Voltage-Aware Chip-Level Design for Reliability-Driven Pin-Constrained EWOD Chips

Sheng-Han Yeh, Jia-Wen Chang, Tsung-Wei Huang, Tsung-Yi Ho
Department of Computer Science and Information Engineering
National Cheng Kung University, Tainan, Taiwan
{hardyyeh, jwchang, electron}@eda.csie.ncku.edu.tw
tyho@csie.ncku.edu.tw

ABSTRACT

Electrowetting-on-dielectric (EWOD) chips have become the most promising technology to realize pin-constrained digital microfluidic biochips (PDMFBs). In the design flow of EWOD chips, reliability is a critical challenge as it directly affects execution of bioassays. The major factor to degrade chip reliability is the trapped charge problem, which is induced by excessive applied voltage. Nevertheless, to comply with the pin constraint for PDMFBs, signal merging is inevitably involved, and thereby incurring trapped charges due to unawareness of applied voltage. Except for the trapped charge problem, wire routing to accomplish electrical connections increases the design complexity of pin-constrained EWOD chips. Unfortunately, no existing works tackle the problems of excessive applied voltage and wire routing, and thus the resultant chip will have more probabilities to fail during execution or can not be realized because of wire routing problem. In this paper, we present a network-flow based algorithm for reliability-driven pin-constrained EWOD chips with the consideration of voltage issue. Our algorithm not only minimizes the reliability problem induced by signal merging but also provides a comprehensive routing solution for EWOD chip-level designs. The experimental results demonstrate the effectiveness of proposed algorithm on real-life chips.

1. INTRODUCTION

As the microfluidic technology advances, digital microfluidic biochips (DMFBs) are emerging for biotechnology applications because of its high throughput, high sensitivity, and low cost compared to conventional laboratory procedures. By precisely controlling the movements of nanoliter droplets that contain samples and reagents, various applications such as DNA sequencing, immunoassays, environmental monitoring, and clinical diagnosis have been successfully realized on DMFBs [8, 13].

To date, different methods have been proposed and used to manipulate droplets. The most promising technology is electrowetting-on-dielectric (EWOD) chip. As the schematic view in Fig. 1(a), a general EWOD chip is composed of a two-dimensional (2D) electrode array, peripheral de-

vices (e.g., optical detector, dispensing port, etc.), and surrounding electrical pads. Sample carriers (i.e., droplets) on the array are controlled by underlying electrodes. By assigning time-varying voltage to actuate electrodes, droplets can be moved toward the actuated electrode due to the electrowetting phenomenon [12].
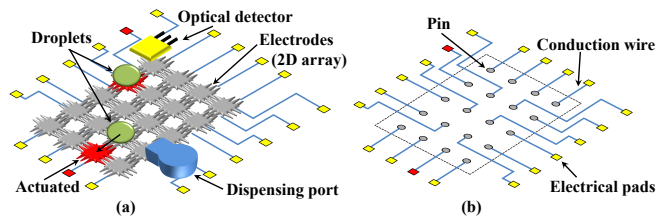

Figure 1: (a) The schematic view for EWOD Chips. (b) Underlying electrical connections.

In order to drive electrodes correctly, electrodes should be addressed with control pins to receive control signals, which is also referred to as *electrode addressing*. Early electrode addressing scheme relies on direct addressing which addresses each electrode with an independent control pin. However, as the chip size increases, it is necessary to limit the number of control pins since these pins are controlled by external controller with a limited number of signal ports. This triggers the demands of pin-constrained DMFBs (PDMFBs). To solve this problem, a widely used approach, *broadcast addressing*, are proposed [15]. Broadcast addressing utilizes the concept of pin sharing to assign single control pin to multiple electrodes without affecting the execution of assay.

Effective as broadcast addressing is, two major problems are incurred if the signals are shared arbitrarily: 1)Trapped charge problem, 2)Wire routing issue. Trapped charge is a phenomenon that the charge gets trapped in the dielectric insulating layer of the chip, causing the reduction of electrowetting force. This phenomenon results in erroneous executing result and even causes permanent dielectric breakdown [4]. To deal with trapped charge problem, chip designers usually apply minimum required voltages to electrodes according to different operations. For example, dispensing may require $60 - 80$ volts to actuate electrodes to precisely dispense droplet with correct size from the dispensing ports, transporting may require at least $10-20$ volts to move droplets around the 2D electrode array [5]. However, when broadcast addressing scheme is involved, trapped charge problem become crucial and more severe. Take Fig. 2 for example. In (a), 3 control pins are used and each electrode is applied with its required voltage. The result of broadcast addressing in (b) to reduce the required

pin count may assign Pin 1 to electrode e_1 and e_3 with $80V$ actuating voltage (suppose e_3 requires at least $80V$ to be actuated). It can be observed that electrode e_1 is actuated with higher voltage value than the result of direct addressing in (a). And in consequence the charges get trapped due to excessive applied voltage.
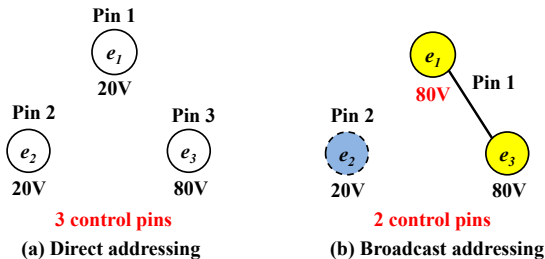

Figure 2: Broadcast addressing may assign inappropriate voltage values to electrodes.

