

Cross-Contamination Avoidance for Droplet Routing in Digital Microfluidic Biochips*

Yang Zhao and Krishnendu Chakrabarty
Department of Electrical and Computer Engineering
Duke University, Durham, NC 27708, USA

Abstract—Recent advances in droplet-based digital microfluidics have enabled biochip devices for DNA sequencing, immunoassays, clinical chemistry, and protein crystallization. Since cross-contamination between droplets of different biomolecules can lead to erroneous outcomes for bioassays, the avoidance of cross-contamination during droplet routing is a key design challenge for biochips. We propose a droplet-routing method that avoids cross-contamination in the optimization of droplet flow paths. The proposed approach targets disjoint droplet routes and minimizes the number of cells used for droplet routing. We also minimize the number of wash operations that must be used between successive routing steps that share unit cells in the microfluidic array. Two real-life biochemical applications are used to evaluate the proposed droplet-routing methods.

I. INTRODUCTION

Droplet-based digital microfluidics is an emerging technology that provides fluid-handling capability on a chip. It has therefore led to the automation of laboratory procedures in biochemistry [1]. By reducing the rate of sample and reagent consumption, digital microfluidic biochips enable continuous sampling and analysis for real-time biochemical analysis, with application to clinical diagnostics, immunoassays, and DNA sequencing. Discrete droplets of nanoliter volumes can be manipulated using electrowetting in a “digital” manner under clock control on a two-dimensional array of electrodes (“unit cells”). Synthesis and droplet-routing methods have been developed recently for the design of microfluidic biochips [3], [5], [9], [10], [12]–[16].

Many biomedical assays require the on-chip transportation of biological substances that contain large molecules such as proteins. However, proteins cannot be transported easily on a microfluidic platform [1]. This is because proteins tend to adsorb irreversibly to hydrophobic surfaces and contaminate them. Silicone oil with its low surface tension and spreading property has been advocated as a filler medium for protein assays to prevent contamination [1]. However, it has also been reported that the use of silicone oil alone is not sufficient for many types of proteins [4]. A set of wash droplets is typically used in such cases for surface cleaning between successive droplet transportation steps, especially for unit cells that are shared by droplet routes.

A drawback of current automated droplet-routing methods is that they are based on the unrestricted sharing of unit cells by various droplet routes. Therefore, cross-contamination in such cases is inevitable. Cross-contamination occurs when the residue left behind by one droplet transfers to another droplet with undesirable consequences, such as misleading assay outcome

(false position, incorrect diagnosis, etc.). As a result, the droplet-routing problem for biochips must consider the avoidance of cross-contamination during droplet transportation.

In this paper, we present a new droplet-routing method for digital microfluidic biochips. A key goal here is to avoid overlap between different droplet routes, thereby minimizing the likelihood of cross-contamination. Our approach attempts to determine disjoint droplet routes and minimize the number of cells used for droplet routing, while satisfying both timing goals and fluidic constraints. Droplet-routing time must be minimized to reduce time-to-result for biochips, which is necessary to prevent sample degradation and ensure real-time response. We also minimize the number of wash operations inserted between successive routing steps that share unit cells in the microfluidic array.

The remainder of the paper is organized as follows. Section II provides an overview of the digital microfluidic platform and related prior work on droplet routing. In Section III, we formulate the problem of disjoint droplet routing. Based on this formulation, we propose a new routing technique in Section IV. A method to optimally insert washing steps between routing steps is also presented. In Section V, we use two real-life bioassay applications as case studies to evaluate the proposed method. Finally, conclusions are drawn in Section VI.

II. BACKGROUND AND RELATED PRIOR WORK

A digital microfluidic biochip utilizes the phenomenon of electrowetting to manipulate and move nanoliter droplets containing biological samples on a two-dimensional electrode array [1]. 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. Droplets are moved by applying a control voltage to a unit cell adjacent to the droplet and, at the same time, deactivating the one just under the droplet. This electronic method of wettability control creates interfacial tension gradients that move the droplets to the charged electrode.

Fluid-handling operations such as droplet merging, splitting, mixing, and dispensing can be executed in a similar manner. 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 result are programmed into a microcontroller that drives electrodes in the array.

A number of techniques have been proposed in the literature to solve the droplet-routing problem for digital microfluidics [5], [9], [14]–[16]. These methods are based on concepts such as path-planning for robots, prioritized A^* search, network flow, integrated synthesis-and-routing, etc. However, the droplet routes

*This work was supported in part by the National Science Foundation under grant CCF-0541055.

may intersect or overlap with each other during different time intervals. Therefore, cross-contamination between droplets may happen at these sites, and this is a serious problem for all known automated droplet-routing methods.

