

A Logic Integrated Optimal Pin-Count Design for Digital Microfluidic Biochips

Trung Anh Dinh¹
trungda@ngc.is.ritsumei.ac.jp
¹Ritsumeikan University, Japan

Shigeru Yamashita¹
ger@cs.ritsumei.ac.jp
²National Cheng Kung University, Taiwan

Tsung-Yi Ho²
tyho@csie.ncku.edu.tw

Abstract—Digital microfluidic biochips have become one of the most promising technologies for biomedical experiments. In modern microfluidic technology, reducing the number of independent control pins that reflects most of the fabrication cost, power consumption and reliability of a microfluidic system, is a key challenge for every digital microfluidic biochip design. However, all the previous chip designs sacrifice the optimality of the problem, and only limited reduction on the number of control pins is observed. Moreover, most existing designs cannot satisfy high-throughput demand for bioassays, and thus inapplicable in practical contexts. In this paper, we propose the *first optimal pin-count design scheme* for digital microfluidic biochips. By integrating a very simple combinational logic circuit into the original chip, the proposed scheme can provide high-throughput for bioassays with an *information-theoretic minimum* number of control pins. Furthermore, to cope with the rapid growth of the chip’s scale, we also propose a scalable and efficient heuristics. Experiments demonstrate that the proposed scheme can obtain much fewer number of control pins compared with the previous state-of-the-art works.

I. INTRODUCTION

Thanks to the recent advance of microfluidic technology, digital microfluidic biochips have been replacing conventional laboratory experiments in a number of biomedical applications including drug discovery, high-throughput DNA sequencing, and environmental toxicity monitoring. By manipulating discrete picoliter biochemical droplets, this kind of biochips offers a number of advantages over traditional procedures such as high sensitivity, high throughput, low power consumption, less human intervention, fast and precise execution [1].

A typical digital microfluidic biochip consists of a two dimensional control electrode array, peripheral devices such as dispensing ports, optical detectors, integrated logic and surrounding control pins as illustrated in Fig. 1. Due to electrowetting phenomenon, droplets can be moved by assigning time-varying voltage values to activate/deactivate the electrodes in an appropriate strategy [2]. To precisely control the movements of the droplets, electrodes are connected to control pins to realize input signals as shown in Fig. 1. Early chip designs based on a *direct-addressing* scheme, in which each electrode is independently assigned to a dedicated control pin [3]. Although this method maximizes the independence and flexibility of electrode control, as the scale of digital microfluidic biochips grows rapidly, the number of independent pins for large-scale biochips also increases significantly. Recently, a chip that contains more than 600,000 electrodes has been successfully demonstrated [4]. In reality, a digital microfluidic biochip that utilizes a large number of control pins suffers from several drawbacks. A large number of pins obviously leads to high

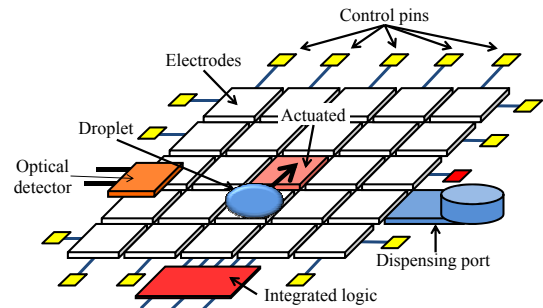


Fig. 1: Schematic view of a DMFB.

production cost, high power consumption, and low reliability, which are pivotal keys of most emerging technologies. Besides, the associated routing complexity (or even infeasible routing solutions) increases the number of PCB layers [3], which also contributes to the production cost, portability and disposability. Furthermore, in current digital microfluidic technology, the control pins are actuated by driving signals from an off-chip micro-controller. A large number of control pins may lead to significant transmission latency between the controller and the control pins. In such a case, electrodes cannot be actuated in a synchronous manner, and thus the whole experiment may be faulty and defected, which is obviously unacceptable in such experiments like drug discovery or point-of-care diagnosis.

Therefore, to satisfy the requirements of the biological marketplace, reducing the number of control pins in a digital microfluidic biochip has a crucial role in every biochip design scheme. In fact, in the last few years, the pin-count minimization challenge of digital microfluidic biochips has attracted a great interest of researchers [5]–[10].

A. Previous work

After direct-addressing scheme was introduced, some heuristic approaches have been proposed in [5], [6]. However, only limited reduction in the number of control pins is observed by using such methods.