It has been reported that the trapped charge problem has a crucial influence on the chip reliability [7]. Reliability is an important issue for PDMFBs as many biochips are expected to be used for reliability-driven applications such as patient health monitoring, clinical diagnosis, etc [2]. These chips require reliable chips for instant decision making. Moreover, for pin-constrained EWOD biochips, once the electrodes malfunction, the entire assay will fail because the underlying wiring can not be rerouted to re-synthesis the assay after fabrication.

In addition to the concern of trapped charge, another problem, wire routing problem, further increases the design complexity. Wire routing problem is the issue to route the conduction wires from the top side electrode array, through the underlying substrate, to the surrounding electrical pads so as to transmit the control signal of each control pin. The connections may be infeasible if the electrodes are not addressed carefully. It may needs extra processes or even additional routing layer(s), which are undesirable for low-cost EWOD chip fabrication.

Accordingly, to design a feasible and reliable chips, there is a urgent demand to develop a dedicated algorithm to minimize the trapped charge impact as well as complete the EWOD chip design automatically.

1.1 Related works

In the literature, there is only one existing work that concerns reliability issue in electrode addressing. The work in [10] evaluates reliability of the addressing result by the maximum actuation unit, $AU_{max} = max\{AU_k\}$, where AU_k represents the required actuation unit in the actuation sequence of electrode e_k. It minimizes AU_{max} by transforming it into a deterministic problem and then solves the sub-problem by a matching-based algorithm. This work is effective in terms of maximum actuation time. However, it has been reported that voltage has a significant influence on the reliability. Trapped charge problem, the major factor of reliability, will not occur if we apply the voltage appropriately even with high actuation time. Moreover, electrode addressing without considering voltage issue will require additional voltage switching devices to meet the voltage concern or consume extra power due to equally high voltage assignment. Therefore, it is more practical to evaluate the reliability by voltage factor (see Section 2.2).

On the other hand, focusing on the electrode addressing technique is insufficient especially for EWOD chip designs since it can not guarantee that the addressing result will

produce feasible routing solution. It may need extra processes to obtain a successful routing solution or even additional routing layer, thereby increasing the cost for chip fabrication. Although there are some works focus on the routing stage for EWOD chips by conducting electrode addressing and wire routing simultaneously [3, 9], they aim to maximize the routability of the chip and do not take the reliability issue into the consideration. Hence, the resultant chip may have more likelihood to fail or burnout due to the unawareness of reliability.

1.2 Our contributions

In this paper, we propose an algorithm for reliability-driven pin-constrained EWOD chips by considering applied voltage to electrodes. We formulate the model to identify the voltage issue that affects chip reliability. Our contributions can be summarized as follows:

- We carry out a comprehensive study on the phenomenon of trapped charge and minimize its impact. Unlike the previous work in [10] which evaluates trapped charge by actuation time of electrodes, we focus on the of factor of voltage value that is more practical for reliability issue.

- Our algorithm not only minimizes the reliability problem but also provides a comprehensive routing solution for chip manufacturing. By taking the reliability as well as routability issue into account, the corresponding wire routing result can be quickly obtained without addition of extra routing layer.

- We propose a network-flow based algorithm that addresses electrodes and route wires simultaneously. Accompanied by a incremental search technique, the addressing and routing problem can be efficiently and effectively solved.

Experimental results shows the effectiveness of our algorithm. By performing our algorithm on a set of real life applications, our algorithm can obtain the best addressing results to reduce inappropriate voltage assignment. And the wire routing results show that our algorithm have a better routability by completing all test cases.

The remainder of this paper is organized as follows. Section 2 describes the related preliminaries including broadcast addressing, trapped charge problem, and the routing model for EWOD chips. Then Section 3 formulates the problem in this paper. Section 4 details our algorithm to solve the reliability-driven chip-level design problem. Finally, Section 5 and 6 demonstrate the experimental result and give the conclusions respectively.

2. PRELIMINARIES

In this section, we introduce the broadcast addressing method and the trapped charge problem which is a critical issue for chip reliability.

2.1 Pin-Constrained Broadcast Electrode Addressing

As mentioned in introduction section, DMFBs are controlled by external micro-controller. Due to the limited I/O ports of the controller, it is inevitable to restrict number of used control pins for large scale DMFBs. Therefore, it justifies the necessity of pin-constrained DMFB (PDMFB) designs. In PDMFBs, a design specification P_{max} is given, indicating the maximum number of control pins this chip

can use. To comply such demand, many approaches to reduce the number of control pin are proposed. The prevailed approach is broadcast addressing, which utilizes the feature of *electrode actuation sequence*.

Electrode actuation sequence stores the control information of each electrode. Each bit in the sequence represents the actuation status of the electrode in different time step (i.e., i-th bit represents the status of the corresponding electrode in i-th time step). And the status can be actuated "1", grounded "0", or don't care "X". The don't care term reveals that the input signal can be either "1" or "0", which has no impact on the droplet controlling.

By observing the actuation sequence, it can be found that multiple actuation sequences can share an identical sequence by replacing don't care terms with "1" or "0". In other words, multiple electrodes can be controlled by the same control pin (same actuation sequence). Based on the this addressing scheme, the required pin count can be reduced to the desired pin count P_{max}. This scheme is referred to as *broadcast addressing*.
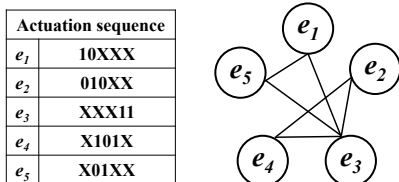
Actuation sequence	
e_1	10XXX
e_2	010XX
e_3	XXX11
e_4	X101X
e_5	X01XX

Figure 3: Actuation sequences of electrodes (e_1 - e_5) and the corresponding compatibility graph.

