (T_i, A_j) . For each synthesized chip, we check if it is routable. A point (T_i, A_j) is referred to as a feasibility boundary point if there are no other points (T_m, A_n) such that G_{ij} is routable and $T_m < T_i, A_n < A_j$. A feasibility frontier is defined by connecting all the feasibility boundary points. Therefore, the feasible design region is defined by the area above the feasibility frontier. Here we set $T = \{320, 340, 360, \dots, 440\}$ and $A = \{60, 70, 80, \dots, 180\}$ and carry out both the routing-oblivious and routing-aware synthesis (the unit of T is sec., while the unit of A is measured in terms of number of electrodes). The feasibility frontier is derived for both methods as plotted in Figure 10. Note that in finding the feasibility frontier, we fix a time limit and search for the minimum chip area for which a routable synthesis result is available.

As shown in Figure 10, routing-aware synthesis leads to a lower feasibility frontier and a larger feasible design region. For tight time limits, for example, 320 sec., the routing-aware method achieves a routable synthesis result with less than 140 electrodes, while the routing-oblivious method fails when the area limit is lower than 170 electrodes. On the other hand, for a fixed-array size, for instance, 110 electrodes, routing-aware synthesis leads to a much lower assay completion time (less than 360 sec.) than the routing-oblivious method (between 380 sec. and 400 sec.). The improvement becomes more significant when routing time is considered and added to the assay completion time. In addition to assay-time reduction, routing-aware synthesis allows us to reduce chip area, and thereby the product cost for disposable and reusable biochips.

Next, we carry out postsynthesis routing for all those routable synthesis results corresponding to the feasible layouts, and use schedule relaxation as defined in Section 4 to derive the adjusted completion time. We add the droplet

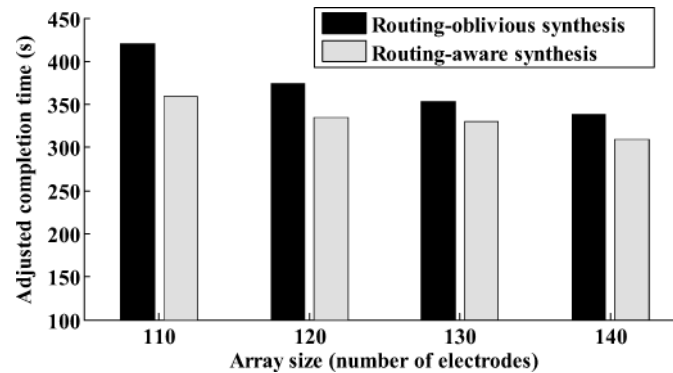


Fig. 11. Assay completion times (with droplet transportation time included) for Su and Chakrabarty [2005] and for the proposed routing-aware synthesis method.

transportation time to the assay completion time in each case. The results are shown in Figure 11. These results show that in addition to providing a greater range of feasible design points, routing-aware synthesis leads to lower assay completion times. Minimization of the assay completion time is essential for environmental-monitoring applications where sensors can provide early warning. Real-time response is also necessary for surgery and neonatal clinical diagnostics. Finally, biological samples are sensitive both to the environment and to temperature variation, and it is difficult to maintain an optimal clinical or laboratory environment on chip. To ensure the integrity of assay results, it is therefore desirable to minimize the time that samples spend on-chip before assay results are obtained. Increased throughput also improves operational reliability. Long assay durations imply that high actuation voltages need to be maintained on some electrodes, which accelerates insulator degradation and dielectric breakdown, reducing the number of assays that can be performed on a chip during its lifetime. Therefore, the minimization of assay-processing time using droplet-aware synthesis is very promising for the emerging biochips market.