III. PROBLEM FORMULATION AND CONSTRAINTS

In this section, we describe the problem of droplet routing to avoid cross-contamination.

A. Problem formulation

Given a schedule of bioassay operations (derived from architectural-level synthesis [12]) and the locations of these modules on the biochip floorplan (derived from module placement [13]), routing determines the paths for droplet transportation using the available cells in the microfluidic array. Droplets are transported along these routes between modules, or between modules and fluidic I/O ports (e.g., on-chip reservoirs and sensors).

The fluidic ports on the boundary of microfluidic modules are referred to as *pins*. The droplet routes between pins of different modules or on-chip reservoirs are referred to as *nets*. Therefore, a fluidic route on which a single droplet is transported between two terminals (i.e., one source and one sink) can be modeled as a 2-pin net. Two droplets from different terminals are often transported to one common module (i.e., mixer or diluter) for mixing or diluting. To allow droplet mixing simultaneously during their transportation, we model such fluidic routes using 3-pin nets.

Cross-contamination is likely to occur when multiple droplet routes intersect or overlap with each other. At the intersection site of two droplet routes, a droplet that arrives at a later clock cycle can be contaminated by the residue left behind by another droplet that passed through at an earlier clock cycle. The more cells that two droplet-routes share, the higher is the likelihood of cross-contamination.

Our first goal is to avoid cross-contamination between different droplet routes. The second goal is to minimize the time needed for droplet routing. Therefore, we focus on disjoint droplet routes and the minimization of the total path length over all routes, where path length is measured by the number of cells in the path from the source to the sink.

In a set of disjoint routes, a droplet route does not share any cell in its path with each of the other droplet routes in that set. Such routes eliminate the possibility of a droplet being transported via a cell when another droplet has already passed through it in an earlier time interval. Therefore, disjoint routes avoid cross-contamination between different droplets. The minimization of the length of the droplet routes leads to a reduction in the droplet-transportation time. It also frees up more spare cells for parallel fluidic operations and fault tolerance [14].

B. Fluidic and timing constraints

Although disjoint droplet routes avoid cross-contamination between different droplets, some droplets may inadvertently mix if they are routed too close simultaneously during their transportation. Therefore, during droplet routing, a minimum spacing between droplets must be maintained to avoid unintended mixing, except for the case that droplet merging is desired (i.e., in 3-pin nets). A segregation region is added to wrap around

the functional region of microfluidic modules to avoid conflicts between droplet routes and assay operations that are scheduled at the same time. Moreover, fluidic constraint rules in [14] need to be satisfied in order to avoid undesirable mixing.

Another constraint in droplet routing is given by an upper limit on droplet transportation time. Since a droplet may be held stationary in some clock cycles during its route, the delay for a droplet route consists of the transportation time as well as the idle time. The delay for each droplet route should not exceed some maximum, e.g., 10% of a time-slot used in scheduling, in order that the droplet-routing time can be ignored for scheduling assay operations [12].

C. Problem decomposition

Since a digital microfluidic array can be reconfigured dynamically at run-time, a series of 2-D placement configurations of modules in different time spans are obtained in the module placement phase [13]. Therefore, the droplet routing is decomposed into a series of sub-problems. In each sub-problem, the nets to be routed between the sources and the sinks are first determined. The microfluidic modules that are active at the time when droplets are transported are considered as *obstacles*. We obtain a complete droplet-routing solution by solving these sub-problems sequentially.

Cross-contamination can occur within one sub-problem, and also across two sub-problems, e.g., a droplet route in the current sub-problem may share the same cells with a droplet route in the predecessor sub-problem. We next address the problem of determining disjoint routes and minimizing the route lengths for both cases.

IV. ROUTING METHOD

In this section, we first present the basic concepts underlying the proposed routing method.

A. Graph model and disjoint routing

The problem of finding feasible disjoint routes for 2-pin or 3-pin nets in the 2-D microfluidic array can be directly mapped to the problem of finding disjoint paths (vertex-disjoint or edge-disjoint) for pairs of vertices in a graph [6], [7]. We consider a planar undirected graph $G = (V, E)$, where V is the vertex set and E is the edge set. Each vertex in the graph represents an electrode in the 2-D microfluidic array, and there is an edge between two vertices if their corresponding electrodes are adjacent. The pins in a 2-pin net are represented by a pair of vertices (t_i, s_i) in the graph. The pins in a 3-pin net are represented by a set of three vertices $(t_{1,i}, t_{2,i}, s_i)$, where $t_{1,i}$ and $t_{2,i}$ represent the two source pins, and s_i represents the destination pin. A 3-pin net $(t_{1,i}, t_{2,i}, s_i)$ can be treated as two 2-pin nets $(t_{1,i}, t_{2,i})$ and (m_i, s_i) , where m_i is the mixing point in the route of the 2-pin net $(t_{1,i}, t_{2,i})$ where the droplets from the two source pins $t_{1,i}$ and $t_{2,i}$ mix together. The route for a net is represented by a path consisting of a set of successive edges in the graph, where the endpoints of a path represent the corresponding pins. Each edge in the path denotes the fact that the electrodes represented by the two endpoints of the edge are adjacent in the droplet route.