A cross-reference driving scheme is discussed in [7]. Basically, in this scheme, electrodes in the same column are connected to the same control pin, and electrodes in the same row are also connected to the same control pin. Therefore, for a $m \times n$ electrode array, only $m + n$ control pins are enough to manipulate the droplets. However, due to electrode interference, this scheme does not allow simultaneous movements of more than two droplets, which is a critical drawback for high-throughput bioassays.

The works in [8]–[10] introduce different design flows to generate the pin-assignment results. Since they are different from the typical approach which starts from the droplet routes

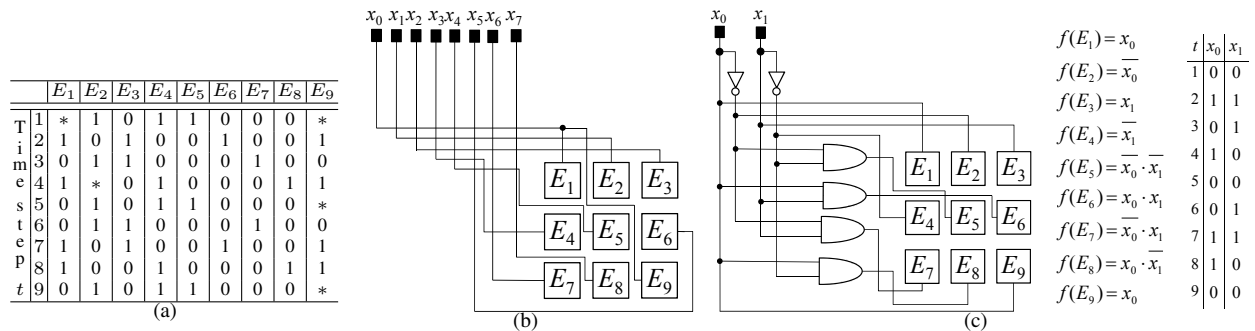


Fig. 2: An illustrative example: (a) electrode-actuation sequences, (b) applied the broadcast-addressing scheme, (c) motivational example of a new design scheme.

and then attempts to reduce the number of control pins [4]–[7], we do not consider them in the rest of this paper.

The most effective work, which can efficiently reduce the number of control pins but still guarantee the completion time of bioassays, is the so-called *broadcast-addressing* scheme [4]. By solving the minimum clique partition problem heuristically, electrodes that have “compatible” input signals are assigned to the same control pin. However, the optimality of the problem cannot be guaranteed and if the compatibility among input signals is low, this method still necessitates a large number of control pins. Moreover, since one pin may be shared among several electrodes, the routing solution of conduction wires inside the chip may become very complex or even infeasible.

To overcome the above drawbacks of the previous works, we propose a new design scheme for digital microfluidic biochips. By taking advantage of recent innovation of the integration between digital microfluidic biochips and semiconductor devices [11]–[14], we utilize an on-chip integrated combinational logic circuit to drive the signals from the control pins to the electrodes. Distinguished from all the previous schemes, the proposed one can achieve an information-theoretic optimal number of control pins to handle a biochip.

B. Semiconductor devices integrated digital microfluidics

Recently, along with the advance of digital microfluidic technology, the integration of digital microfluidics with active semiconductor devices also attracts much attention of researchers and biochip designers. At the earliest stage, the works in [11], [12] introduced CMOS compatible fabrication procedures for digital microfluidic systems which triggered the possibility of integrating semiconductor devices into a digital microfluidic system. After that, a number of applications has been developed based on such integration [13]–[15]. For example, in [13], in order to determine the presence and size of a droplet on an electrode, each electrode of a digital microfluidic biochip is assigned an individual sensing circuit to measure the electrical impedance of the droplet above it.

Recently, a concept of cyberphysical digital microfluidic biochips has been introduced in [14]. In such a chip, complex sensing systems whose objective is to send feedback of the biochemical application to the control software are integrated to the original biochips.

In this paper, exploiting this integration, we propose a logic integrated design scheme for digital microfluidic biochips. Compared to all the aforementioned applications that use complex integrated systems, in the proposed scheme, a much simpler combinational logic circuit, which is used to drive the

control signals to the electrodes, is integrated into the original biochip. By doing so, we can obtain the information-theoretic minimum number of control pins for a biochip.