6.2 Results for Postsynthesis Defect Tolerance

Next we investigate defect tolerance using the previous example. Assume that the aforementioned biochip has been fabricated. Suppose that, due to particle contamination, three cells in the 10×10 microfluidic array in Figure 8 are rendered defective, as shown in Figure 12(a). In order to ensure that the protein assay can still be carried out on this biochip, we need to bypass these faulty cells during assay operation. Moreover, due to defective cells, some nonreconfigurable resources may no longer be available. In this example, we assume that one optical detector is rendered defective after manufacturing. Thus, the operations assigned to this detector have to be remapped to other detectors. The modified synthesis method proposed in Section 5.1 is used here to carry out the reconfiguration so as to tolerate these manufacturing defects. The reconfiguration results are shown in Figure 12(b). This new design allows the

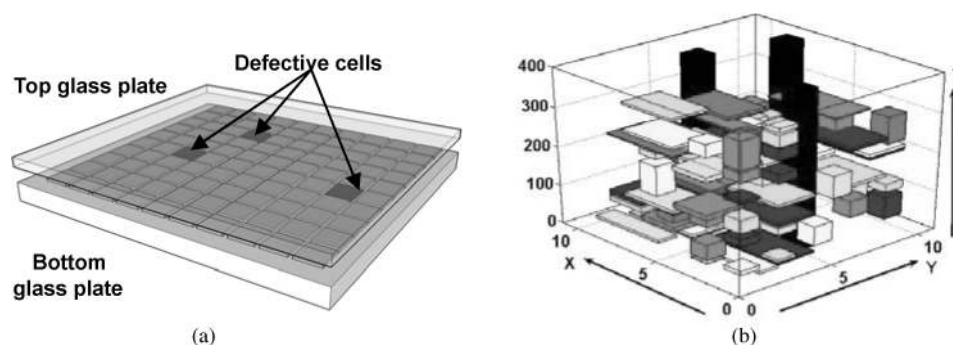


Fig. 12. (a) A defective 10×10 microfluidic array; (b) reconfiguration results for postsynthesis defect tolerance.

protein assay to operate on this defective biochip, with an increase of only 7% in the completion time; that is, the completion time is now 387 sec.

6.3 Results for Presynthesis Defect Tolerance

We next evaluate the presynthesis defect tolerance that is achieved for the protein assay using the enhanced routing-aware method described in Section 5.2. We first use this method to find a desirable solution that satisfies design specifications. The solution thus obtained yields a biochip design with a 10×10 microfluidic array and an assay completion time of 390 sec. Next, we compare this with a design synthesized using the routing-aware defect-oblivious method described in Section 4. The metric for comparison is the amount of defect tolerance exhibited by each design. For the defect-oblivious case, we use the design shown in Figure 8(b), which is a 10×10 microfluidic array with an assay completion time of 377 sec.

We evaluate the defect tolerance of the two synthesized designs by injecting random defects. Note that defects may occur in different patterns with different probabilities, due to their different underlying causes. A comprehensive modeling of realistic defects for digital microfluidics has been reported in Xu and Chakrabarty [2007]. Irrespective of their underlying cause (electrical, fluidic, mechanical, etc.), defects can be broadly classified into two categories. The first includes all defects on electrodes and wire connections, such as bad soldering, electrode charging, and electrode breakdown [Xu and Chakrabarty 2007]. These defects usually affect only one electrode. In digital microfluidics, electrodes are identical and isolated from each other. Therefore, if the occurrences of these defects are treated as random variables, we can assume that they are independent and identically distributed. The second category consists of all defects occurred in the coating layer or on the top plate. Unlike isolated electrodes, the coating layer and top plate are continuous; a defect in such continuous media may affect more than one electrode. In other words, if an electrode is affected by such a defect, it is likely that its neighbors are also affected. Therefore, we can model this phenomenon using conditional probabilities. For a single electrode, if all of its neighbors are defect free, the defect occurrence probability for the electrode is p . However, if one of its neighbors is defective, its defect

occurrence probability becomes p' . Note that an electrode may have multiple defective neighbors. For simplicity, we assume the defect occurrence probability for the electrode is also p' in this case.

A design is deemed robust if the injected defects can be bypassed by partial reconfiguration. Defects can be classified into one of three categories based on their impact on bioassay functionality. The first category includes defects that affect only unused cells in the array. As the biochip functionality is not compromised, these defects are referred to as *benign*. The second category refers to defects that cause significant “fragmentation” of the array, whereby it is no longer possible to relocate a microfluidic module to another part of the array due to lack of availability of defect-free cells. These defects are referred to as *catastrophic*. The third category includes defects that are neither benign nor catastrophic. The microfluidic array can be reconfigured for such defects; hence we refer to these defects as *repairable*.