To realize broadcast addressing, a *compatibility graph* is used. Each node in the graph represents an electrode and the edge indicates the adjacent two nodes are compatible (i.e., the two electrodes can share the same control pin without affecting the controlling of droplets), illustrated in Fig. 3. To reduce the pin count and solve the electrode addressing problem, multiple cliques are selected, where each clique represents an independent control pin. It has been shown that finding a minimum pin count (cliques partition) is equivalent to the classical NP-hard problem [15]. Heuristic algorithm is used to reduce the pin count as few as possible [9, 15].
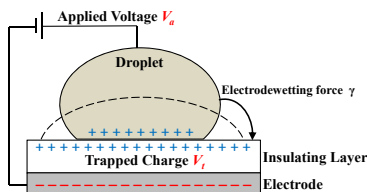
2.2 Trapped Charge Problem

Figure 4: Illustration of trapped charge.

When a voltage is applied to an electrode, an electrowetting force γ is generated and charge forms at the liquid/solid interface. The interfacial charging changes the free energy at the liquid/solid interface, causing the droplet to spread as it seeks a new equilibrium state (dashed line in Fig. 4). This is the essence of the EWOD chips which brings various operations to be performed to DMFBs. However, there is a possibility that charge becomes trapped in or on the insulating layer when the interaction of the ions with the solid is stronger than with the liquid (see Fig. 4). The accumulation of trapped charge weakens the electrowetting

force γ, thereby affecting the electrowetting performance. Researches have pointed out that trapped charge voltage has a strong relationship with the applied voltage to electrodes [1, 7].
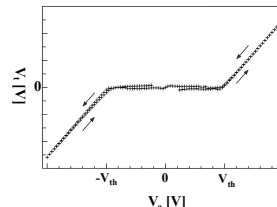

Figure 5: The trapped charge voltage, V_t, of a function of applied voltage V_a.

A threshold voltage V_{th} depended on the different devices and materials are noticed to influence trapped charge. When the voltage V is applied, it can be divided into two scenarios: 1) $V \leq V_{th}$, 2) $V > V_{th}$. For $V \leq V_{th}$, the voltage of trapped charge equals zero, which means no charge trapped and that electrowetting force $\gamma \propto V^2$. But when $V > V_{th}$, charges are trapped and accumulated in or on the insulating layer over time and are seen as a reduction in the electrowetting force [1]. Fig. 5 (from [1]) shows that the voltage due to trapped charge, V_t, is a function of applied potential (voltage). Accordingly, the following relationship can be derived when $V > V_{th}$:

$$\gamma \propto (V - V_t)^2 \tag{1}$$

$$V_t \propto V \tag{2}$$

Once the force γ is weaker than the desired force to manipulate droplets, unexpected behavior such as stuck droplet and unbalanced splitting occur. Such errors may lead to the needs to re-execute the entire assay.

A way to enhance γ is to increase the applied voltage V_a. Nevertheless, according to equation (1) and (2), V_a and V_t are mutually reinforcing. Increasing V_a will accelerate the accumulation of trapped charge, then a higher voltage value to manipulate droplets is required. With higher and higher applied voltage, it may cause completely dielectric breakdown due to excessive field intensity. Consequently, increasing the actuating voltage value is unsuitable. Another alternative to resolve such problem is to apply appropriate voltage value according to different electrodes and operations.

2.3 Driving Voltage Issue

A solution to relieve the trapped charge problem is to actuate electrodes with appropriate voltage. In practicality of DMFBs, each operation prefers a specific driving voltage value. The value of required voltage is depended on the chip material (e.g, the material and the thickness of insulating layer, substrates, environment, etc), operations, and the property of reagents. For example, Pollack et al. use $15 - 40$V to transport droplets where the bottom plate is soaked in silicon oil and $50 - 60$V without oil (i.e., in air) [12]; Paik et al. use $140 - 220$V to manipulate droplets on coplanar system [11]; Lin et al. use 11.4V and 7.2V to dispense and transport a 300pl droplet respectively on a 95um electrode with a 20um SU8 gasket [16]. If the applied voltage is lower than operation's preferred voltage, operation may be conducted incorrectly, leading the whole assay to fail. Therefore, voltage applied to the electrode

that performs specific operations must be higher than its preferred voltage. On the other hand, as aforementioned in Section 2.2, charge density increases as the applied voltage enhances. If the applied voltage is too high, charges will trap and accumulate more and more as the voltage increases.

It has also been reported that excessive voltage value may instead cause unexpected operations. Researches have indicated that droplets may be split during transportation if a too high voltage are applied [12]. As a result, it is more desirable to apply the electrode with voltage value which is approximative to the preferred operation voltage while satisfying design constraints (e.g., pin-constraint).
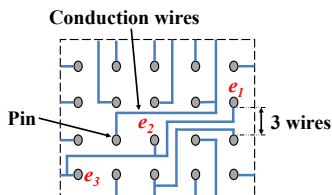
2.4 EWOD Chip Routing

Figure 6: **Routing model for EWOD chips.**

After electrodes are addressed with control pins, conduction wires must be appropriately routed to establish the correspondence between control pins (i.e., electrodes with the same control pin must be connected with conduction wires) and the signal ports. As shown in Fig. 6, electrodes e_1, e_2, e_3 share the same control pin. So there should be conduction wires to establish the signal transmission. Additionally, in order to receive the input signal, the corresponding wires have to connect with outside signal ports. This criteria is similar to the typical escape routing which route each individual terminal pin to the boundary of the chips [14]. However, control pin sharing in pin-constrained EWOD chips makes it difficult because EWOD chip routing exists many multiple terminal nets instead of two-terminal net in typical escape routing problem. Moreover, freedom of electrode addressing (i.e., electrodes have the choices to share the control pin with different electrodes) and the reliability issue further increase the design complexity for EWOD chip routing. Therefore, a specialized router for reliability-aware EWOD chips is desired.