Consider a set of disjoint routes where a droplet route does not share any cell with each of the other droplet routes. The

corresponding paths in the graph are mutually *vertex-disjoint* since a path does not share any vertex with each of the other paths in the set. Similarly, in a set of disjoint routes where a droplet route does not share any pair of adjacent cells with each of the other droplet routes, their corresponding paths in the graph are mutually *edge-disjoint* since a path does not share any edge with each of the other paths in the set.

Since a 3-pin net can be treated as two 2-pin nets, henceforth we only consider 2-pin nets for disjoint routing. In the 2-D microfluidic array, given a set of n 2-pin nets with corresponding pins $(t_1, s_1), (t_2, s_2), \dots, (t_n, s_n)$, the problem of finding feasible disjoint routes for these nets is equivalent to the problem of finding mutually vertex-disjoint or edge-disjoint paths in G , such that the endpoints of each path in G represent the corresponding pins of each net. Given $G = (V, E)$ and the vertex pairs $(t_1, s_1), (t_2, s_2), \dots, (t_n, s_n)$, the problem of determining whether mutually vertex-disjoint paths $P_1, P_2, \dots, P_n$ exist such that P_i has endpoints t_i and s_i , is NP-complete [7]. Furthermore, the problem of determining whether mutually edge-disjoint paths exist is also NP-complete, even if the graph G is a 2-D mesh [6]. Therefore, we use a heuristic approach in this work. Furthermore, it is often difficult to find vertex-disjoint paths in the underlying graph model to solve the droplet-routing problem; such paths might not exist for a given set of nets. It is therefore more practical to relax the overlap restriction and search for edge-disjoint routes in such cases. Edge-disjoint routes lead to a reduction in the number of array sites that need to be washed.

B. Avoiding cross-contamination within one sub-problem

We next present a droplet-routing algorithm based on disjoint routes that minimizes route lengths and avoids cross-contamination within a sub-problem. The input to the algorithm is a list of nets to be routed in the sub-problem, and the output is a set of vertex-disjoint (preferred) or edge-disjoint (as a design compromise) routes with minimized lengths, subject to both fluidic and timing constraints.

Within a sub-problem, the Lee algorithm, a popular technique used in grid routing [8], can obtain a single droplet route for each net. The Lee algorithm is guaranteed to find the shortest path between two pins in a two-pin net. For 3-pin net, we use the modified Lee algorithm from [14] to obtain a feasible route connecting these 3 pins. Note that the interconnection obtained by the modified Lee algorithm from [14] is not guaranteed to be of minimum length. Nevertheless, it is a desirable route in practice, allowing concurrent mixing during transportation. However, the Lee algorithm does not avoid cross contamination.

The microfluidic modules that are active in this sub-problem are considered as obstacles. The individual net in this sub-problem is routed using the modified Lee algorithm sequentially. After any net has been routed, the cells occupied by its path are marked as obstacles for the unrouted nets. Therefore, the latter route is disjoint with respect to all the previous routes. However, the routing order in the sub-problem influences the routability of all the nets. Even if each of the nets is individually routable, the routing order may prevent successful completion of routing for all the nets.

Therefore, we modify the net-routing ordering method proposed in [8] to obtain an optimized order for the routing of n

nets in a sub-problem. We first define the bounding box of a net. Assuming that two pins of a 2-pin net p are p_1 and p_2 with coordinates (x_1, y_1) and (x_2, y_2) in a 2-D microfluidic array, the coordinates of the four vertices of its corresponding bounding box are $(x_1, y_1), (x_1, y_2), (x_2, y_1)$ and (x_2, y_2) . Next we define $pin(p)$ to be the number of pins (for other nets to be routed in the current sub-problem) within the bounding box of net p . Let $Xrange(p) = |x_1 - x_2|$ and $Yrange(p) = |y_1 - y_2|$. The bounding box for a 3-pin net is defined in a similar manner. The priority number for a net p is given by the following equation, where A is a user defined parameter. The larger the priority number, the lower the priority of the corresponding net in the net-routing order.

