

A Two-Stage ILP-Based Droplet Routing Algorithm for Pin-Constrained Digital Microfluidic Biochips

Tsung-Wei Huang
Department of Computer Science and
Information Engineering
National Cheng Kung University
Tainan, Taiwan
electron@eda.csie.ncku.edu.tw

Tsung-Yi Ho*
Department of Computer Science and
Information Engineering
National Cheng Kung University
Tainan, Taiwan
tyho@csie.ncku.edu.tw

ABSTRACT

With the increasing design complexities, the design of pin-constrained digital microfluidic biochips (PDMFBs) is of practical importance for the emerging marketplace. However, the solution of current pin-count aware technique is inevitably limited by simply adopting it after the droplet routing stage. In this paper, we propose the *first* droplet routing algorithm for PDMFBs that can integrate pin-count technique with droplet routing stage. Furthermore, our algorithm is capable of *simultaneously* minimizing the number of control pins, the number of used cells, and the latest arrival time. We first present a basic integer linear programming (ILP) formulation to optimally solve the droplet routing problem for PDMFBs with simultaneous multi-objective optimization. Due to the complexity of this ILP formulation, we also propose a two-stage technique of global routing followed by *incremental* ILP-based routing to reduce the solution space. To further reduce the runtime, we present a *deterministic* ILP formulation that casts the original routing optimization problem into a decision problem, and solve it by a binary solution search method that searches in logarithmic time. Extensive experiments demonstrate that in terms of the number of the control pins, the number of the used cells, and the latest arrival time, we acquire much better achievement than all the state-of-the-art algorithms in any aspect.

Categories and Subject Descriptors: B.7.2 [Integrated Circuits]: Design Aids - Layout, Place and Route

General Terms: Algorithms, Designs

Keywords: Biochip, ILP, microfluidics, routing

1. INTRODUCTION

As the microfluidics technology advances, digital microfluidic biochips (DMFBs) have attracted much attention recently. Compared with the conventional laboratory experiment procedures, which are usually cumbersome and expensive, these miniaturized and automated DMFBs show numerous advantages such as high portability, high throughput, high sensitivity, minimal human intervention, and low sample/reagent volume consumption.

Nonetheless, while more bioassays are executed concurrently

*This work was partially supported by the National Science Council of Taiwan ROC under Grant No. NSC 96-2220-E-006-013.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

ISPD'10, March 14–17, 2010, San Francisco, California, USA.

Copyright 2010 ACM 978-1-60558-920-6/10/03 ...\$10.00.

on the digital microfluidic platforms, the complexity of the system and the number of the electrodes are bound to increase steadily [8]. Recently, a DMFB that embeds more than 600,000 $20\ \mu\text{m}$ by $20\ \mu\text{m}$ electrodes has been demonstrated [2]. Thus, the design of droplet control scheme with pin minimization is of great practical importance for the pin-constrained DMFBs (PDMFBs).

In the most common droplet control scheme, each electrode is directly addressed and controlled by a dedicated control pin, which allows each electrode to be individually activated. In this paper, we refer to these types of DMFBs as *direct-addressing* DMFBs. The previous droplet routing algorithms mainly focus on direct-addressing DMFBs [3, 4, 5, 7, 9, 11]. This scheme maximizes the freedom of the droplet manipulation, but it suffers from the major deficiency that the number of control pins rapidly increases as the system complexity increases. Moreover, a large number of control pins necessitate multiple PCB layers, which potentially raise the price of production cost. Pin-constrained design for direct-addressing DMFBs was addressed in [8]. However, this method is for the exclusive use of some target biofluidic applications, which is not applicable to large-scale PDMFBs.

Recently, a novel *broadcast-addressing* design scheme for PDMFBs has been proposed to overcome the drawbacks of the previous two schemes [10]. This scheme provides high throughput for bioassays and reduces the number of control pins by identifying and connecting them with “compatible” activation sequences. Another advantage of the broadcast-addressing scheme is that it provides the maximum freedom of droplet movement as the direct-addressing scheme. The compatible activation sequences can be derived by applying minimal clique partitioning to electrodes. However, the minimal clique partitioning problem is known to be NP-hard. Furthermore, the solution is inevitably limited by simply using the direct-addressing-based routing result as the input to apply the broadcast-addressing scheme. Therefore, the traditional broadcast-addressing scheme may result in suboptimal solutions.

To overcome these drawbacks, we propose the *first* ILP-based droplet routing algorithm that *simultaneously* takes the droplet routing and the broadcast-addressing schemes into consideration for PDMFBs. The main challenge of this routing problem is to derive different constraints into ILP formulations while ensuring correct droplet movement and minimizing the number of pins. Different from the aforementioned works, our algorithm, by using well-formulated constraints in ILP formulations, is capable of *simultaneously* minimizing the number of the control pins, the number of the used cells, and the latest arrival time.

1.1 Our Contribution

In this paper, we propose the *first* droplet routing algorithm for PDMFBs that *simultaneously* minimizes the number of control pins, the number of used cells, and the latest arrival time. We first present a basic integer linear programming (ILP) formulation to optimally solve the droplet routing problem for PDMFBs. Due to its complexity, we also propose a two-stage technique of global routing followed by incremental ILP-based routing to reduce the solution space effectively. Our algorithm divide the original routing

ing problem to global routing paths spatially to reduce the solution space of ILP formulations. In this way, the original problem is reduced to a manageable size, then we can practically apply an incremental ILP-based method to finding a high-quality solution within reasonable CPU time. To achieve further efficiency, we propose a *deterministic* ILP formulation that casts the original optimization into a decision problem and solve it by a logarithmic search technique. The major contributions of this paper include the followings:

- We propose the *first* droplet routing algorithm that considers the droplet routing and the broadcast-addressing scheme *simultaneously* for PDMFBs. In contrast with the previous works that start with an initial direct-addressing-based routing result, our algorithm has higher flexibility to solve the droplet routing problem on PDMFBs globally.
- Unlike the previous works that only minimize the number of control pins, our algorithm can *simultaneously* minimize not only the number of the control pins but also the number of used cells and the latest arrival time, which is attributed to the well-founded formulation of the constraints into our ILP formulations.
- To tackle the complexity of the basic ILP formulations, we propose a two-stage routing scheme of global routing followed by incremental ILP-based routing. For the basic ILP, the problem instance is whole 2D plane and it handles all droplets simultaneously. For our two-stage ILP, the problem instance is reduced to global routing paths and the droplets are routed in incremental manner that reduce the solution space significantly. Therefore, our algorithm can obtain a high-quality solution within reasonable CPU time.
- To further reduce the runtime, we present a *deterministic* ILP formulation that casts the original routing optimization problem into a decision problem, and then solves it by a binary solution search method that searches in logarithmic time.

Compared with the direct-addressing and the broadcast-addressing schemes, the extensive experiments demonstrate that in terms of the number of the control pins, the number of the used cells and the latest arrival time, we acquire much better achievement than all the current state-of-the-art algorithms in any aspect.