C. Our contributions

The key contributions of the proposed method can be summarized as follows:

- In traditional design schemes, control signals from off-chip pins are directly driven to the on-chip electrodes. However, we discover a new way to alternate control signals before they are driven to the electrodes. In order to do so, we first construct an on-chip logic circuit. Signals from control pins become inputs of this circuit, and then output signals of this circuit are driven to the electrodes. To the best of our knowledge, it is the *first* time such method of driving control signals is introduced for digital microfluidic biochips.
- As similar to the direct-addressing and the broadcast-addressing schemes, the proposed one can provide high-throughput for bioassays. Moreover, we can obtain the *information-theoretic optimal* number of control pins for digital microfluidic biochips.
- Although experiments demonstrate that the optimal method works perfectly well for current practical benchmarks, to deal with the rapid grow of the chip’s scale, a scalable heuristic approach is also proposed. A number of large test cases are used to evaluate the proposed heuristic algorithm.

The rest of this paper is organized as follows. Section II discusses the background of this work. After that, the proposed new design scheme for digital microfluidic biochip is introduced in Section III. We demonstrate our experiments in Section IV and conclude the paper in the final section.

II. BACKGROUND

A. Electrode-actuation sequences

To execute a specific bioassay, information of droplet routing and scheduling must be stored in form of electrode-actuation sequences [4]. Each bit in a sequence, which can be denoted as “1” (activated), “0” (deactivated) or “*” (don’t care), represents the status of an electrode at a specific time step. Specifically, at a given time step, the status of an electrode, which is at high (low) voltage, is denoted as “1” (“0”). The symbol “*” indicates that the input signal can be “1” or “0” without any impact on the movements of droplets.

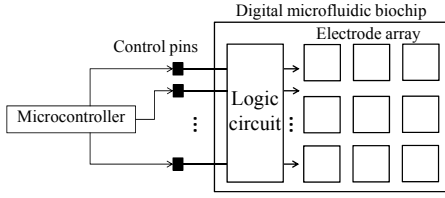


Fig. 3: Schematic view of the proposed design scheme.

An example of the electrode-actuation sequences is shown in Fig. 2 (a). For the sake of convenience, the i^{th} electrode is denoted by E_i ; the status of an electrode E_i at time step t is denoted by $S(E_i, t)$, e.g., in Fig. 2 (a), $S(E_3, 3) = "1"$. This example will be used to demonstrate a motivational example in the next subsection.

B. Motivational example

To correctly control the status of electrodes, electrodes must be assigned to control pins. In the broadcast electrode-addressing scheme, by carefully replacing "*" with "1" or "0", we can merge multiple compatible actuation sequences to a single control signal, and only one control pin is enough to handle such compatible sequences. In other words, electrodes that have compatible actuation sequences can be assigned to the same control pin.

In the example in Fig. 2 (a), the actuation sequences of E_9 is considered to be compatible to the one of E_1 by setting both $S(E_9, 5)$ and $S(E_9, 9)$ to "0", i.e., the two sequences of E_1 and E_9 are compatible. Moreover, in this example, there is no other pair of compatible sequences. Fig. 2 (b) illustrates the broadcast-addressing scheme's result of this example, while E_1 and E_9 are connected to the same control pin, i.e., x_0 , the remaining seven electrodes are assigned to a seven dedicated control pins, i.e., x_1, x_2, \dots, x_7 . Totally, we need eight control pins by the broadcast-addressing scheme.

However, if we consider an actuation sequence of an electrode as output signal of a logic function, and implement nine such logic functions for nine electrodes individually, we can decrease the number of required control pins to only two. Specifically, Fig. 2 (c) describes the logic functions corresponding to the electrodes with respect to only two control pins x_0 and x_1 , and the values of x_0, x_1 during time steps. As shown in Fig. 2 (c), each electrode E_i is assigned a logic function $f(E_i)$, e.g., $f(E_5) = \overline{x_0} \cdot \overline{x_1}$.

For an electrode E_i , the value of the function $f(E_i)$ at a given time step t is exactly the same as the status of this electrode at t . For example, the status of E_5 at time step $t = 5$ can be calculated by $f(E_5) = \overline{x_0} \cdot \overline{x_1} = 1 (= S(E_5, 5))$ since the values of x_0 and x_1 at $t = 5$ are both "0". In summary, all the actuation sequences can be realized by logic functions with respect to x_0 and x_1 ; two control pins are enough.

This example demonstrates that if we implement a logic circuit in order to drive the control signals to the electrodes of a digital microfluidic biochip, the number of control pins can be reduced significantly when compared to the broadcast-addressing method. In this example, 75% reduction on the number of control pins (2 compared to 8) is observed. This example motivates a new design scheme for digital microfluidic biochips as discussed in the next section.