$$priority(p) = pin(p) + A \cdot \max\{Xrange(p), Yrange(p)\}. \quad (1)$$

There are two stages in the net-routing process. In the first stage, we apply the modified Lee algorithm to each net sequentially in the net-routing order. The cells occupied by the routed paths are marked as obstacles for the unrouted nets. Each route thus obtained needs to pass the “timing delay constraint check” (TDCC). Additionally, each route also need to go through the “fluidic constraint rule check” (FCRC) with previously acquired routes according to the net-routing order. If a rule violation is found, we modify droplet motion using the modification rules in [14] to override the violation. The shortest route with minimum number of cells used is selected as the output for each net. If all the shortest routes for some net do not satisfy the timing constraint, or violate the fluidic constraint and fail the droplet-motion modification, we put this net to a conflict list in order to resolve it at the end. Note that all routes obtained in this stage are not only edge-disjoint, but also vertex-disjoint (e.g., no two droplet routes share the same cell).

In the second stage, for each net in the conflict list, we execute the modified Lee algorithm to generate several shortest routes without considering the previously routed paths in the first stage as obstacles. From these shortest routes, we select a route that has no common pair of adjacent cells with the previously obtained routes as the output. Therefore, the resulting route may intersect, but still be edge-disjoint, with the previously obtained routes. As before, this route also needs to pass both TDCC and FCRC constraints.

At this stage, we have obtained disjoint routes for all the nets. If all the shortest routes for some net in this stage do not pass either TDCC or FCRC constraint, placement refinement is required to increase the corresponding routability.

C. Optimization of wash-droplet routing

Since cross-contamination can also happen across two sub-problems, the droplet routes for the nets in the current sub-problem should avoid sharing cells with the routes in the predecessor sub-problem. Therefore, along with the active modules, the routes for all nets in the predecessor sub-problem are also treated as obstacles when routes in the current sub-problem are generated. However, in order to bypass these additional obstacles, some routes will become longer and may fail TDCC. Furthermore, a net that can be routed easily without these additional obstacles may become unroutable. Therefore, after the droplet-routing process in one sub-problem, a wash operation needs to

be introduced. In a wash operation, a wash droplet is routed to traverse selected cells and remove residue from them. The delay introduced by the wash operation is equivalent to the time needed to route the wash droplet. For a sub-problem that includes a follow-up wash operation, its droplet-routes will not be treated as obstacles for the next sub-problem.

Suppose a bioassay can be decomposed into N droplet-routing sub-problems. The parameter T_i , $1 \leq i \leq N$, is defined as the maximum transportation time needed among the droplet routes in sub-problem i if the wash operation is not performed after droplet routing in sub-problem $(i-1)$. The parameter T_i^* , $1 \leq i \leq N$, is defined as the maximum transportation time needed in sub-problem i if the wash operation is performed after the droplet routing in sub-problem $(i-1)$. Let Tw_i , $1 \leq i \leq N$, be the time needed for the wash operation after droplet routing in sub-problem $(i-1)$. Therefore, $(T_i^* + Tw_i)$ denotes the total droplet transportation time that includes the wash operation after sub-problem $(i-1)$ and routing for sub-problem i . If a wash operation is carried out between sub-problem $(i-1)$ and sub-problem i , it adds to the routing time for sub-problem i , but it also frees up more cells for routing and reduces the routing time in sub-problem i . As a result, in most cases, $T_i^* < T_i$. The decision on whether to add a washing step between two sub-problems therefore depends on the individual routing time for the sub-problem and the nature of these routes.

We next describe an optimization model to determine when the wash operation should be performed between successive routing steps. Let x_i be a binary variable defined as follows: $x_i = 1$ if the wash operation is performed between sub-problems $(i-1)$ and i , otherwise $x_i = 0$. Our goal is to minimize the total time needed for droplet routing for the sub-problems. The objective function for the optimization problem can therefore be stated as follows:

$$\text{Minimize } \mathcal{F} = T_1 + \sum_{i=2}^N (x_i(T_i^* + Tw_i) + (1 - x_i)T_i).$$

We also need to incorporate the constraint that the number of wash operations is less than the number of sub-problems. Therefore, we now formulate the optimization problem as follows: Minimize $\mathcal{F}$, subject to: $\sum_{i=2}^N x_i \leq N-1$. To solve this problem, we simply note that to minimize $\mathcal{F}$, x_i needs to be set to 1 if $T_i^* + Tw_i < T_i$, otherwise $x_i = 0$.

V. EXPERIMENTAL EVALUATION

In this section, we evaluate the proposed disjoint droplet-routing methods for two real-life bioassays, namely multiplexed *in-vitro* diagnostics on human physiological fluids, and DNA sequencing.