A biochip that contains only benign defects is placed in group I; a biochip that contains catastrophic defects is placed in group II; and finally, a biochip that contains only repairable and benign defects is placed in group III. Let N_t be the total number of biochips in a representative sample, and let N_i be the number of biochips in group I, $1 < i < 3$. Clearly $N_1 + N_2 + N_3 = N_t$. We next define two ratios related to the defect-tolerance capability of the synthesized biochip: (i) robustness index $r = (N_1 + N_3)/N_t$; (ii) failure index $f = N_2/N_t$. The goal of defect-aware synthesis is to maximize r and minimize f .

Resynthesis must be carried out for the biochips in group II, that is, for biochips that suffer from catastrophic defects. Let the bioassay completion time before (after) resynthesis be $T_1(T_2)$. We define the time degradation td as follows: $td = (T_2 - T_1)/T_1$. Another goal of defect-aware synthesis is to minimize td .

We take 100 simulated samples of a microfluidic biochip synthesized for the protein assay with defect tolerance as a criterion and without defect tolerance, namely, in a defect-oblivious manner. We first carry out the simulation considering only the electrode defects. In each case, we randomly inject defects by assuming that each unit cell is defective with probability p ($p = 0.01, 0.05, 0.1$ in our experiments). We then determine the ratios r , f , and td for both methods. The results are shown in Table II.

As shown in Table II, although the defect-tolerant design leads to slightly higher assay times, it also leads to a DTI value of 0.8918 at $p = 0.05$, which implies that almost 90% of the modules can be reconfigured if they are affected by defects. This is a considerable improvement over the DTI value of 0.0144 obtained using the defect-oblivious method. This improvement is also apparent from the comparison of the failure ratio f , the robustness index r , and the time degradation td from Table II.

For all three values of p , defect-aware synthesis results in a higher value of r and a considerably lower f . Moreover, the defect-aware biochip design also provides a much lower value of td , which implies that for resynthesized biochips, the performance is compromised much less. Since the original time-cost for the two methods are comparable, the difference in td is therefore even more significant. Moreover, td falls more sharply for smaller values of p for the

Table II. Defect Tolerance for Defect-Oblivious and Defect-Tolerant Method for the Protein Assay

	DTI value	Assay time (s)	Array area	r	f	td
Defect-oblivious	0.0081	377	10×10	0.12	0.88	1.45
Defect-tolerant	0.7812	390	10×10	0.83	0.17	1.21

(a) $p = 0.1$

	DTI value	Assay time (s)	Array area	r	f	td
Defect-oblivious	0.0144	377	10×10	0.22	0.78	1.29
Defect-tolerant	0.8918	390	10×10	0.83	0.17	1.09

(b) $p = 0.05$

	DTI value	Assay Time (s)	Array Area	r	f	td
Defect-oblivious	0.1035	377	10×10	0.30	0.70	1.21
Defect-tolerant	0.9414	390	10×10	0.94	0.07	1.04

(c) $p = 0.01$

defect-aware design. Therefore, for low defect probabilities (as is often the case for mature manufacturing processes), the proposed defect-tolerant synthesis method allows resynthesis in cases of catastrophic defects, with lower time-cost increases. This feature is often required for many biochip applications.

Next, we rerun the simulations, considering the coating and grounding defects. Again, we first inject defects by assuming that each unit cell is defective with probability p ($p = 0.01, 0.05, 0.1$). Next, a second injection iteration is carried out and additional defects are injected at all the neighbors of defective electrodes in the first iteration with probability $p' = 0.1$. Simulation results are shown in Table III. Similar conclusions can be drawn as in Table II.

In summary, incorporation of presynthesis defect tolerance into the routing-aware synthesis tool leads to a significant improvement in robustness of the synthesized design. It also allows the search for an optimal design under multiple design specifications, including completion time, chip area, routability, and system dependability.