In this paper, the inexpensively fabrication technique, printed circuit board (PCB), is used [6]. Fig.6 demonstrates the routing model which is based on a uniform grid routing structure. Similar to the state-of-the-art works for EWOD chip routing [3, 9], we route the wires by horizontal and vertical fashions with routing angle of 90 degree. And available routing layer of single is used. Besides, we allow at most three wires passing through adjacent pins.

3. PROBLEM FORMULATION

Regarding above discussions, the reliability-driven routing problem for pin-constrained EWOD chips can be formulated as follows:

Input: *A set of electrodes $E = \{e_1, e_2, ..., e_n\}$ with actuation sequence and the preferred voltage value V_{pre_i} for electrode $e_i \in E$, a threshold voltage V_{th} that incurs trapped charge, a specified value of P_{max} indicating the maximum pin count supported by external controller, and chip specification.*

Constraints:

- Voltage constraint: *Voltage value assigned to each electrode should be higher or equal to the electrode's preferred voltage.*
- Broadcast-addressing constraint: *A set of electrodes can be addressed to a single control pin if and only if their actuation sequences are mutually compatible.*
- Pin constraint: *The used control pins to control all electrodes must be less or equal to pin constraint P_{max}.*
- Routing constraint: *Satisfying the design rules of the chip.*

Since the trapped charge density will increase as the applied voltage enhances, to precisely evaluate the effect of trapped charge and then obtain a more reliable addressing result, we would like to measure the voltage difference, V_{d_i}, between assigned voltage $V_{e_i}^*$ and preferred voltage V_{pre_i} among all electrodes e_i, $\forall e_i \in E$. Note that for the assigned voltage $V_{e_i}^*$ less than V_{th}, trapped charge problem will not be incurred. Therefore, V_{d_i} can be expressed as following:

$$V_{d_i} = \begin{cases} V_{e_i}^* - max(V_{th}, V_{pre_i}), & \text{if } V_{e_i}^* \geq V_{th} \\ 0, & \text{if } V_{e_i}^* < V_{th} \end{cases}$$

The global effect of trapped charge can be reduced by minimizing $V_{d_{max}}$, which is the maximum of V_{d_i} (see Eq. 3).

$$V_{d_{max}} = max\{V_{d_i} | \forall e_i \in E\} \quad (3)$$

In summary, the objective of our problem is mathematically formulated as following.

Objective: *Correctly deriving an electrode-addressing result that each electrode is assigned with a voltage $V_{e_i}^*$ and corresponding feasible routing solution based on the addressing result. The major goal is to relieve the trapped charge problem by minimizing $V_{d_{max}}$.*

4. ALGORITHM

We present our network-flow based routing algorithm in this section. We first discuss the voltage-constrained compatibility graph used in our routing algorithm. Then, two main parts of our algorithm, (1) the *incremental search technique* and (2) *simultaneous broadcast addressing and routing*, are respectively detailed.

4.1 Voltage-Constrained Compatibility Graph

As mentioned in Section 2, compatibility graph (G_c) can be constructed by identifying the compatibility of control signals between electrodes. To take the reliability issue into account, we first introduce a *voltage-constrained compatibility graph* G_{vcc}, which is a restricted graph derived from G_c. That is, we add a *voltage difference constraint* into compatibility graph to form voltage-constrained compatibility graph.

$$V_{d_i} \leq V_{bound} \quad (\forall e_i \in E_e) \quad (4)$$

In voltage difference constraint, a bound voltage difference V_{bound} is given in order to force the value of V_{d_i} among all electrodes e_i to satisfy Eq (4). Two electrodes e_a and e_b in G_{vcc} are compatible if and only if they are compatible in G_c and:

1. $V_{max_{ab}}$ - $\max(V_{th}, V_{d_a}) \leq V_{bound}$

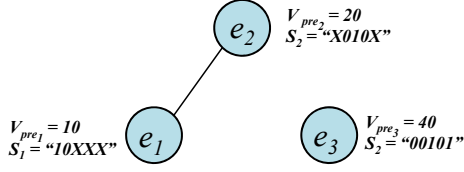2. $V_{max_{ab}}$ - $\max(V_{th}, V_{d_b}) \leq V_{bound}$

Figure 7: Voltage-constrained compatibility graph with bound voltage = 15, $V_{th} = 15$.

Where $V_{max_{ab}}$ is the larger preferred voltage between e_a and e_b (i.e, $V_{max_{ab}} = \max(V_{pre_a}, V_{pre_b})$). For example in Fig. 7, e_2 is compatible to e_1 and e_3. Suppose V_{bound} is 15 and V_{th} is 15. With voltage difference constraint, e_2 is **incompatible** with e_3 because if we address e_2 and e_3 together, the value of V_{d_2} would become 20 ($V_{e_2}^*$ - $\max\{V_{pre_2}, V_{th}\}$ = 40 - 20) which violates the voltage difference constraint. So e_2 and e_3 are incompatible in voltage-constrained compatibility graph since they violate the voltage difference constraint, $V_{bound} = 15$.