The remainder of this paper is organized as follows. Section 2 details routing on broadcast-addressing DMFBs and formulates the droplet routing problem. Section 3 presents the basic ILP formulations for droplet routing problem. Section 4 details the two-stage ILP routing scheme, while Section 5 shows the experimental results. Finally, concluding remarks are provided in Section 6.

2. ROUTING ON BROADCAST-ADDRESSING DMFBs

In this section, we first show the broadcast-addressing scheme. Then we present the problem formulation of the droplet routing problem for PDMFBs.

2.1 Broadcast-Addressing Scheme

To execute a specific bioassay, the routing and the operation scheduling for droplets are programmed into a microcontroller to drive the electrodes. The information of routing and scheduling is stored in the form of electrode activation sequences. Each bit in the sequence represents the activation status of the electrode in a specific time step, and can be represented as activated (“1”), deactivated (“0”), or don’t care (“X”). A don’t care signal represents that the input signal of electrode can be either activated or deactivated, which doesn’t change the routing scheme. An example is shown in Figure 1 (a). When the droplet d_1 moves from (0,6) to (1,6), the electrode in cell (1,6) must be assigned “1” and the cell (0,6) and (2,6) must be assigned “0”. In this time step, the cell (3,6) is treated as don’t care that we can assign “1” or “0” to this cell which has no impact on d_1 ’s movement. We use the three value “1”, “0”, and “X” to represent the electrode activation sequences for a bioassay. As shown in Figure 1 (a), when droplet d_1 moves from cell (0,6) to cell (3,6) within time 0

to 3, the corresponding activation sequences of cell (0,6) to cell (3,6) (noted as c_1 to c_4) can be represented as “10XX”, “010X”, “0010”, and “X001”. By carefully replacing the don’t care terms in c_4 , we can identify c_4 with c_1 in this activation sequence “1001”. We refer to the sequences of c_1 and c_4 as “compatible sequence”. In broadcast-addressing scheme, the corresponding electrodes of c_1 and c_4 can be connected to a single control pin. Therefore, compared with direct-addressing scheme, the number of control pins can be significantly reduced. However, with increased design complexities, the solution is inevitably limited by using the direct-addressing-based routing result as the input to apply the broadcast-addressing scheme [10]. In Figure 1 (a), if we only adopt the broadcast-addressing scheme to a given routed result, we need 15 control pins to execute this bioassay. But if we simultaneously consider the routing and the broadcast-addressing scheme, as shown in Figure 1 (b), we only need 13 control pins for this bioassay. In addition to minimizing the number of control pins in PDMFBs, it is desirable to minimize the number of used cells and the latest arrival time for fast bioassay execution and better reliability. In Figure 1 (c), our droplet routing algorithm can concurrently address these optimization issues to minimize the number of control pins, the number of used cells, and the latest arrival time, thereby achieving significantly better routing solution. Therefore, in addition to simultaneously considering the droplet routing and the broadcast-addressing scheme, it is desirable to minimize the number of control pins, the number of used cells, and the latest arrival time for PDMFBs.

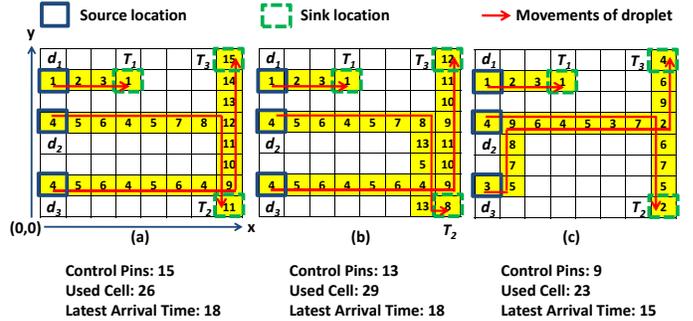


Figure 1: Example of an 8×8 DMFB with three droplets. (a) apply the broadcast-addressing scheme to a routing result. (b) apply the droplet routing and broadcast-addressing scheme simultaneously. (c) apply the droplet routing and broadcast-addressing scheme simultaneously with minimizing the number of control pins, the number of used cells, and the latest arrival time.

2.2 Problem Formulation

In addition to minimizing the number of control pins in PDMFBs, it is desirable to minimize the number of unit cells that are used during routing. Since a unit cell of a DMFB can be defective due to manufacturing or environmental issues, using a smaller number of unit cells for routing can be beneficial for robustness. Furthermore, it is desirable to minimize the latest arrival time among all droplets to achieve fast bioassay execution and better reliability.

In addition to the objectives of droplet routing, there are two routing constraints in droplet routing: the fluidic constraint and the timing constraint. The fluidic constraint is used to avoid the unexpected mixtures between two droplets of different nets during their transportation and it can further be divided into the static and dynamic fluidic constraints [7]. Let d_i at cell (x_t^i, y_t^i) and d_j at cell (x_t^j, y_t^j) denote two independent droplets at time t . Then, the following constraints should be satisfied for any t during routing:

- **Static constraint:** $|x_t^i - x_t^j| > 1$ or $|y_t^i - y_t^j| > 1$.
- **Dynamic constraint:** $|x_{t+1}^i - x_t^j| > 1$ or $|y_{t+1}^i - y_t^j| > 1$ or $|x_t^i - x_{t+1}^j| > 1$ or $|y_t^i - y_{t+1}^j| > 1$.

The static fluidic constraint states that the minimum spacing between two droplets is one cell for any t during routing. The dynamic fluidic constraint states that the activated cell for d_i cannot be adjacent to d_j . The reason is there can be more than one activated neighboring cell for d_j . Therefore, we may have an unexpected mixing between d_i and d_j . Besides the fluidic constraint, there exists the timing constraint. The timing constraint specifies the maximum arrival time of a droplet from its source to sink. The droplet routing problem for the PDMFBs can be formulated as follows:

Input: A netlist of n droplets $D = \{d_1, d_2, \dots, d_n\}$, the locations of blockages, and the timing constraint T_{max} .

Constraint: Both fluidic and timing constraints are satisfied.

Objective: Route all droplets from their source cells to their sink cells while minimizing (1) the number of pins, (2) the number of used cells, and (3) the latest arrival time among all droplets.

3. ILP FORMULATION FOR DROPLET ROUTING

In this section, we propose the *first* basic ILP formulation that considers the droplet routing and the broadcast-addressing scheme *simultaneously* for PDMFBs. We show how the ILP optimizes droplet routing with the consideration of three objective minimizations. For the sake of brevity, we first focus on 2-pin net routing.