For the protein assay example, we next run the defect-tolerant routing-aware and defect-oblivious routing-aware algorithms under a set of combinations of weights in the fitness function. For each combination of weights, if the derived synthesis result is not routable, the algorithm is repeated until a routable design is found. Next we carry out random defect injection into each design and obtain its failure rate f as defined in Section 5. We map each design G to a 3D point (T_G, A_G, F_G) , where T_G, A_G, F_G are completion time, chip area, and failure rate of the design, respectively. Similar to the definition in Section 6.1, a point (T_G, A_G, F_G) is referred to as a feasibility boundary point if there are no other points (T_m, A_m, F_m) such that $T_m < T_G, A_m < A_G$, and $F_m < F_G$. A feasibility frontier surface is obtained by connecting all the feasibility boundary points, as shown in Figure 13. The feasible design region corresponds to the space above the feasible surface. Any design specification can be met whose corresponding is point located in this region; otherwise, no feasible design exists for this specification.

As shown in Figure 13, defect-tolerant routing-aware synthesis leads to a lower-feasibility frontier surface and a larger feasible design space as compared to the defect-oblivious method. For tight time- and area limits, for example,

Table III. Defect Tolerance for Defect-Oblivious and Defect-Tolerant Method for the Protein Assay ($p' = 0.1$)

	DTI value	Assay time (s)	Array area	r	f	td
Defect-oblivious	0.0081	377	10×10	0.07	0.93	1.49
Defect-tolerant	0.7812	390	10×10	0.68	0.32	1.31

(a) $p = 0.1$

	DTI value	Assay time (s)	Array area	r	f	td
Defect-oblivious	0.0144	377	10×10	0.14	0.86	1.32
Defect-tolerant	0.8918	390	10×10	0.80	0.20	1.21

(b) $p = 0.05$

	DTI value	Assay Time (s)	Array Area	r	f	td
Defect-oblivious	0.1035	377	10×10	0.21	0.79	1.26
Defect-tolerant	0.9414	390	10×10	0.88	0.12	1.11

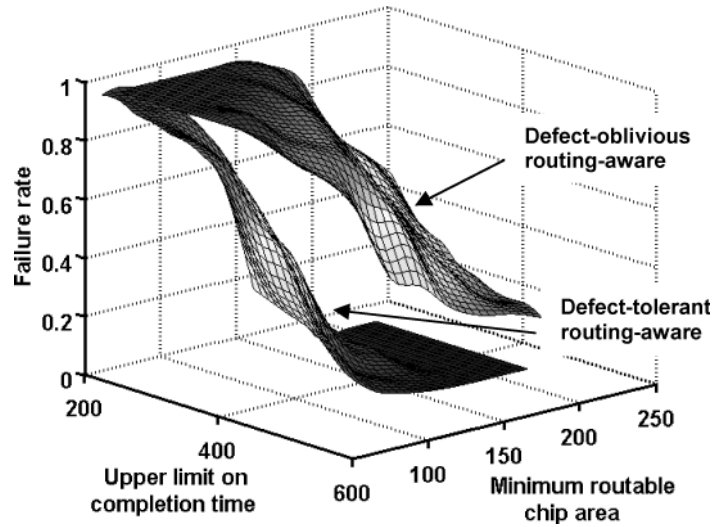
(c) $p = 0.01$ 

Fig. 13. Feasibility frontier surface and feasible design region for defect-tolerant and defect-oblivious routing-aware synthesis methods.

400 sec., 110 electrodes, the defect-tolerant method achieves a routable synthesis result with failure rate less than 0.5, while the defect-oblivious method requires a failure rate of at most 0.1. On the other hand, for a predetermined failure-rate limit, for instance, 0.3, defect-aware synthesis leads to a much lower assay completion time (less than 390 sec.) and smaller chip area (less than 100 electrodes) than does the defect-oblivious method, which requires a completion time of 450 sec. and a minimum chip area of 150 electrodes. The improvement becomes more significant for limits on the failure rate. Defect-aware synthesis allows us to reduce chip area and assay completion time, and thereby the product cost for reliable biochip platforms.