With the voltage-constrained compatibility graph, it can be guaranteed that $V_{d_{max}} \leq V_{bound}$ in the resultant addressing result. Thus, we can focus on the pin-count and routability issues in our algorithm.

4.2 Incremental Search Method

Algorithm 1: Incremental Search

Input : A 2D pin array, an electrode set E_e, pin constraint P_{max}, preferred voltage V_{pre_i} for each electrode

```
/* V_s:  the search voltage            */
/* V_th:  threshold voltage            */
/* V_UpperBound :  the upper bound voltage of
   incremental search                  */
```
1 **begin**
2    $V_{max} \leftarrow$ maximum preferred voltage among all electrodes;
3    $V_{UpperBound} \leftarrow V_{max} - V_{th}$;
4    $V_s \leftarrow 0$;
5   **while** $V_s \leq V_{UpperBound}$ **do**
6   build G_{vcc} under V_s;
7   simultaneously broadcast addressing and routing based in G_{vcc};
8   **if** *unsuccessful route* **then**
9        $V_s \leftarrow V_s + 1$;
10   **else**
11   **return** feasible routing result;
12   **end**
13   **end**
14 **end**

The overview of incremental search method is presented in Algorithm 1. In order to solve the problem of trapped charge, the major goal of incremental search method is to obtain a feasible addressing and routing solution while minimizing $V_{d_{max}}$. Thus, to find the minimum $V_{d_{max}}$, we conduct several iterations with the concept of incremental search. Each iteration of incremental search is associated with a voltage V_s, indicating the *bound voltage* in G_{vcc} that we discussed in Section 4.1. That is, we construct G_{vcc}

under specific V_s in each iteration of the while loop (line 6). Then, simultaneously addressing and routing scheme is conducted based on the constructed G_{vcc} (line 7). Obviously, it can be found that the maximum possible value of $V_{d_{max}}$ is bound within the range from 0 to $V_{max} - V_{th}$. Therefore, we search the minimum $V_{d_{max}}$ by setting V_s as 0, and terminate when $V_s > (V_{max} - V_{th})$ or a feasible solution is found. We have following advantages to use incremental search method:

- By utilizing the property of G_{vcc}, our incremental search method can maintain the minimum value of $V_{d_{max}}$ once the corresponding feasible addressing and routing solution with the given V_s is found.

- The complexity to find a feasible electrode addressing result is dramatically reduced. We can construct different G_{vcc} under different V_s. The compatibility graph constructed with higher V_s is complexer than that with lower V_s. Since we start our incremental search from $V_{bound} = 0$, the beginning G_{vcc} is the lowest-complexity graph.

4.3 Simultaneously Broadcast Addressing And Routing

Based on G_{vcc} constructed in the previous step, we conduct addressing and routing in this subsection. There are two major goals in this step:

1. All the electrodes are addressed with control pins while the number of pins can satisfy P_{max}.
2. All the electrodes assigned to the same pins forming a number of nets should be successfully routed to obtain a feasible routing solution.

Algorithm 2: Simultaneously Addressing and Routing

Input : An G_{vcc} derived from incremental search, pin constraint P_{max}

```
/* E_u:  The set of unaddressed electrodes   */
```
1 **begin**
2   indentify $E_{initial}$ from G_{vcc};
3   assign each eletrode in $E_{initial}$ with individual pin;
4   **while** *the number of unaddressed electrode > 0* **do**
5   identify an unaddressed electrode set E_e' from E_u;
6   construct MCMF network based on existing pins and E_e';
7   obtain addressing result by solving MCMF network;
8   **for** *each net in addressing result* **do**
9   route the net;
10   **if** *failure route* **then**
11   reroute or drop this addressing result;
12   **else**
13   update pin assignment;
14
15   **end**
16   **end**
17   escape route for all nets;
18   **return** pin assignment result and routing solution;
19 **end**

The main idea in our addressing and routing algorithm is to divide the original problem into a set of manageable sub-problems corresponding to a *pin-electrode merging*. The

actions of pin-electrode merging including both **electrode addressing** and **wire routing**. In each iteration of pin-electrode merging, the entire electrode set is decomposed into two subsets, an unaddressed electrode set and an addressed electrode. All the addressed electrode was in a existing pin. We then try to address the electrode in unaddressed electrode set with existing pins. It ends when all electrodes are addressed.

The overview of addressing and routing algorithm is presented in Algorithm 2. In the beginning of our algorithm, we would identify a set of electrodes (called $E_{initial}$) while it is mutually incompatible between any two electrodes in G_{vcc} (line 2). Mutually incompatible set is a set that any two electrodes in this set are not compatible. The problem to find the mutually incompatible set can be mapped to the typical NP-hard problem of finding maximum independent set and a number of heuristics can be used. Since any two electrodes in $E_{initial}$ cannot share the same pin, the minimum number of pins we need is the size of $E_{initial}$. In other words, it needs at least $|E_{initial}|$ control pins to correctly actuate all electrodes because any electrode in $E_{initial}$ is not compatible with others in $E_{initial}$. Then, the electrodes in $E_{initial}$ are addressed with individual control pins. For each iteration of pin-electrode merging (line 4 - 16), we first identify an electrode group E'_e from G_{vcc} with mutually incompatible control signals (line 5). Then, electrode addressing and routing are well-formulated into a minimum-cost maximum-flow (MCMF) network to utilize the available existing control pins (line 6). Since the solution of the MCMF network represents a solution of electrode addressing, by tracing the MCMF network we can obtain an addressing result (line 7). For each net in the result of addressing, we start to route them by breadth-first-search (BFS) based algorithm. We also provide some reroute technique that we will detailed in section 4.3.2. Electrode addressing and routing terminate when all the electrodes are addressed or the pin constraint is satisfied. Finally, we conduct escape route [14] for all nets to obtain a feasible routing solution from each net to chip boundary (line 17). Escape routing is done by modelling original problem into maximum-flow problem. After we find a feasible routing solution, a comprehensive addressing and routing solution with $V_{d_{max}} = V_s$ is obtained.