A. Example 1: Multiplexed *in-vitro* diagnostics

The assay protocol is described as follows. Three types of human physiological fluids, i.e., urine, serum and plasma, are sampled and dispensed to the digital microfluidic biochip. Glucose and lactate measurements are performed for each type of physiological fluid.

The sequencing graph and the schedule for the assay protocol are presented in [14]. Note that one time-slot in the schedule is set to 2 seconds. Module placement for different time-slots on a

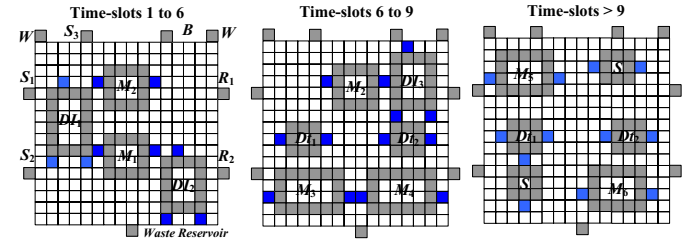


Fig. 1. Module placement for the multiplexed *in-vitro* diagnostics example.

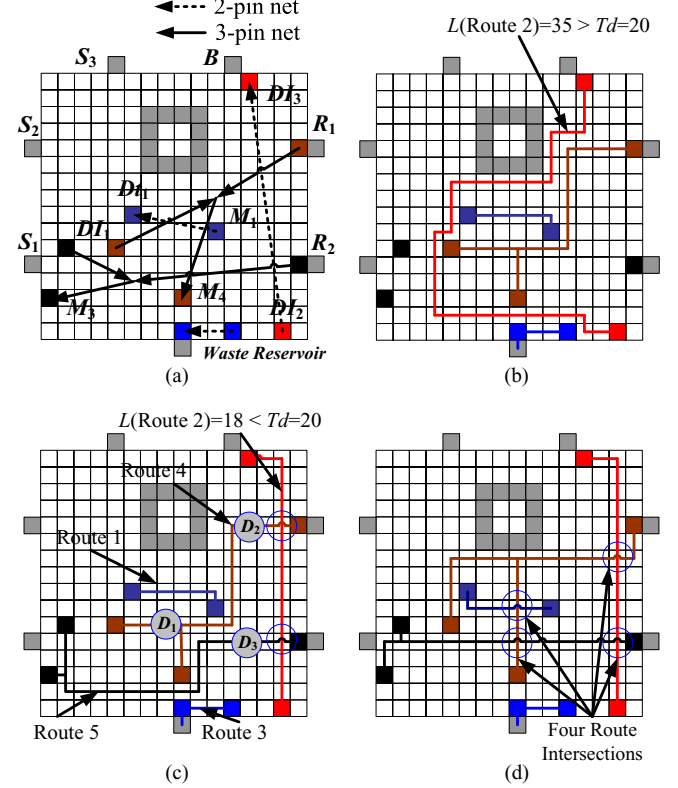


Fig. 2. Disjoint routing and cross-contamination-oblivious routing for sub-problem 3: (a) three 2-pin nets and two 3-pin nets; (b) Route 2 violates the timing-delay constraint; (c) feasible routes for all the nets using the disjoint-routing method; (d) routing results obtained using the cross-contamination-oblivious routing method.

16×16 microfluidic array is shown in Fig. 1. M_1 - M_6 are mixers, DI_1 - DI_3 are dilutors, S is a storage unit, Dt_1 and Dt_2 are two detectors, S_1 - S_3 are reservoirs for three samples, R_1 and R_2 are reservoirs for two reagents, B and W are reservoirs for buffer and wash droplets, respectively.

The routing problem is decomposed into eleven sub-problems [14]. We address these sub-problems serially by attempting to determine the set of vertex-disjoint or edge-disjoint droplet routes and minimize the number of cells used by these routes, subject to both the timing and fluidic constraints. The maximum delay constraint is set to 10% of one time slot, i.e., 0.2 second. We assume a typical clock frequency of 100 Hz. Therefore, the timing constraint Td is equal to 20 clock cycles.

Here we use sub-problem 3 to illustrate the proposed routing method. As shown in Fig. 2(a), there are three 2-pin nets and two 3-pin nets. Route 1 is defined as the path between Dt_1 and M_1 (i.e., Net 1), Route 2 is defined as the path between DI_2 and DI_3 (i.e., Net 2), Route 3 is defined as the path between DI_2 and the waste reservoir (i.e., Net 3). Route 4 is defined as the

path for 3-pin net whose pins are R_1 , DI_1 and M_4 (i.e., Net 4), while Route 5 is defined as the path for 3-pin net whose pins are R_2 , DI_1 and M_3 (i.e., Net 5). Module M_2 that is active during this time interval is considered as an obstacle for routing.