3.1 Basic ILP Formulation

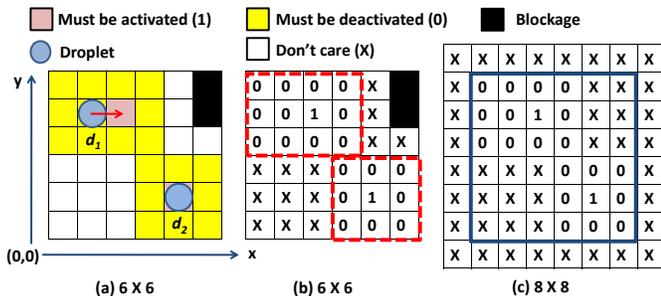


Figure 2: Modeling of electrode activation constraint. (a) A droplet d_1 moves while the other droplet d_2 stalls. (b) The corresponding activations of electrodes. (c) The one bounding box enlarged array.

Unlike traditional very large scale integration routing, in addition to sharing the routing path in a time-multiplexed fashion, our ILP formulations address the issue of scheduling droplets under practical constraints imposed by the fluidic property and timing restriction. Furthermore, our ILP formulations simultaneously consider the electrode sharing between different droplet routes by adopting the broadcast-addressing scheme. Then, we concurrently formulate the three minimizations into our objective functions for the integration of droplet routing.

One of the most difficult challenges of this problem is to model the electrode activation constraint into an ILP formulation, considering the activated (“1”), deactivated (“0”), and don’t care (“X”) activation terms. To successfully obtain each cell’s activation sequence, we add “must” restriction to this constraint in the formulation. As shown in Figure 2 (a), when droplet 1 moves from cell (1,4) to cell (2,4) at time t , all neighboring cells of cell (1,4) and cell (2,4) must be deactivated, except for the cell (2,4), which must be activated. When droplet 2 stalls at its original cell at time t , all the neighboring cells of cell (4,1) must be deactivated, while the cell (4,1) must be activated to hold this droplet [8]. Those cells that have no impact on droplet transportation are regarded as don’t care terms which can be assigned “1” or “0” [10]. Figure 2 (b) describes the corresponding electrode activation. To model

this constraint, we use the notation (x_t^i, y_t^i) to represent the location of droplet d_i at time t . Therefore, the electrode activation constraint can be formulated in the following rules:

- **EC-Rule I:** If a droplet d_i moves from cell (x_{t-1}^i, y_{t-1}^i) to cell $(x_t^i, y_t^i) \in E_5^C(x_{t-1}^i, y_{t-1}^i)$, all the cells $(x', y') \in \{E_9^C(x_{t-1}^i, y_{t-1}^i) \cup E_9^C(x_t^i, y_t^i)\}$ must be deactivated at time t , except for the cell (x_t^i, y_t^i) , which must be activated at time t .
- **EC-Rule II:** If a droplet d_i stalls at time t , the exact number of must-be-deactivated cells is 8; otherwise, if d_i moves to the four adjacent cells at time t , the exact number of must-be-deactivated cells is 11.
- **EC-Rule III:** The cells that have no impact on droplets transportation are don’t care terms.

Because of the blockages and the boundary restriction in the microfluidic array, it is hard to directly apply the three rules to the ILP formulations in the cell set C . For example, if a droplet d_i stalls at the location (0,0) within time $t - 1$ to t in Figure 2 (a), due to the boundary restriction of microfluidic array, the exact number of must-be-deactivated cells is 3 instead of 8. In other words, we may need extra constraints and variables to determine the exact number in EC-Rule II, which significantly increases the complexities of the ILP formulations. Therefore, we apply the three rules on the microfluidic array which is enlarged one bounding box of the original microfluidic array to solve this problem. As shown in Figure 2 (c), the cells inside the 8×8 array belong to the cell set B . As the example mentioned earlier, to hold the droplet d_i at cell (0,0) at time t , the exact number of must-be-deactivated cells in B is 8. In this way, we can achieve the three rules without increasing the size of electrode activation constraint.

Another major challenge in the routing problem is to model the broadcast constraint into an ILP formulation. Each activation sequence may contain several don’t care terms, which can be replaced by “1” or “0”. This feature increases the solution space of our ILP. In other words, a naïve formulation may increase the size of constraints and the complexity of ILP. Therefore, we propose the three major rules to tackle the broadcast constraint as follows:

- **BC-Rule I:** Two activation sequences are compatible if and only if the corresponding binary values are the same.
- **BC-Rule II:** If the activation sequences of two cells are incompatible, we cannot broadcast the two cells with the same control pin.
- **BC-Rule III:** If the activation sequences of two cells are compatible, we can broadcast the two cells with the same control pin *or not*.

BC-Rule I states the compatible relation between two activation sequences. Both BC-Rule II and III describe the broadcast rules for two cells.

In the following subsections, we introduce the objective function and constraints of our basic ILP formulations. The notations used in our ILP formulations are shown in Table I.

3.2 Objective Function

Our goal is to minimize the number of control pins, the number of used cells, and the latest arrival time. Therefore, the objective function is defined in the following equation:

$$\text{Minimize} : \alpha \cdot \sum_{p=1}^{P_{max}} up(p) + \beta \cdot \sum_{(x,y) \in C} uc(x,y) + \gamma \cdot T_l \quad (1)$$

where α , β , and γ are set to one as the default value.

3.3 Constraints

There are total ten constraints in our basic ILP formulations.

1. **Source requirement:** All droplets are at their source location at time zero. Therefore, the source requirement can be represented in the following constraint:

TABLE I: NOTATIONS USED IN OUR BASIC ILP FORMULATION.

D	set of droplets
C	set of available cells
B	set of cells inside one enlarged bounding box of a DMFB
T_{max}	constraint for maximum completion time
P_{max}	constraint for maximum available control pins
$E_5^C(x, y)$	set of cell (x, y) and its four adjacent cells in C
$E_8^C(x, y)$	set of cell (x, y) 's eight neighboring cells in C
$E_9^C(x, y)$	set of cell (x, y) and its eight neighboring cells in C
$E_8^B(x, y)$	set of cell (x, y) 's eight neighboring cells in B
(s_x^i, s_y^i)	location of the source cell of droplet d_i
(t_x^i, t_y^i)	location of the sink cell of net n_i
$c(i, x, y, t)$	a 0-1 variable represents that droplet d_i locates at cell (x, y) at time t
T_l	latest arrival time for routing
$uc(x, y)$	a 0-1 variable represents that cell (x, y) is used
$st(i, t)$	a 0-1 variable represents that d_i stalls from time $t - 1$ to t
$a_0(i, x, y, t)$	a 0-1 variable represents that cell (x, y) must be deactivated in controlling d_i 's movement at time t
$a_1(i, x, y, t)$	a 0-1 variable represents that cell (x, y) must be activated in controlling d_i 's movement at time t
$a_X(i, x, y, t)$	a 0-1 variable represents that cell (x, y) is don't care in controlling d_i 's movement at time t
$A_0(x, y, t)$	a 0-1 variable represents that cell (x, y) must be deactivated in total movements control at time t
$A_1(x, y, t)$	a 0-1 variable represents that cell (x, y) must be activated in total movements control at time t
$A_X(x, y, t)$	a 0-1 variable represents that cell (x, y) is don't care in total movements control at time t
$as(x, y, t)$	activation sequence of cell (x, y) at time t
$cmp(x_1, y_1, x_2, y_2)$	a 0-1 variable represents the activation sequences of cell (x_1, y_1) and (x_2, y_2) are compatible
$cp(x, y, p)$	a 0-1 variable represents that the cell (x, y) is controlled by pin p
$up(p)$	a 0-1 variable represents that pin p is used