III. PROPOSED DESIGN SCHEME

In this paper, we propose a new design scheme for digital microfluidic biochips, which integrates a combinational logic

circuit into the original chip. The schematic view of the proposed scheme is illustrated in Fig. 3. In the proposed scheme, signals from control pins become input signals of the combinational logic circuit, and the output signals of the circuit are now the control signals of the chip's electrodes. The main challenges of the proposed design scheme are how to construct the logic circuit and how to drive the signals to the electrodes correctly. In this section, we first formulate the problem formally. Then, we present an optimal and a heuristic algorithm to solve the problem. Finally, some features of the on-chip logic integration is discussed.

A. Problem formulation

Given: Set of N electrodes $SN = \{E_1, E_2, \dots, E_N\}$ of a given biochip and their actuation sequences.

Objective: For each electrode $E_i (i = 1, 2, \dots, N)$, construct an associated logic function $f(E_i)$, which can realize its actuation sequence, while minimizing the total number of control pins.

B. Optimal pin assignment

This subsection describes an optimal algorithm, which can construct a logic function for each electrode with an optimal number of control pins. The proposed algorithm consists of two main stages: (1) optimal partitioning of time steps; (2) logic circuit construction, which will be explained in the following.

1) *Optimal partitioning of time steps:* For a better explanation, for each time step t , we define a *status sequence* T_t , which is a sequence concatenating the status of all the electrodes from 1 to N at time step t . For example, in Fig. 2 (a), $T_1 = "*1011000*"$, $T_5 = "01011000*"$.

In this paper, we consider compatibility in terms of time steps, not for electrodes as the previous broadcast electrode-addressing approach does. More specifically, two time steps t_i and t_j are said to be *compatible* if by carefully replacing "*" in their corresponding status sequences (T_{t_i} and T_{t_j}) with "1" or "0", we can obtain the same sequence. For example, since setting the first "*" of T_1 to "0", we can achieve the same sequence to T_5 , time step 1 and time step 5 are compatible. We try to minimize the number of groups of compatible time steps by setting "*" to appropriate values. The problem can be easily transformed to the minimum clique-partition in graph theory [16] by the below procedure:

- Step 1. Construct the following undirected graph \mathcal{G} :
 - For each time step t , there has a corresponding vertex v_t in \mathcal{G} .
 - If two time steps, t_i and t_j , are compatible, there is an edge between the corresponding two vertices v_{t_i} and v_{t_j} in \mathcal{G} .
- Step 2. Solve the minimum clique-partition problem of \mathcal{G} .
- Step 3. Each clique of the result of Step 2 corresponds to a groups of compatible time steps.

Based on the actuation sequences in Fig. 2 (a), we can construct the graph \mathcal{G} as shown in Fig. 4. For example, since time step 1 and time step 5 are compatible, there is an edge between v_1 and v_5 in Fig. 4.

The implication of the above procedure can be explained

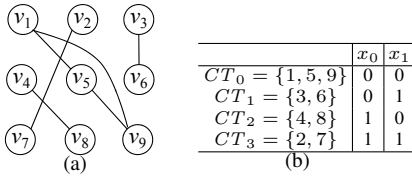


Fig. 4: Explanation of the example in Fig. 2: (a) graph \mathcal{G} of the time steps, (b) minimum number of groups of compatible time steps and input sequences.

as follows. There is an edge between any pair of two vertices in a clique. Therefore, a clique corresponds to one group of mutually compatible time steps. Also, since the number of cliques (in the result of Step 2) is minimum, the number of the groups of compatible time steps should be minimum. The minimum clique partition result of the graph \mathcal{G} in Fig. 4 (a) is $\{v_1, v_5, v_9\}, \{v_2, v_7\}, \{v_3, v_6\}, \{v_4, v_8\}$; and the number of cliques is 4. Our key idea is that when the number of cliques is K , $\lceil \log_2 K \rceil$ control pins are enough to control the electrodes as we will explain in the next subsection.

2) *Logic circuit construction:* It is left to answer a question how to design logic function for each electrode. In this subsection, we mention a key theorem to support our design scheme, and the proof of the theorem is an answer to the question. The logic circuit construction of the electrodes in Fig. 2 (a) will be given to exemplify the theorem.

Theorem 1: If the number of groups of compatible time steps is K , the number of control pins of a biochip can be decreased to $\lceil \log_2 K \rceil$.