Our addressing and routing algorithm offers two major advantages as follows.

1. To reduce the design complexity, we do not directly solve the original problem but focus on each manageably-sized subproblem.
2. By formulating the electrode addressing and routing into a flow network, the pin-count can be minimally determined.

4.3.1 Min-Cost Maximum-Flow Formulation

Our major challenge of pin-electrode merging is the potential interference between addressed and unaddressed electrodes. In this subsection, we first introduce our network-flow model to merge unaddressed electrodes into existing control pins. Then, we detail the edge cost and function of *HPWL extension* used in the network-flow model.

To formulate the problem of pin-electrode merging, we construct a minimum-cost maximum-flow (MCMF) graph $G_{mcmf} = (V_{mcmf}; E_{mcmf})$ and propose two formulation rules. The first rule describes the establishment of V_{mcmf}, and the second rule describes the formulation of E_{mcmf}. The following rules respectively describes the formulation of V_{mcmf} and E_{mcmf}:

MCMF-Rule #1: Formulation of V_{mcmf}

1. For each unaddressed electrode $e_i \in E'_e$, create a node v_{e_i}.
2. For each using control pin $p_i \in P_s$, create a node v_{p_i}.
3. Create a source node S, and a sink node T.

MCMF-Rule #2: Formulation of E_{mcmf}

1. For each node v_{p_i}, create a directed edge $S \to v_{p_i}$ with *one* unit capacity and *zero* cost per unit flow.
2. For each node v_{e_i}, create a directed edge $v_{e_i} \to T$ with *one* unit capacity and *zero* cost per unit flow.
3. For each node pair (v_{e_i}, v_{p_j}), where $e_i \in E'_e$ and $p_j \in P_s$, create a directed edge $v_{p_j} \to v_{e_i}$ with *one* unit capacity and *HPWL-extension*(p_j, e_i) cost per unit flow.
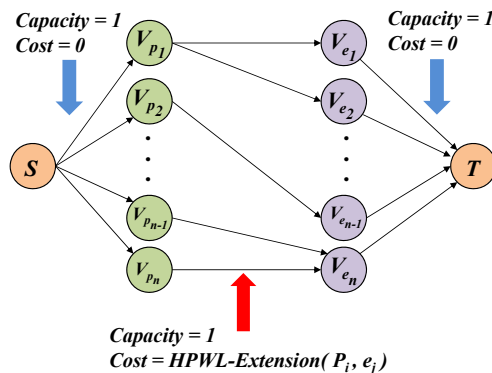

Figure 8: Minimum-cost Maximum-flow construction for our algorithm

Note that *HPWL-extension*(p_j, e_i) in above rules is used to estimate the routing cost and will be discussed later. Fig. 8 shows an example of MCMF network construction. By solving this MCMF network, we can obtain an addressing result for used pins corresponding to unaddressed electrodes. Each edge with $flow = 1$ between V_{p_j} and V_{e_i} in the result of flow network indicates that e_i is addressed to existing pin p_j.

The most important objective in our MCMF model is to maximize *the number of unaddressed electrodes* which can be addressed with used pins P_i while maximize the routability of corresponding wire routing. To take the routability into account, we map the estimated routing wirelength into flow cost to minimize wirelength and increase routability for routing. The wirelength estimation function HPWL-extension(P_i, e_j) can be done by calculating the half-perimeter wirelength (HPWL) of the bounding box formed by unaddressed electrode e_j to the nearest electrode in existing pin P_i. For example in Fig. 9, e_1 and e_2 are sharing the same pin, P_1. To address the e_3 with P_1, we need to route from e_3 to the wire of P_1. So the HPWL extension to address e_3 with P_1 is 6, which is covered by green region (green dashed line).

In the flow cost model with routing consideration, we can obtain a solution of MCMF that the total cost of HPWL extension are minimized. Thus, a set of pin-to-electrode nets are routed between (1) electrodes in used pin p_j and (2) unaddressed electrode e_i.
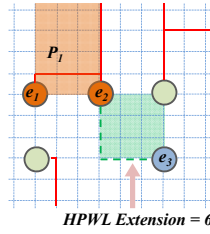
Figure 9: Calculation of HPWL extension between an existing control pin and unaddressed electrode