$$c(i, s_x^i, s_y^i, 0) = 1, \forall d_i \in D \quad (2)$$

2. *Sink requirements:* All droplets must reach their sinks within timing constraint. Once a droplet reaches its sink, it remains there. Therefore, the sink requirements can be represented in the following constraints:

$$\sum_{t=0}^{T_{max}} c(i, t_x^i, t_y^i, t) \geq 1, \forall d_i \in D \quad (3)$$

$$c(i, t_x^i, t_y^i, t) - c(i, t_x^i, t_y^i, t+1) \leq 0, \forall d_i \in D, 0 \leq t < T_{max} \quad (4)$$

3. *Exclusivity constraint:* Each droplet has only one location at each time step. Therefore, the exclusivity constraint can be represented in the following constraint:

$$\sum_{(x,y) \in C} c(i, x, y, t) = 1, \forall d_i \in D, 0 \leq t \leq T_{max} \quad (5)$$

4. *Computation of latest arrival time:* If a droplet reaches its sink at time t , then the time it reaches its sink can be computed as t times the difference of $c(i, t_x^i, t_y^i, t)$ and $c(i, t_x^i, t_y^i, t-1)$. Therefore, the computation of latest arrival time can be represented as follows:

$$t \cdot (c(i, t_x^i, t_y^i, t) - c(i, t_x^i, t_y^i, t-1)) \leq T_l, \quad \forall d_i \in D, 0 < t \leq T_{max} \quad (6)$$

5. *Computation of total used cells:* A cell (x, y) is used if a droplet ever located at this cell before. Otherwise, if there is no droplet locating at the cell (x, y) during the whole

bioassay execution, the cell is un-used. Therefore, the above two constraints can be represented as follows:

$$uc(x, y) \geq c(i, x, y, t), \forall d_i \in D, (x, y) \in C, 0 \leq t \leq T_{max} \quad (7)$$

$$uc(x, y) \leq \sum_{d_i \in D} \sum_{t=0}^{T_{max}} c(i, x, y, t), (x, y) \in C \quad (8)$$

6. *Droplet movement constraint:* A droplet can have only five possible movements; stall or move to four adjacent cells from t to $t+1$. Therefore, the movement constraint can be represented in the following constraint:

$$c(i, x, y, t) \leq \sum_{(x', y') \in E_5^C(x, y)} c(i, x', y', t+1), \quad \forall d_i \in D, (x, y) \in C, 0 \leq t < T_{max} \quad (9)$$

7. *Fluidic constraints:* As described in Section 2, there are two fluidic constraints: static and dynamic fluidic constraints. Static fluidic constraint states the minimum spacing between two droplets must be one cell. In other words, there are no other droplets in the 3×3 region centered by a droplet. Therefore, the static fluidic constraint can be represented in the following constraints:

$$c(i, x, y, t) + \sum_{(x', y') \in E_9^C(x, y)} c(j, x', y', t) \leq 1, \quad \forall d_i, d_j \in D, d_i \neq d_j, (x, y) \in C, 0 \leq t \leq T_{max} \quad (10)$$

To prevent unexpected mixing during droplet movement, dynamic fluidic constraint requires that at time $t+1$, d_i cannot move to the cell (x, y) , which is the neighboring cells of d_j 's location at time t . Therefore, the dynamic fluidic constraint can be represented in the following constraint:

$$c(i, x, y, t+1) + \sum_{(x', y') \in E_9^C(x, y)} c(j, x', y', t) \leq 1, \quad \forall d_i, d_j \in D, d_i \neq d_j, (x, y) \in C, 0 \leq t < T_{max} \quad (11)$$

8. *Electrode constraints:* EC-Rule I states that if droplet locates at cell (x, y) at time t , this cell (x, y) must be activated. Therefore, this activated rule can be represented in the following constraints:

$$a_1(i, x, y, t) = c(i, x, y, t), \forall d_i \in D, (x, y) \in C, 0 \leq t \leq T_{max} \quad (12)$$

$$\sum_{(x,y) \in B} a_1(i, x, y, t) = 1, \forall d_i \in D, 0 \leq t \leq T_{max} \quad (13)$$

Note that the constraint (13) states the exclusivity constraint in cell set B . The following two constraints state the deactivated condition in EC-Rule I.

$$\sum_{(x', y') \in E_8^B(x, y)} a_0(i, x', y', t) \geq 8 \cdot c(i, x, y, t), \quad \forall d_i \in D, (x, y) \in C, 0 \leq t \leq T_{max} \quad (14)$$

$$\sum_{(x', y') \in E_8^B(x, y)} a_0(i, x', y', t) \geq 7 \cdot c(i, x, y, t-1), \quad \forall d_i \in D, (x, y) \in C, 0 < t \leq T_{max} \quad (15)$$

Note that as shown in Figure 2 (b), constraints (14) and (15) only determine the number of cells which must be deactivated in the dash-line area (e.g., the lower bound number of cells that must be deactivated). For example, for droplet

1, we can assign “0” to the cell (0,0), which still satisfies the constraint (14) and (15), but violates the “must be deactivated” condition. To tackle this problem, we use $st(i, t)$ to represent that droplet d_i stalls within time $t - 1$ to t , and use EC-rule II to determine the exact number of cells which must be deactivated.

$$st(i, t) \geq c(i, x, y, t) + c(i, x, y, t - 1) - 1, \quad \forall d_i \in D, (x, y) \in C, 0 < t \leq T_{max} \quad (16)$$

$$st(i, 0) = 1, \forall d_i \in D \quad (17)$$

$$\sum_{(x, y) \in B} a_0(i, x, y, t) = 8 + 3 \cdot (1 - st(i, t)), \quad \forall d_i \in D, 0 \leq t \leq T_{max} \quad (18)$$

Note that even if constraint (16) makes $st(i, t)$ to be 0 or 1 when droplet d_i doesn't stall within time $t - 1$ to t . Due to the constraint (14) and (15), the lower bound of the number of cells that must be deactivated in B is 11 when droplet d_i moves. Therefore, to satisfy constraints (14), (15), and (18), the value of $st(i, t)$ is restricted to be 0 only.

9. *Activation sequence constraints:* Due to the electrode constraint, we obtain the electrode activation for each droplet at any time t . We use the following constraints to derive the global activation sequences for total movements control. Note that the “must” condition still holds.