First, we decide the order for routing these nets. The parameter A in Equation (1) is set to 1. The priority number for Net 1 is 5, since the number of pins within its bounding box is 0 and the maximum of its $Xrange$ and $Yrange$ is 5. The priority number for Net 4 is 13, since the number of pins within its bounding box is 2 and the maximum of its $Xrange$ and $Yrange$ is 11. The priority number for Net 2, Net 3 and Net 5 are 15, 3 and 15 respectively. Therefore, the net-routing order is Net 3, Net 1, Net 4, Net 2 and Net 5.

While generating the route for Net 2, since the paths for previous routed Net 3, Net 1 and Net 4 have been marked as obstacles, the shortest paths between DI_2 and DI_3 (i.e., Route 2) violate the timing delay constraint, i.e., $L(\text{Route } 2) = 35$ (in cells) $> Td = 20$ (clock cycles), as shown in Fig. 2(b). Thus, we put Net 2 to the conflict list, then generate the route for Net 5. After the route for Net 5 is generated, we generate the route for Net 2 without considering previously generated paths as obstacles. A route that has no common pair of adjacent cells with the previously routed paths is selected. The obtained route for Net 2 satisfies the timing constraint, i.e., $L(\text{Route } 2) = 18$ (in cells) $< Td = 20$ (clock cycles), as shown in Fig. 2(c). The route for Net 2 induces two intersections with the routes for Net 4 and Net 5 respectively. Therefore, the desirable edge-disjoint droplet-route set with total 63 cells is finally obtained. All routes satisfy both timing and fluidic constraints, as shown in Fig. 2(c). Note that for droplets D_1 and D_3 in Route 4 and Route 5 respectively, fluidic constraint might be violated if they start moving at the same time to their destinations. However, based on the modification rules, we force D_1 to stay in the current location until D_2 is transported to the mixing site and mixed with it, at the same time we continue moving D_3 to its destination, thereby overriding the constraint violation. Since the delay for Route 4 is determined by the transportation time for D_2 , no extra delay is introduced during this process. The droplet-transportation time for all the routes in this sub-problem is 19 clock cycles.

We compare the routing results obtained using the proposed method for sub-problem 3 with a baseline method that utilizes the modified Lee algorithm in [14] to solve a sub-problem without considering cross-contamination between different routes. The baseline method uses four route intersections, two more than that obtained by the proposed method, as shown in Fig. 2(d). We refer to an intersection of two routes as a cross-contamination site. The number of cross-contamination sites can be used to evaluate the likelihood of cross-contamination for a set of routes in a sub-problem.

Consider an intersection of two droplet routes in a sub-problem. When a droplet passes through the intersection, a wash droplet is dispensed from the wash reservoir and transported via this site to the waste reservoir. (We assume, without loss of generality, that one wash droplet suffices; multiple wash droplets can also be considered.) After that, the other droplet can be transported via this cleaned site. Therefore, cross-contamination between the above droplets is avoided. For the routing results in Fig. 2(c) obtained using the proposed method, at first, droplets are

TABLE I
COMPARISON OF THE PROPOSED METHOD WITH THE BASELINE METHOD FOR
EXAMPLE 1 (CONTAMINATION WITHIN SUBPROBLEM).

Sub-problem no.	Result for proposed method				Result for baseline method			
	N_{cs}	N_{cell}	T_r	T_{rw}	N_{cs}	N_{cell}	T_r	T_{rw}
1	0	46	17	17	1	39	17	55
2	1	41	12	47	1	41	12	47
3	2	63	19	59	4	57	18	97
4	0	6	5	5	0	6	5	5
5	0	63	17	17	3	55	15	55
6	1	57	16	51	3	55	16	52
7	0	54	18	18	1	51	18	57
8	0	24	13	13	0	24	13	13
9	0	24	14	14	0	24	14	14
10	0	14	13	13	0	14	13	13
11	0	15	14	14	0	15	14	14
Total	4	407	158	268	13	381	155	422

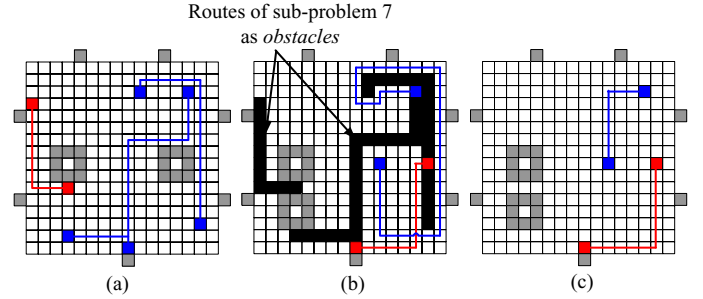


Fig. 3. Routes for sub-problem 7 with or without wash operation: (a) disjoint-routing solution for sub-problem 7; (b) disjoint routes for sub-problem 8 when no wash operation is performed before it; (c) disjoint routes for sub-problem 8 when a wash operation is performed.