7. CONCLUSIONS

We have presented a droplet-routing-aware automated synthesis tool for microfluidic biochips. Droplet routability, defined as the ease with which droplet pathways can be determined, has been estimated and integrated in the synthesis flow. The proposed method increases the likelihood that feasible droplet pathways can be found for area-constrained biochip layouts. We have demonstrated the advantages of this approach using a large-scale protein assay based on the Bradford reaction. To increase system dependability, two defect-tolerance schemes, namely, presynthesis and postsynthesis, have been incorporated into the routing-aware design method. Simulation results have highlighted the increase in defect tolerance achieved in each case. The design techniques presented in this article relieve the chip user from the burden of postsynthesis droplet routing, and facilitate the automated design of biochips with guaranteed high levels of defect tolerance. In this way, the biochip user can concentrate on the development of nano- and microscale bioassays, leaving the cumbersome implementation details to the synthesis tools.

REFERENCES

- BOHRINGER, K. F. 2006. Modeling and controlling parallel tasks in droplet-based microfluidic systems. *IEEE Trans. Comput. Aided Des. Integr. Circ. Syst.* 25, 329–339.
- DEAMER, D. W. AND AKESON, M. 2000. Nanopores and nucleic acids: Prospects for ultrarapid sequencing. *Trends Biotechnol.* 18, 147–151.
- FAIR, R. B., KHLYSTOV, A., SRINIVASAN, V., PAMULA, V. K., AND WEAVER, K. N. 2004. Integrated chemical/biochemical sample collection, pre-concentration, and analysis on a digital microfluidic lab-on-a-chip platform. In *Proc. of International Society for Optical Engineering 5591*, 113–124.
- FAIR, R. B., KHLYSTOV, A., TAILOR, T. D., IVANOV, V., EVANS, R. D., SRINIVASAN, V., PAMULA, V. K., POLLACK, M. G., GRIFFIN, P. B., AND ZHOU, J. 2006. Chemical and biological applications of digital microfluidic devices. *IEEE Des. Test Comput.* 24, 10–24.
- FAIR, R. B., SRINIVASAN, V., REN, H., PAIK, P., PAMULA, V. K., AND POLLACK, M. G. 2003. Electrowetting-Based on-chip sample processing for integrated microfluidics. In *Proceedings of the International Electronic Device Meeting*, 32.5.1–32.5.4.
- GRIFFITH, E. J., AKELLA, S., AND GOLDBERG, M. K. 2006. Performance characterization of a reconfigurable planar array digital microfluidic system. *IEEE Trans. Comput. Aided Des. Integr. Circ. Syst.* 25, 340–352.
- GUISEPPI-ELIE, A., BRAHIM, S., SLAUGHTER, G., AND WARD, K. R. 2005. Design of a subcutaneous implantable biochip for monitoring of glucose and lactate. *IEEE Sensors J.* 5, 345–355.
- HWANG, W. L., SU, F., AND CHAKRABARTY, K. 2006. Automated design of “pin-constrained” digital microfluidic arrays for lab-on-a-chip applications. In *Proceedings of the Design Automation Conference (DAC)*, 925–930.
- PAIK, P. Y., PAMULA, V. K., AND CHAKRABARTY, K. 2007. *Adaptive Cooling of Integrated Circuits using Digital Microfluidics*. Artech House, Norwood, MA.
- PAIK, P. Y., PAMULA, V. K., POLLACK, M. G., AND FAIR, R. B. 2003. Rapid droplet mixers for digital microfluidic systems. *Lab on a Chip* 3, 253–259.
- PAMULA, V. K., SRINIVASAN, V., CHAKRAPANI, H., FAIR, R. B., AND TOONE, E. J. 2005. A droplet-based lab-on-a-chip for colorimetric detection of nitroaromatic explosives. In *Proceedings of the IEEE Annual International Workshop on Micro Electro Mechanical Systems (MEMS)*, 722–725.
- PFEIFFER, A. J., MUKHERJEE, T., AND HAUAN, S. 2006. Synthesis of multiplexed biofluidic microchips. *IEEE Trans. Comput. Aided Des. Integr. Circ. Syst.* 2, 321–333.
- RICKETTS, A. J., IRICK, K., VIJAYKRISHNAN, N., AND IRWIN, M. J. 2006. Priority scheduling in digital microfluidics-based biochips. In *Proceedings of the Design Automation and Test in Europe (DATE)*, 329–334.