For the “must” be activated cells:

$$A_1(x, y, t) = \sum_{d_i \in D} a_1(i, x, y, t), (x, y) \in B, 0 \leq t \leq T_{max} \quad (19)$$

For the “must” be deactivated cells:

$$A_0(x, y, t) \geq a_0(i, x, y, t), \forall d_i \in D, (x, y) \in B, 0 \leq t \leq T_{max} \quad (20)$$

$$A_0(x, y, t) \leq \sum_{d_i \in D} a_0(i, x, y, t), (x, y) \in B, 0 \leq t \leq T_{max} \quad (21)$$

For the don't cares cells (EC-Rule III):

$$A_0(x, y, t) + A_1(x, y, t) + A_X(x, y, t) = 1, \quad (x, y) \in B, 0 \leq t \leq T_{max} \quad (22)$$

Since the don't care term can be replaced by “1” or “0”, we use the following constraints to obtain all possible activation sequences of each cell.

$$0 \cdot A_0(x, y, t) + 1 \cdot A_1(x, y, t) + 0 \cdot A_X(x, y, t) \leq as(x, y, t), \quad (x, y) \in B, 0 \leq t \leq T_{max} \quad (23)$$

$$0 \cdot A_0(x, y, t) + 1 \cdot A_1(x, y, t) + 1 \cdot A_X(x, y, t) \geq as(x, y, t), \quad (x, y) \in B, 0 \leq t \leq T_{max} \quad (24)$$

10. *Broadcast constraints:* The three broadcast rules mentioned earlier can be represented in the following constraints:

$$1 - cmp(x_1, y_1, x_2, y_2) \geq as(x_1, y_1, t) - as(x_2, y_2, t), \quad (x_1, y_1), (x_2, y_2) \in C, 0 \leq t \leq T_{max} \quad (25)$$

$$1 - cmp(x_1, y_1, x_2, y_2) \geq as(x_2, y_2, t) - as(x_1, y_1, t), \quad (x_1, y_1), (x_2, y_2) \in C, 0 \leq t \leq T_{max} \quad (26)$$

$$cp(x_1, y_1, p) + cp(x_2, y_2, p) \leq cmp(x_1, y_1, x_2, y_2) + 1, \quad (x_1, y_1), (x_2, y_2) \in C, 1 \leq p \leq P_{max} \quad (27)$$

where constraint (25) and (26) represent the BC-Rule I, and (27) represents the BC-Rule II and III. Note that if two cells (x_1, y_1) and (x_2, y_2) are compatible, the value of

$cmp(x_1, y_1, x_2, y_2)$ can be 0 or 1, which still holds due to the BC-Rule III. The following three constraints state the computation of minimized the number of control pins.

$$\sum_{p=1}^{P_{max}} cp(x, y, p) = uc(x, y), (x, y) \in C \quad (28)$$

$$cp(x, y, p) \leq up(p), (x, y) \in C, 1 \leq p \leq P_{max} \quad (29)$$

$$up(p) \leq \sum_{(x, y) \in C} cp(x, y, p), 1 \leq p \leq P_{max} \quad (30)$$

Constraint (28) states that we should assign a pin to the cell which is used [10]. Constraint (29) and (30) state that if a cell (x, y) is controlled by a pin p , then p is used; otherwise p is un-used.

4. TWO-STAGE ILP-BASED ALGORITHM

Although the basic ILP formulations can optimally solve the droplet routing problem for PDMFBs, it is still limited in handling the dramatically growing complexity in practical bioassays. In this section, we propose a two-stage ILP-based droplet routing algorithm of global routing followed by incremental ILP-based routing for PDMFBs. We first overview our two-stage routing algorithm, and then detail each phase of our algorithm in the following subsections.

4.1 Routing Algorithm Overview

The essential intuition behind our algorithm is to reduce the complexity of the solution space in the basic ILP formulations by using a two-stage technique of global routing followed by incremental ILP-based routing.

The global routing stage first constructs the global routing tracks by analyzing the preferred moving direction of each droplet to guide the A* maze searching. Since droplets are recommended to route along the global routing tracks orderly, it can reduce the number of used cells and routing complexity. By performing A* maze routing for all droplets in global routing, the solution space for each droplet is reduced significantly from whole 2D plane to a global routing path.

In net criticality calculation, we determine the criticality of each droplet. A droplet is said to be critical if it is difficult to route it, due to the severe interferences with other droplets. This criticality information will be used in the incremental ILP-based routing stage.

Instead of considering all droplets at the same time, we propose an incremental ILP (IILP) approach to solve the routing problem in several manageable iterations to reduce the number of variables and constraints of the ILP formulations significantly. In each iteration, we select an un-routed droplet with the highest criticality, then route it with the previous routed droplets by solving the ILP formulations incrementally. Since searching a feasible solution is much faster than searching the optimal solution in a given ILP formulation, to further reduce the runtime, we propose a *deterministic integer linear programming* (DILP) formulation that casts the original routing optimization problem into a decision problem. The DILP will determine whether a feasible solution exists within given routing resources. To search a feasible solution in a decision problem efficiently, we perform a binary solution search method that searches it in logarithmic time. If this droplet cannot be routed, we will increase routing resources to improve the routability. Finally, iterations terminate until all droplets are routed.

4.2 Global Routing

The goal of global routing is to schedule the initial droplet routing paths to reduce the complexity of the solution space in the ILP formulations from whole 2D plane to global routing paths. With the increased design complexities, any naïve routing path may violate the timing and fluidic constraints easily. Furthermore, if droplets route disorderly, a large number of cells and independent control pins will be used. Hence, the reliability and fault tolerance for bioassays will be significantly degraded. To overcome

TABLE II: NOTATIONS USED IN OUR TWO-STAGE ILP FORMULATION.

G_i	set of used cells in global routing path for droplet d_i
G'_i	set of used cells by the previous routed droplet d_i
N	netlist among all subproblems
T_{max}^i	maximum available completion time that can be used for routing droplet d_i
P_{max}^i	maximum available number of control pins that can be used for routing droplet d_i
T_l^i	lower bound of T_{max}^i
T_u^i	upper bound of T_{max}^i
P_l^i	lower bound of P_{max}^i
P_u^i	upper bound of P_{max}^i
M_i	routing resources for droplet d_i to route with the previous routed droplets
IS	increasing scalar of routing resources
BB_i	set of available cells in bounding box of droplet d_i
E_b	set of blockage cells
E_{s_i}	set of available cells in the 3×3 area center by source s_i
E_{t_i}	set of available cells in the 5×5 area center by target t_i

these drawbacks, we construct the global routing tracks with the preferred moving direction to derive an initial routing path on these tracks for each droplet. Due to the fluidic constraints, it is desirable to maintain a minimum space when droplets move on the microfluidic array. Therefore, the initial global routing tracks are constructed on non-adjacent rows and columns. Then we determine the preferred moving direction of these tracks by analyzing the preferred moving direction of each net. We define $pmdl_i(x, y)$, $pmdr_i(x, y)$, $pmdu_i(x, y)$, and $pmdd_i(x, y)$ to represent the cell (x, y) with the left, right, up, and down preferred moving directions, respectively, within the bounding box of net n_i . Note that we use the real bounding box computed by the maze routing algorithm. For each cell (x, y) in the bounding box of net n_i , there are two preferred moving directions which are determined by the coordinates of source and sink. Therefore, the preferred moving direction of global routing tracks can be defined as follows:

- For tracks on rows (tr_j):
If

$$\sum_{(x,y) \in tr_j} \sum_{n_i \in N} pmdr_i(x, y) \geq \sum_{(x,y) \in tr_j} \sum_{n_i \in N} pmdl_i(x, y)$$

, the preferred moving direction is right; otherwise it is left.