4.3.2 Addressing and Routing

All the pair of existing control pin and unaddressed electrode in the solution of MCMF network would be routed in the increasing order of the cost of HPWL extension. To deal with the failure in routing. We provide the reroute technique for failed route as follows. A failed route occurs when (1) no routing path exists, (2) some electrodes which are blocked by other nets and can not be routed to the chip boundary. In the condition of (1), it means that the electrode is not appropriate to be assigned to it's corresponding pin due to non-existing routing path. So we drop this addressing solution of pin-electrode pair and leave the electrode unaddressed. And in the second condition, for simply expression, we illustrate the situation by Fig. 10. In Fig. 10(a), an existing routing wire connects with e_3 and now we are going to route the nets between e_1 and e_4. By using breadth-first-search (BFS) based algorithm for routing, we can find a routing path from e_1 to e_4 with minimum wirelength (see (b)). In (b), we can observe that a failed route occur since e_2 is blocked by wires of other nets. The key idea of reroute technique is to increase the cost of previous routing path and then route the net again. As shown in (c), the dashed line represent the new routing path from e_1 to e_4. It is successfully routed when no electrode is blocked by other wires (see (d)). In our observation, failed route occurs mostly due to the second situation. The reroute technique can be a key role in our routing algorithm to increase the routability.
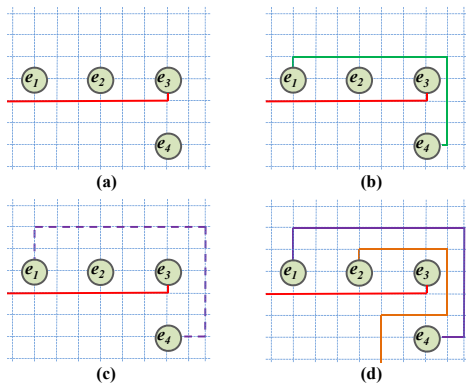

Figure 10: Example of failed route and reroute technique.

5. EXPERIMENTAL RESULTS

We implement the proposed algorithm with C++ language on a 2.63-GHz 64-bit Linux machine with 32GB memory. A set of PDMFB applications are used to evaluate our algorithm. These applications include amino-acid synthesis, multiplexed assay, PCR amplification, multifunctional chip, and DNA sample preparation [15, 10].

TABLE I: STATISTICS OF ALL CASES

Chip	Size	#E	P_{max}	Avg. V_{pre}
amino	6 X 8	20	16	32.8
multiplex	15 x 15	59	32	17.5
PCR	15 X 15	62	32	21.9
multifunctional	15 X 15	91	64	19.7
DNA preparation	13 X 21	77	32	20.8

The threshold voltage V_{th} to incur trapped charge is set as 22 [1]. Table I lists the specifications of all test cases.

We compare the electrode addressing and the wire routing results with [10]. Since [10] does not provide the wire routing result, we implement the wire routing according to the electrode addressing result obtained from [10] and assign voltage to each control pin with the maximum preferred voltage value among corresponding electrodes. In the routing scheme, we first iteratively route the wires to connect electrodes together based on maze routing algorithm and then apply escape routing to establish the paths between outside electrical pads and inside control pins. When a failure occurs, rip-up and reroute schemes are adopted to find the feasible routing solution. And the history-based technique is also used to prevent the maze routing go through the previous failed paths.

Table II and III demonstrate the overall comparison results. First, we compare the electrode addressing result. Our algorithm shows the better reliability by achieving averaged $V_{d_{max}}$ of 3, while it is 55.8 in [10]. This indicates that most electrodes in our algorithm are assigned with appropriate voltages or below the threshold voltage. To further analysis the result, we plot the distributions of voltage difference. Fig. 11 shows the voltage difference (V_d) distribution among all electrodes for both methods. It can be found that in the result of [10], there is a large number of electrodes with the V_d of $40-60$ for all cases. For example, there are 11 electrodes (18% of totally 59 electrodes) with the voltage difference between 51 to 60 in multiplex assay. It represents that many electrodes are assigned with the voltages that are much higher than their preferred voltage due to unawareness of voltage issue. Moreover, since maximum voltage difference is minimized in the proposed algorithm, the number of electrodes that are in the risk of trapped charge problem is also reduced. Column 3 and 7 in Table II show the number of electrodes whose voltage difference (V_{d_i}) are not zero (denoted as $\#E_{v_d \neq 0}$). And column 4 and 8 demonstrate the proportion of $\#E_{v_d \neq 0}$ to #E. As the table shows, the proposed algorithm can achieve totally 78.9% fewer number of electrodes that are applied with excessive voltage. These results justify the practicality and effectiveness of our algorithm.

Secondly, we compare the wire routing results between these two methods. "WL" denotes the wire length measured by the number of routing grids. "#Fail" denotes the number of failed electrodes (i.e., unable to find a valid addressing and routing manner). "CPU" denotes the execution time measured in seconds. As shown in Table II, we can complete all of the test chips while [10] can only complete one. The reason is that [10] does not take the routability into account such that the wire routing result can not be guaranteed. Moreover, both methods are given with the same pin constraint, P_{max}. Our algorithm terminates when the pin constraint is satisfied to avoid unnecessary routing problems incurred from further pin merging. But the method in [10] minimize the used pin count instead. Therefore, even with fewer used pin count, the wire routing result can not be guaranteed. Finally, in terms of

359

TABLE II: ELECTRODE ADDRESSING COMPARISON BETWEEN [10] AND OURS

Chip	[10]				Ours			
	$V_{d_{max}}$	$\#E_{v_d\neq0}$	$\#E_{v_d\neq0}/\#E(\%)$	#Pin	$V_{d_{max}}$	$\#E_{v_d\neq0}$	$\#E_{v_d\neq0}/\#E(\%)$	#Pin
amino	57	5	25%	13	2	1	5%	16
multiplex	56	24	40%	7	0	0	0%	32
PCR	54	40	64.5%	12	5	12	19.3%	32
multifunctional	55	40	43.9%	12	0	0	0%	64
DNA preparation	57	19	24.6%	22	8	14	18.1%	32
Total		128				27		