transported along Routes 1 and 3-5 at the same time. After that, two wash droplets are dispensed from the wash reservoirs and transported via the two cross-contamination sites to the waste reservoir, respectively. After the wash operation, the droplet is transported along Route 2 via the two sites that have already been cleaned. The maximum droplet-transportation time with washing steps is 59 clock cycles. For the routing results in Fig. 2(d) obtained using the baseline method, the maximum droplet-transportation time with washing steps (97 clock cycles) is much higher than that for the proposed method. This is because two wash operations must be inserted sequentially during droplet routing to clean the residue on the four cross-contamination sites.

In Table I, for each sub-problem, we compare for the two methods, the number of cross-contamination sites (N_{cs}), the number of cells used by the nets (N_{cell}), and the maximum droplet-transportation time for the nets with and without wash steps in this sub-problem, i.e., T_{rw} and T_r . $N_{cs} = 0$ denotes the fact the routes obtained are vertex-disjoint. The results show that the proposed method significantly reduces the overall routing time, especially when wash droplets must be routed.

In Fig. 3, we illustrate the influence of cross-contamination across two sub-problems on the droplet routing. Fig. 3(a) shows the disjoint routing solution for sub-problem 7. If no wash operation is performed between sub-problems 7 and 8, the routes for all nets in sub-problem 7 are treated as obstacles for sub-problem 8, as shown in Fig. 3(b). The disjoint routes for sub-problem 8 are also shown in Fig. 3(b). The maximum transportation time T_8 is 76 clock cycles. If a wash operation

TABLE II
PARAMETERS FOR OPTIMIZATION MODEL (IN CLOCK CYCLES).

Sub-problem no.	Proposed method			Baseline method		
	T	T^*	T_w	T	T^*	T_w
1	17	17	0	55	55	0
2	10000	47	17	10000	47	17
3	10000	59	12	10000	97	12
4	5	5	19	10000	5	18
5	17	17	5	55	55	5
6	10000	51	17	10000	52	15
7	39	18	16	66	57	16
8	76	13	18	76	13	18
9	14	14	13	14	14	13
10	13	13	14	13	13	14
11	14	14	13	14	14	13

TABLE III
COMPARISON OF THE PROPOSED METHOD WITH THE BASELINE METHOD FOR EXAMPLE 2 (CONTAMINATION WITHIN SUBPROBLEM).

Sub-problem no.	Result for proposed method				Result for baseline method			
	N_{cs}	N_{cell}	T_r	T_{rw}	N_{cs}	N_{cell}	T_r	T_{rw}
1	0	44	20	20	1	40	18	56
2	0	44	19	19	1	42	19	56
3	0	38	19	19	2	36	19	55
4	0	8	7	7	0	8	7	7
5	0	75	17	17	2	66	15	52
6	0	31	14	14	1	30	14	50
7	0	14	13	13	0	14	13	13
8	0	20	11	11	0	20	11	11
9	0	30	13	13	1	28	13	48
10	0	17	16	16	0	17	16	16
11	0	11	10	10	0	11	10	10
Total	0	332	159	159	8	312	155	374

is performed, the routes for sub-problem 8 do not consider the routes for sub-problem 7 to be obstacles, as shown in Fig. 3(c). The maximum transportation time T_8^* is now reduced to 13 clock cycles. Moreover, since the routes for sub-problem 2 require a large number of cells, we cannot obtain feasible routes for sub-problem 3 if a wash operation is not performed between sub-problems 2 and 3. For such cases, we set T_i to be a large value, i.e., 10000, for this work.

Table II lists the values of T_i , T_i^* and T_{wi} , $1 \leq i \leq 11$, for each sub-problem using both the proposed disjoint-routing and the baseline methods. For the proposed method, using the optimization model presented in Section IV, we obtain the minimum routing time for all the sub-problems to be 348 clock cycles. A wash operation must be performed before sub-problems 2, 3 and 6-8 for this optimal routing plan. For the baseline method, the minimum routing time for all the sub-problems is much higher, 511 clock cycles, and we have to perform a wash operation before sub-problems 2, 3, 4, 6 and 8.