- For tracks on columns (tc_j):
If

$$\sum_{(x,y) \in tc_j} \sum_{n_i \in N} pmdu_i(x, y) \geq \sum_{(x,y) \in tc_j} \sum_{n_i \in N} pmdd_i(x, y)$$

, the preferred moving direction is up; otherwise it is down.

After that, we model the routing path of droplet d_i as $P_{d_i} = \{v_1, v_2, \dots, v_n\}$ where each node v_i represents the cell used in microfluidic array, then apply A* maze searching to find a min-cost routing path for each droplet. Note that v_1 is the location of source and v_n is the location of sink. If droplet moves along the preferred moving direction from v_i to v_{i+1} , we assign the routing cost c_1 ; otherwise, we assign a higher routing cost c_2 for penalty. In this paper, we set c_1 and c_2 to be 1 and 3, respectively.

4.3 Net Criticality Calculation

A key issue in the droplet routing problem is the determination of the droplet routing order. A droplet d_i is said to be critical if d_i has fewer possible solutions (routing paths and schedules) due to the severe interferences with other droplets or blockage cells. We use $crit(d_i)$ to denote the criticality of droplet d_i and $crit(d_i)$ is defined as follows:

$$crit(d_i) = \frac{(|E_b^i| + |E_s^i|) - |E_t^i|}{|BB_i|} \quad (31)$$

where

$$E_b^i = \{c|c \in E_b \cap BB_i\}$$

$$E_s^i = \{c|c \in E_{s_j} \cap BB_i, \forall d_j \in D/d_i\}$$

$$E_t^i = \{c|c \in E_{t_j} \cap BB_i, \forall d_j \in D/d_i\}$$

The intuition behind the net criticality can be described as follows. Due to the blockage constraints and fluidic property, the cell of blockages and source cells inside BB_i will have detrimental effects on the routability of droplet d_i . On the contrary, since the target cells will become blockages after they are routed, a droplet with many target cells inside its bounding box has more routing solutions before these target cells are routed. As shown in the equation (31), the larger the $crit(d_i)$ is, the more critical the d_i is. This is because 1) as the numerator increases, the droplet d_i suffers from more interferences with other nets or blockages and 2) since the cells in BB_i are possible used frequently for routing, as the denominator decreases, there are fewer routing solutions for d_i .

4.4 Incremental ILP-Based Routing

After the global routing stage, the solution space is reduced significantly from whole 2D plane to global routing paths. To further reduce the solution space that directly considers all droplets at the same time, the incremental ILP-based routing routes an un-routed droplet with the previous routed droplets incrementally. Thus, for an un-routed droplet d_i and a previous routed droplet d_j , we reformulate the ILP constraints by replacing the whole 2D available cell set C with the cell set G_i which is used in its global routing path and the cell set G'_j which is used by the previous routed paths, respectively.

To further reduce the runtime, we cast the original optimization problem into a decision problem by solving the DILP formulations. In each iteration, we select an un-routed droplet with the highest criticality, then route it with the previous routed droplets by solving the DILP formulations incrementally. To search a feasible solution within minimal routing resources efficiently, we perform a binary solution search method that searches the feasibility in logarithmic time.

Although the above proposed method can solve the droplet routing problem in a reasonable runtime by global routing followed by incremental routing. However, as the increased design complexity of DMFBs, if the routing paths are restricted to the global routing paths, the freedom of droplets is also restricted, which may cause routability problem. Therefore, if we cannot route an un-routed droplet d_i with the previous routed droplets in the cell set G_i of global routing path, we increase the cell set G_i by one bounding box and reroute it. Finally, iteration terminates until all droplets are routed.

4.4.1 DILP Formulation

By global routing and incremental routing scheme, the solution space is reduced significantly. However, the ILP formulation is still limited in handling the dramatically growing complexity in current and future PDMFBs. To further reduce the runtime, we propose a DILP formulation that casts the original routing optimization problem into a decision problem. Thus, we redefine the objective function as follows:

$$\text{Minimize} : 1 \quad (32)$$

Instead of directly searching the original objective function within the fixed maximum available set T_{max} and P_{max} , the DILP determines a feasible solution only within the minimal routing resources. Therefore, for an un-routed droplet d_i with the previous routed droplets, the routing resources in our DILP formulation are T_{max}^i and P_{max}^i . In the constraints formulation, we replace the ILP sets of maximum completion time T_{max} with T_{max}^i , and the maximum available control pins P_{max} with P_{max}^i , respectively. Thus, the DILP try to minimize these two routing resources and determine if there exists a feasible routing solution within them.

4.4.2 Solution Search of DILP

The key issue in the DILP formulations is to minimize the routing resources that we can successfully route an un-routed

droplet d_i with the previous routed droplets. A naive approach is to exhaustively search all the permutation of routing resources among the range of $[0, T_{max}]$ and $[0, P_{max}]$. This method is time-consuming due to the time complexity is $O(T_{max} \times P_{max})$. Furthermore, it is hard to directly handle the two objective functions T_{max}^i and P_{max}^i efficiently. To remedy these deficiencies, we use a linear combination of these two objective functions to be one single objective function M_i and define the increasing scalar to characterize the growth rate of routing resources. Thus, the routing resources M_i can be defined as follows:

$$M_i = (T_l^i + \sigma_1 \cdot IS) + (P_l^i + \sigma_2 \cdot IS) \quad (33)$$

where IS is the growth rate of routing resources M_i and both σ_1 and σ_2 are user specified constants. As the experimental setting, we set σ_1 and σ_2 to be 1 and 0.5, respectively. Based on the definition, we have the following lemma.

Lemma 1: Given two increasing scalars IS_1 and IS_2 where $IS_1 < IS_2$. If droplet d_i can be routed with IS_1 , then droplet d_i can be also routed with IS_2 .

This lemma follows true since the increasing scalar IS of routing resources increases monotonically. If we have found a feasible routing resources that can route an un-routed droplet d_i , increasing the routing resources only increases the solution space of our DILP formulations. The lemma shows the *continuous* relationship of increasing scalar and feasibility of DILP formulations. This feature of increasing scalar shows the capability of solving in logarithmic time by performing a binary search method. Therefore, to avoid the runtime overhead caused by the exhaustive permutations, we propose a binary solution search method to optimally search the minimum increasing scalar, denoted by IS^* , for routing resources to route the un-routed droplet d_i successfully. Algorithm 1 shows our binary solution search method for IS^* .