Proof: We show an explicit construction below. Let K groups of compatible time steps be $CT_0, CT_1, \dots, CT_{K-1}$. In our example, $CT_0 = \{1, 5, 9\}$, $CT_1 = \{3, 6\}$, $CT_2 = \{4, 8\}$ and $CT_3 = \{2, 7\}$ as shown in Fig. 4 (b).

Firstly, we prepare k ($= \lceil \log_2 K \rceil$) logic variables representing k control pins: x_0, x_1, \dots, x_{k-1} such that the values of x_0, x_1, \dots, x_{k-1} at time step $t \in CT_m$ become the binary representation of the integer m ($m = 0, \dots, K-1$). In our example, since the number of groups of compatible time steps is 4, we need $k = 2$ logic variables x_0 and x_1 . At time step 3, $x_0, x_1 = 0, 1$ respectively since $3 \in CT_1$ and $1_{10} = 01_2$. In this way, the values of x_0 and x_1 during time steps can be determined as shown in Fig. 2 (c).

Next, for each electrode, we construct a logic function f such that the outputs of f during time steps corresponds to its actuation sequence. For example, we want to construct a logic function for an electrode E_i . Our key observation is that $S(E_i, t_l)$ are the same for all time steps t_l in a group of compatible time steps. Since if there exist t_1 and t_2 such that $S(E_i, t_1)$ and $S(E_i, t_2)$ are not the same, obviously t_1 and t_2 cannot be in the same group of compatible time steps. Let $CT_{i_1}, CT_{i_2}, \dots, CT_{i_{m_i}}$ be the groups of compatible time steps such that $S(E_i, t_l) = 1$ for all $t_l \in \cup_{l=1,2,\dots,m_i} CT_{i_l}$. Then the function corresponding to the actuation sequence for E_i can be given by the following formula: $f(E_i) = P_{i_1} + P_{i_2} + \dots + P_{i_{m_i}}$, where P_{i_l} is a logic product (min-term) with respect to x_0, x_1, \dots, x_{k-1} such that x_j is negated (i.e., \bar{x}_j is appeared in the product form) if the j -th bit (from the left) is “0” in the binary representation of an integer i_l . Otherwise (i.e., when the j -th bit (from the left) is “1” in the binary representation of an integer i_l), x_j is appeared in P_{i_l} . For example, when $k = 2$, $P_1 = \bar{x}_0 \cdot x_1$ since $1_{10} = 01_2$.

Consider the logic construction for electrode E_9 in our example, since the status of E_9 is “1” at all time steps in $CT_2 = \{4, 8\}$ and $CT_3 = \{2, 7\}$ (see Fig. 2 (a)), we can see that $S(E_9, t) = 1$ for all $t \in CT_2 \cup CT_3$. Therefore, the logic function for E_9 can be given as follows:

$$f(E_9) = P_2 + P_3 = x_0 \cdot \bar{x}_1 + x_0 \cdot x_1 = x_0$$

The justification of the above construction is as follows. First it can be seen that the logic product P_{i_l} becomes 1 at all time steps in CT_{i_l} . This is because P_{i_l} becomes 1 only when $x_j = 1$ (0) if the j -th bit (from the left) is 1 (0), respectively, in the binary representation of an integer i_l . This condition means that P_{i_l} becomes 1 when the inputs x_0, x_1, \dots, x_{k-1} are in the binary representation of an integer i_l , which happens at time steps in CT_{i_l} . Therefore, $f(E_i) = P_{i_1} + P_{i_2} + \dots + P_{i_{m_i}}$ becomes 1 at all time steps in $CT_{i_1}, CT_{i_2}, \dots, CT_{i_{m_i}}$. In other words, the values during time steps of $f(E_i)$ becomes exactly the same as the actuation sequence for E_i .

In conclusion, we can always make corresponding logic functions for all electrodes with respect to the logic variables, x_0, x_1, \dots, x_{k-1} . It is almost obvious that a different encoding from the index of G_i to the binary representation (x_0, x_1, \dots, x_{k-1}) leads to a different logic complexity. It may need similar techniques as state encoding for state machines to get simple logic functions. However, the discussion is out of the scope of this paper. ■

Using Theorem 1, we can construct logic functions for all electrodes in Fig. 2 (a) as illustrated in Fig. 2 (c). Moreover, based on this theorem, we can claim the following two corollaries:

Corollary 1: The number of control pins used in our proposed scheme is the *information-theoretic minimum* result.