TABLE III: WIRE ROUTING COMPARISON BETWEEN [10] AND OURS

Chip	[10] + Wire Routing				Ours			
	#Pin	WL	#Fail	CPU	#Pin	WL	#Fail	CPU
amino	13	206	0	0.0	16	180	0	0.0
multiplex	7	N/A	35	0.55	32	995	0	0.3
PCR	12	N/A	30	0.42	32	1217	0	1.5
multifunctional	12	N/A	39	0.76	64	1175	0	0.4
DNA preparation	22	N/A	28	0.89	32	1450	0	2.4
Total		132				0		
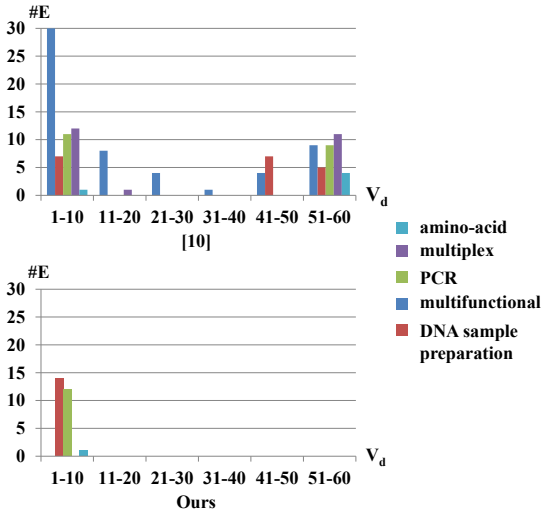
Figure 11: The voltage difference (V_d) distributions by using [10]'s approach and our approach for all test cases.

execution time, our algorithm can complete all test chips within few seconds. This result demonstrates our algorithm is very efficient to complete the reliability-driven chip-level designs.

6. CONCLUSIONS

We presented in this paper electrode addressing and wire routing algorithm for voltage-aware, reliability-driven pin-constrained EWOD chips. Our algorithm was based on progressive network-flow as well as incremental search technique to minimize the excessive applied voltage to electrodes. And the property of trapped charge problem were studied and addressed into our algorithm. The experimental results on a set of real-life applications demonstrated that the proposed approach was very effective and efficient.

7. ACKNOWLEDGEMENT

8. REFERENCES

[1] Shaun Berry, Jakub Kedzierski, and Behrouz Abedian "Irreversible Electrowetting on Thin Fluoropolymer Films," *ACS J. Langmuir*, no. 23, pp. 12429–12435, 2007.

[2] K. Chakrabarty, "Towards fault-tolerant digital microfluidic lab-on-chip: defects, fault modeling, testing, and reconfiguration," *Proc. IEEE ICBCS*, pp. 329–332, 2008.

[3] J.-W. Chang, T.-W. Huang, and T.-Y. Ho, "An ILP-based obstacle-avoiding routing algorithm for pin-constraint EWOD chips," *Proc. ASP-DAC*, pp. 67–72, 2012.

[4] A. I. Drygiannakis, A. G. Papathanasiou, and A. G. Boudouvis, "On the connection between dielectric breakdown strength, trapping of charge, and contact angle saturation in electrowetting," *ACS J. Langmuir*, no. 25, pp. 147–152, 2009.

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

[6] J. Gong and C. J. Kim, "Direct-referencing two-dimensional-array digital microfluidics using multilayer printed circuit board," *IEEE J. MEMS*, no. 2, pp. 257–264, 2008.

[7] H. J. J. Verheijen and M. W. J. Prins, "Reversible electrowetting and trapping of charge: model and experiments," *ACS J. Langmuir*, no. 15, pp. 6616–6620, 1999.

[8] T.-Y. Ho, J. Zeng, and K. Chakrabarty, "Digital microfluidic biochips: A vision for functional diversity and more than Moore," *IEEE/ACM ICCAD*, pp. 578–585, 2010.

[9] T.-W. Huang, S.-Y. Yeh, and T.-Y. Ho, "A network-flow based pin-count aware routing algorithm for broadcast electrode-addressing EWOD chips," *Proc. IEEE/ACM ICCAD*, pp. 425–431, 2010.

[10] T.-W. Huang, K. Chakrabarty, and T.-Y. Ho, "Reliability-oriented broadcast electrode-addressing for pin-constrained digital microfluidic biochips," *Proc. IEEE/ACM ICCAD*, pp. 448–455, 2011.

[11] P. Y. Paik, V. K. Pamula, M. G. Pollack and K. Chakrabarty, "Coplanar digital microfluidics using standard printed circuit board processes," Proc. MicroTAS, pp. 566–568, 2005.

[12] M. G. Pollack, A. D. Shenderov, and R. B. Fair, "Electrowetting-based actuation of droplets for integrated microfluidics," *Lab on chip*, pp. 96–101, 2002.

[13] F. Su, K. Chakrabarty, and R. B. Fair, "Microfluidics based biochips: Technology issues, implementation platforms, and design-automation challenges," *IEEE TCAD*, pp. 211–223, 2006.

[14] T. Yan and M. D. F. Wong, "A correct network flow model for escape routing," *Proc. ACM/IEEE DAC*, pp. 332–335, 2009.

[15] T. Xu and K. Chakrabarty, "Broadcast electrode-addressing for pin-constrained multi-functional digital microfluidic biochips," *Proc. IEEE/ACM DAC*, pp. 173–178, 2008.

[16] Y. Y. Lin, R. D. Evans, E. Welch, B. N. Hsu, A. C. Madison, R. B. Fair, "Low voltage electrowetting-on-dielectric platform using multi-layer insulators," Sensor. Actuat. B-Chem. 150, pp. 465–470, 2010.