Algorithm 1: Binary Search for IS^*

```

1 begin
2   //Set the lower bound and upper bound of routing resources
3   ;
4    $T_l^i \leftarrow \max\{G_i, T_{max}^j\}$  ;
5    $T_u^i \leftarrow T_{max}$  ;
6    $P_l^i \leftarrow \max\{0, P_{max}^j\}$  ;
7    $P_u^i \leftarrow P_{max}$  ;
8   //Set the lower bound and upper bound of  $IS$  ;
9    $IS_l \leftarrow 0$  ;
10   $IS_u \leftarrow \max\{(T_{max} - T_l^i)/\sigma_1, (P_{max} - P_l^i)/\sigma_2\}$  ;
11  while  $IS_l < IS_u$  do
12     $IS_m \leftarrow (IS_l + IS_u)/2$  ;
13    Set the corresponding routing resources  $M_i$  with  $IS_m$  ;
14    if  $d_i$  can be routed then
15       $IS_u = IS_m$  ;
16    else
17       $IS_l = IS_m + 1$  ;
18    end
19  endw
20 return  $IS_u$  ;
21 end

```

When routing an un-routed droplet d_i , we first set the lower and upper bound of routing resources as shown in line 2-6. Since we route each un-routed droplet with the previous routed droplet incrementally, and the cell set in DILP formulations is based on the global routing paths, we define the lower bound T_l^i of T_{max}^i to be the maximum value between the cells in global routing path G_i and T_{max}^j found by the previous routed droplet d_j . We define the lower bound of P_{max}^i to be the maximum value between zero and P_{max}^j since there may exist a subproblem without any droplet to be routed in PDMFBs. The upper bound of T_{max}^i and P_{max}^i is equal to the original constraints of T_{max}^i and P_{max}^i . The goal of our searching algorithm is to search the IS^* of M_i . By determining the lower and upper bound of routing resources, we can find the lower and upper bound of IS for M_i as shown in line 8-9. Then we perform the binary solution search method to find IS^* in line 10-18. For each searching iteration, the mean value

$IS_m = (IS_l + IS_u)/2$ is used to formulate the DILP. Finally, the searching iterations terminate until we find IS^* .

By the binary solution searching algorithm, the complexity of iterations is reduced to $O(\log(IS_u - IS_l))$. Compared with the exhaustively searching, the proposed algorithm reduces the runtime significantly.

5. EXPERIMENTAL RESULTS

Our two-stage ILP-based droplet routing algorithm was implemented in the C++ language and ran on a 2GHz 64-bit Linux machine with 8GB memory, and GLPK [1] was used as our ILP solver. We evaluated all routing algorithms on the two practical bioassays used in the previous work [11]: the in-vitro diagnostics and the colorimetric protein assay. Table III shows the statistics of each benchmark. To show the effectiveness and the robustness of our algorithm, we conducted three experiments for the number of control pins, the number of used cells, and the latest arrival time among the direct-addressing ([11], [4]), the broadcast-addressing scheme ([11] + [10] and [4] + [10]), and ours ([11] + IILP, [4] + IILP, and two-stage ILP) in Table IV. For fair comparison, we compare the maximum and average values among all subproblems. We also conduct an experiment on integrating our IILP routing approach with those direct-addressing-based droplet routing algorithms to demonstrate the effectiveness and efficiency of our IILP formulations in Table V.

For the first experiment, we compared the number of control pins among different schemes. Compared with the direct-addressing scheme ([11], [4]), the respective maximum and average number of control pins among all subproblems are (4.53 \times , 3.82 \times) and (4.44 \times , 4.03 \times) of our algorithm. Compared with the broadcast-addressing scheme ([11] + [10], [4] + [10]), the respective maximum and average number of control pins among all subproblems are (1.74 \times , 1.90 \times) and (1.78 \times , 2.06 \times) of our algorithm. To demonstrate the effectiveness of our IILP formulations, compared with the integrated scheme ([11] + IILP, [4] + IILP), the respective maximum and average number of control pins among all subproblems are (1.32 \times , 1.73 \times) and (1.54 \times , 1.83 \times) of our algorithm.

In the second experiment, we compared the number of used cells among different schemes. For the direct-addressing scheme ([11], [4]), the number of used cells among all subproblems are (1.02 \times , 1.07 \times) of our algorithm. Since the broadcast-addressing scheme is directly applied to the direct-addressing-based routing result, the number of used cells are the same with the direct-addressing scheme. Compared with the integrated scheme ([11] + IILP, [4] + IILP), the number of used cells among all subproblems are (1.00 \times , 1.02 \times) of our algorithm.

In the third experiment, we compared the latest arrival time among different schemes. Compared with the direct-addressing scheme ([11], [4]), the respective maximum and average latest arrival time among all subproblems are (1.01 \times , 1.05 \times) and (1.04 \times , 1.14 \times) of our algorithm. Since the broadcast-addressing scheme is directly applied to the direct-addressing-based routing result, the statistics of the latest arrival time are the same with the direct-addressing scheme. To demonstrate the effectiveness of our IILP formulations, compared with the integrated scheme ([11] + IILP, [4] + IILP), the average latest arrival time among all subproblems are (1.03 \times , 1.08 \times) of our algorithm.

Table V shows the runtime comparison among the basic ILP, direct-addressing + IILP, and our two-stage ILP algorithm. For the basic ILP, the problem instance is whole 2D plane and it solves all the droplets simultaneously. For our two-stage ILP, the problem instance is reduced to global routing paths and the droplets are routed in incremental manner that reduce the solution space significantly. The results show that the basic ILP needs at least five days to solve all 2D planes of one benchmark, which is not feasible for this problem; in contrast, our two-stage ILP algorithm needs at most 30.13 second due to the significantly smaller solution space. To demonstrate the effectiveness of our IILP formulations, compared with the integrated scheme ([11] + IILP, [4] + IILP), our algorithm reduced the runtime by about (34%, 55%).