- SAIT, S. AND YOUSSEF, H. 1995. *VLSI Physical Design Automation: Theory and Practice*. IEEE Press, New York.
- SECHEN, C. 1988. *VLSI Placement and Global Routing Using Simulated Annealing*. Kluwer Academic, Boston, MA.
- SCHULTE, T. H., BARDELL, R. L., AND WEIGL, B. H. 2002. Microfluidic technologies in clinical diagnostics. *Clinica Chimica Acta* 321, 1–10.
- SRINIVASAN, V., PAMULA, V. K., AND FAIR, R. B. 2004. An integrated digital microfluidic lab-on-a-chip for clinical diagnostics on human physiological fluids. *Lab on a Chip*, 310–315.
- SRINIVASAN, V., PAMULA, V. K., PAIK, P. Y., AND FAIR, R. B. 2004. Protein stamping for MALDI mass spectrometry using an electrowetting-based microfluidic platform. In *Proceedings of the International Society for Optical Engineering*, vol. 5591, 26–32.
- SRINIVASAN, V., PAMULA, V. K., POLLACK, M. G., AND FAIR, R. B. 2004. Clinical diagnostics on human whole blood, plasma, serum, urine, saliva, sweat, and tears on a digital microfluidic platform. In *Proceedings of the Miniaturized Systems for Chemistry and Life Sciences (μ TAS)*, 1287–1290.
- SU, F. AND CHAKRABARTY, K. 2004. Architectural-Level synthesis of digital microfluidics-based biochips. In *Proceedings of the International Conference on Computer-Aided Design (ICCAD)*, 223–228.
- SU, F. AND CHAKRABARTY, K. 2005. Unified high-level synthesis and module placement for defect-tolerant microfluidic biochips. In *Proceedings of the Design Automation Conference (DAC)*, 825–830.
- SU, F. AND CHAKRABARTY, K. 2006. Module placement for fault-tolerant microfluidics-based biochips. *ACM Trans. Des. Autom. Electron. Syst.* 11, 682–710.
- SU, F., HWANG, W., AND CHAKRABARTY, K. 2006. Droplet routing in the synthesis of digital microfluidic biochips. In *Proceedings of the Design Automation and Test in Europe (DATE)*, 323–328.
- SU, F., HWANG, W., MUKHERJEE, A., AND CHAKRABARTY, K. 2005. Defect-Oriented testing and diagnosis of digital microfluidics-based biochips. In *Proceedings of the International Test Conference (ITC)*, 487–496.
- SU, F., OZEV, S., AND CHAKRABARTY, K. 2003. Testing of droplet-based microelectrofluidic systems. In *Proceedings of the IEEE International Test Conference (ITC)*, 1192–1200.
- VANDER WOERD, M., FERREE, D., AND PUSEY, M. 2003. The promise of macromolecular crystallization in microfluidic chips. *J. Structur. Biol.* 142, 180–187.
- VERPOORTE, E. AND DE ROOLJ, N. F. 2003. Microfluidics meets MEMS. *Proc. IEEE* 91, 930–953.
- XU, T. AND CHAKRABARTY, K. 2007. Functional testing of digital microfluidic biochips. In *Proceedings of International Test Conference*.
- YUH, P.-H., YANG, C.-L., AND CHANG, Y.-W. 2006. Placement of digital microfluidic biochips using the T-tree formulation. In *Proceedings of the Design Automation Conference (DAC)*, 931–934.
- ZENG, J. AND KORSMEYER, T. 2004. Principles of droplet electrohydrodynamics for lab-on-a-chip. *Lab on a Chip*, 265–277.

Received July 2007; revised January 2008; accepted January 2008