B. Example 2: DNA sequencing

In the DNA sequencing method introduced in [11], the DNA fragment of interest is incubated with enzymes. Four nucleotides are added to the DNA fragment. A nucleotide binds with the growing DNA strand only if it is complementary to the base in the strand. The amount of light produced in the luciferase-catalyzed reaction can readily be estimated by a suitable light-sensitive device such as a luminometer CCD camera. The assay protocol for DNA sequencing can be modeled by a sequencing

graph, and the schedule for assay operations, as well as resource binding and module placement can be obtained via architectural-level synthesis [3].

The routing problem is decomposed into eleven sub-problems. In Table III, for each sub-problem, we compare values for N_{cs} , N_{cell} , T_r and T_{rw} for the disjoint-routing method and the cross-contamination-oblivious baseline method. In sub-problems 4, 7-8 and 10-11, the baseline method yields the same values for N_{cs} , N_{cell} , T_r and T_{rw} as the proposed method, since the two methods generate the same routes.

Using the optimization method presented in Section IV, we obtain the minimum routing time for all the sub-problems to be 244 clock cycles. For this optimal routing plan including wash operations, a wash operation must be performed before sub-problems 2, 3 and 6. For the baseline method, the minimum routing time for all the sub-problems is as high as 455 clock cycles, and we have to perform a wash operation before sub-problems 2, 3, 6 and 8.

VI. CONCLUSION

We have presented a droplet-routing and optimization method for digital microfluidics that avoids cross-contamination. This method targets disjoint droplet routes and minimizes the number of cells used for droplet routing. An optimization method has also been presented to minimize the number of wash operations that must be used between successive routing steps. Two real-life bioassay applications have been used to evaluate the effectiveness of the proposed method.

REFERENCES

- [1] R. B. Fair et al., "Chemical and biological applications of digital-microfluidic devices", *IEEE Design & Test of Computers*, vol. 24, pp. 10-24, 2007.
- [2] R. B. Fair et al., "Electrowetting-based on-chip sample processing for integrated microfluidics", *Proc. IEDM*, pp. 32.5.1C32.5.4, 2003.
- [3] K. Chakrabarty and F. Su, *Digital Microfluidic Biochips: Synthesis, Testing, and Reconfiguration Techniques*, CRC Press, Boca Raton, FL, 2006.
- [4] H. Moon et al., "An integrated digital microfluidic chip for multiplexed proteomic sample preparation and analysis by MALDI-MS", *Lab on a Chip*, vol. 6, pp. 1213-1219, 2006.
- [5] K. F. Bohringer, "Modeling and controlling parallel tasks in droplet-based microfluidic systems", *IEEE Trans. CAD*, vol. 25, pp. 334-344, 2006.
- [6] M. E. Kramer and J. van Leeuwen, "The complexity of wire routing and finding the minimum area layouts for arbitrary VLSI circuits", in *Advances in Computing Research 2: VLSI Theory*, JAI Press, London, 1984.
- [7] J. F. Lynch, "The equivalence of theorem proving and the interconnection problem", *ACM SIGDA Newsletter*, vol. 5, pp. 31-36, 1975.
- [8] S. Sait and H. Youssef, *VLSI Physical Design Automation: Theory and Practice*, IEEE Press, NY, 1995.
- [9] M. Cho and D. Z. Pan, "A high-performance droplet router for digital microfluidic biochips", *Proc. Int. Symp. Physical Design*, 2008.
- [10] A. J. Ricketts et al., "Priority scheduling in digital microfluidics-based biochips", *Proc. DATE Conf.*, pp. 329-334, 2006.
- [11] M. Ronaghi et al., "DNA Sequencing: a sequencing method based on real-time pyrophosphate", *Science*, vol. 281, pp. 363-365, 1998.
- [12] F. Su and K. Chakrabarty, "High-level synthesis of digital microfluidic biochips", *ACM J. Emerging Tech. Computing Sys.*, vol. 3, pp. 16.1-16.32, 2008.
- [13] F. Su and K. Chakrabarty, "Unified high-level synthesis and module placement for defect-tolerant microfluidic biochips", *Proc. DAC*, pp. 825-830, 2005.
- [14] F. Su, W. Hwang, and K. Chakrabarty, "Droplet routing in the synthesis of digital microfluidic biochips", *Proc. DATE*, pp. 323-328, 2006.
- [15] P.-H. Yuh et al., "BioRoute: A network flow based routing algorithm for digital microfluidic biochips", *Proc. ICCAD*, pp. 752-757, 2007.
- [16] T. Xu and K. Chakrabarty, "Integrated droplet routing in the synthesis of microfluidic biochips", *Proc. DAC*, pp. 948-953, 2007.