Based on the evaluation of four experiments, our two-stage

TABLE IV: COMPARISONS FOR NUMBER OF PINS, NUMBER OF USED CELLS, AND LATEST ARRIVAL TIME AMONG THE DIRECT-ADDRESSING, BROADCAST-ADDRESSING, AND OURS

Benchmark	Direct Addressing				Broadcast Addressing				Two-Stage ILP					
	[11]		[4]		[11]+[10]		[4]+[10]		[11]+ILP		[4]+ILP		Ours	
	P _{max}	P _{avg}	P _{max}	P _{avg}	P _{max}	P _{avg}	P _{max}	P _{avg}	P _{max}	P _{avg}	P _{max}	P _{avg}	P _{max}	P _{avg}
vitro_1	45	21.55	50	23.45	21	9.48	22	10.11	15	9.11	18	9.49	13	4.51
vitro_2	50	15.73	42	16.40	21	8.95	24	10.64	17	8.03	17	9.21	12	5.01
protein_1	67	25.28	75	26.38	18	9.52	18	10.55	14	8.54	15	9.25	12	5.43
protein_2	54	12.03	46	12.35	23	8.73	21	8.55	17	7.72	23	7.38	11	4.43
	4.53	3.82	4.44	4.03	1.74	1.90	1.78	2.06	1.32	1.73	1.54	1.83	1	1

Benchmark	Direct Addressing		Broadcast Addressing		Two-Stage ILP		
	[11]	[4]	[11]+[10]	[4]+[10]	[11]+ILP	[4]+ILP	Ours
	U.C.	U.C.	U.C.	U.C.	U.C.	U.C.	U.C.
vitro_1	237	258	237	258	231	243	231
vitro_2	236	246	236	246	231	229	229
protein_1	1618	1688	1618	1688	1597	1627	1582
protein_2	939	963	939	963	927	943	930
	1.02	1.07	1.02	1.07	1.00	1.02	1

Benchmark	Direct Addressing				Broadcast Addressing				Two-Stage ILP					
	[11]		[4]		[11]+[10]		[4]+[10]		[11]+ILP		[4]+ILP		Ours	
	Max. T _i	Avg. T _i	Max. T _i	Avg. T _i	Max. T _i	Avg. T _i	Max. T _i	Avg. T _i	Max. T _i	Avg. T _i	Max. T _i	Avg. T _i	Max. T _i	Avg. T _i
vitro_1	20	13.00	19	14.30	20	13.00	19	14.30	19	12.47	19	13.55	18	12.41
vitro_2	17	11.33	20	12.00	17	11.33	20	12.00	17	11.01	17	11.48	18	10.46
protein_1	20	16.31	20	16.55	20	16.31	20	16.55	20	16.08	20	15.44	20	15.42
protein_2	20	10.51	20	12.19	20	10.51	20	12.19	20	10.33	20	11.52	20	10.22
	1.01	1.05	1.04	1.14	1.01	1.05	1.04	1.14	1.00	1.03	1.00	1.08	1	1

■ P_{max}: maximum number of control pins among all subproblems. ■ Avg. T_i: average latest arrival time among all subproblems.
 ■ U.C.: total number of unit cells used for routing. ■ Max. T_i: maximum latest arrival time among all subproblems.
 ■ P_{avg}: average number of control pins among all subproblems.

ILP-based droplet routing algorithm achieves the best result of the number of control pins, the number of used cells, and the latest arrival time over the existing algorithms within reasonable CPU times. The experimental results demonstrate that our algorithm is very effective for droplet routing on PDMFBs.

TABLE III: STATISTICS OF THE ROUTING BENCHMARKS

Benchmark	Size	#Sub	T _{max}	#Nets	#D _{max}
vitro_1	16 X 16	11	20	28	5
vitro_2	14 X 14	15	20	35	6
protein_1	21 X 21	64	20	181	6
protein_2	13 X 13	78	20	178	6

■ Size: size of microfluidic array. ■ #Sub: number of subproblems. ■ T_{max}: timing constraint.
 ■ #Nets: total number of nets. ■ #D_{max}: maximum number of droplets among subproblems.

TABLE V: COMPARISONS FOR RUNTIME AMONG BASIC ILP, DIRECT-ADDRESSING + ILP, AND OURS

Benchmark	Basic ILP	[11]+ILP	[4]+ILP	Ours
	CPU (min)	CPU (sec)	CPU (sec)	CPU (sec)
vitro_1	> 7200	14.33	15.31	10.11
vitro_2	> 7200	16.49	18.38	8.32
protein_1	> 7200	28.43	34.51	30.13
protein_2	> 7200	22.16	28.33	21.38
	N.C.	1.34	1.55	1

■ N.C.: Non comparable

6. CONCLUSION

In this paper, we proposed the *first* droplet routing algorithm that considers the droplet routing and the broadcast-addressing scheme *simultaneously* for PDMFBs. We first presented a basic integer linear programming (ILP) formulation to optimally solve the droplet routing problem with simultaneously minimizing the number of control pins, the number of used cells, and the latest arrival time. Due to its complexity, we also proposed a two-stage technique of global routing followed by incremental ILP-based routing. To further reduce the runtime, we presented a *deterministic* ILP formulation that casts the original routing optimization problem into a decision problem, and then solves it by a binary solution search method that searches in logarithmic

time. Extensive experiments demonstrate that in terms of the number of the control pins, the number of the used cells and the latest arrival time, we acquire much better achievement than all the state-of-the-art algorithms in any aspect.

7. REFERENCES

- [1] <http://www.gnu.org/software/glpk/>.
- [2] <http://www.siliconbiosystems.com/>.
- [3] K. F. Böhringer, "Modeling and controlling parallel tasks in droplet based microfluidic systems," *IEEE Trans. on CAD*, vol. 25, no. 2, pp. 334–344, Feb. 2006.
- [4] M. Cho and D. Z. Pan, "A high-performance droplet routing algorithm for digital microfluidic biochips," *IEEE Trans. on CAD*, vol. 27, no. 10, pp. 1714–1724, Oct. 2008.
- [5] E. J. Griffith, S. Akella, and M. K. Goldberg, "Performance characterization of a reconfigurable planar-array digital microfluidic system," *IEEE Trans. on CAD*, vol. 25, no. 2, pp. 345–357, Feb. 2006.
- [6] J. Peng and S. Akella, "Coordinating multiple robots with kinodynamic constraints along specified paths," *Int. J. Rob. Res.*, vol. 24, no. 4, pp. 295–310, Apr. 2005.
- [7] F. Su, W. Hwang, and K. Chakrabarty, "Droplet routing in the synthesis of digital microfluidic biochips," *Proc. IEEE/ACM DATE*, pp. 1–6, Mar. 2006.
- [8] T. Xu and K. Chakrabarty, "Droplet-trace-based array partitioning and a pin assignment algorithm for the automated design of digital microfluidic biochips," *Proc. IEEE/ACM CODES+ISSS*, pp. 112–117, 2006.
- [9] T. Xu and K. Chakrabarty, "Integrated droplet routing in the synthesis of microfluidic biochips," *Proc. IEEE/ACM DAC*, pp. 948–953, Jun. 2007.
- [10] T. Xu and K. Chakrabarty, "Broadcast electrode-addressing for pin-constrained multi-functional digital microfluidic biochips," *Proc. IEEE/ACM DAC*, pp. 173–178, Jun. 2008.
- [11] P.-H. Yuh, C.-L. Yang, and Y.-W. Chang, "BioRoute: A network-flow based routing algorithm for digital microfluidic biochips," *Proc. IEEE/ACM ICCAD*, pp. 752–757, Nov. 2007.