Proof: K groups of compatible time steps in our problem can be considered as K different patterns in information theory, and we need at least $\lceil \log_2 K \rceil$ bits to encode K patterns. In our problem, $\lceil \log_2 K \rceil$ control pins is the smallest number that can encode K different patterns of time steps. Therefore, our design scheme can obtain the information-theoretic minimum number of control pins to handle a digital microfluidic biochip. ■

Corollary 2: k control pins can be used to actuate a set of electrodes that has at most $K = 2^k$ groups of compatible time steps.

Proof: This corollary can be deduced directly from Theorem 1. ■

While Corollary 1 proves the optimality of the proposed scheme, Corollary 2 is an important property, which will be applied in the next subsection.

C. Heuristic pin assignment

Although general minimum clique partition problem is known to be NP-hard, an exact branch-and-bound algorithm proposed in [17] works perfectly well for all practical benchmarks as will be illustrated in Section IV. However, as microfluidic technology has developed significantly, the scale of a digital microfluidic biochip is expected to increase rapidly in recent decade. Also, there may be a problem in our proposed design scheme such that the complexity of logic functions may increase if every function depends on a large number of input variables when the number of the control pins becomes large.

Electrode Partition Procedure

Require: SN is a given set of electrodes.

Require: Actuation sequences of all the electrodes.

Require: k is to specify the number of control pins for each new set of electrodes.

```

1:  $M \leftarrow 0$  {/*total number of new sets*/}
2: while  $|SN| > 0$  do
3:    $M \leftarrow M + 1$ 
4:    $SE_M \leftarrow E_i$  that is selected randomly from  $SN$ .
5:    $CurrentNum \leftarrow 2$ 
6:    $flag \leftarrow true$ 
7:   while  $flag == true$  do
8:      $MinIncrease \leftarrow \infty$ 
9:     for all  $E_i \in SN$  do
10:       $Increase = Try(E_i, SE_M)$  {/*try adding  $E_i$  to  $SE_M$ */}
11:      if  $Increase < MinIncrease$  then
12:         $MinIncrease = IncreaseNum$ 
13:         $E_{Best} = E_i$ 
14:      end if
15:    end for
16:    if  $CurrentNum + MinIncrease \leq 2^k$  then
17:       $SE_M \leftarrow SE_M \cup E_{Best}$ 
18:       $SN \leftarrow SN \setminus E_{Best}$ 
19:       $CurrentNum \leftarrow CurrentNum + MinIncrease$ 
20:    else
21:       $flag = false$ 
22:    end if
23:  end while
24: end while

```

Fig. 5: Pseudocode for the algorithm to obtain minimal number of control pins.

To deal with such challenges, in this paper, we also propose an efficient heuristic to solve the problem. The objective of the heuristic algorithm is to partition the electrodes of a given biochip into M sets of electrodes such that for each set, only less than a predefined value k control pins are enough to drive actuation signals to electrodes of this set. If we can do so, the logic function for an electrode always depends on less than k input variables, and only less than $k \times M$ control pins are enough to actuate all electrodes of the biochip. Moreover, it should be noted that according to the aforementioned Corollary 2, the number of compatible time steps in each of M electrode sets *must be smaller than or equal to* 2^k . In short, the problem can be formulated formally as follows:

Given: (1) A set of N electrodes of a given biochip: $SN = \{E_1, E_2, \dots, E_N\}$ and their actuation sequences; (2) a predefined value k .

Objective: Partition N electrodes into M smaller sets of electrodes denoted by SE_1, SE_2, \dots, SE_M such that for each set SE_i ($i = 1, \dots, M$), less than k control pins are enough to generate actuation sequences for all electrodes in this set while minimizing the number of groups, i.e., M .

Fig. 5 depicts a pseudocode of the proposed heuristic algorithm. Basically, for a currently constructed set of electrodes SE_M , we keep the number of the groups of compatible time steps under 2^k while adding a new electrodes to SE_M . The current number of groups of compatible time steps of SE_M is kept as $CurrentNum$ in Fig. 5. Note that for only one electrode E_i , the number of groups of compatible time steps is 2, so the initial value of $CurrentNum$ is set to 2 at line 5. (We do not consider actuation sequence with only 1 (or 0).)

Our greedy strategy tries to find the best electrode, E_{Best} , such that adding E_{Best} to SE_M increases the minimum number of groups of compatible time steps of SE_M . This number is hold on $Increase$, and calculated by $Try(E_i, SE_M)$ at line 10. $Try(E_i, SE_M)$ is a function that tries adding E_i

to SE_M and calculates the increasing number of the groups of compatible time steps in SE_M . Therefore, between lines 8 and 15, we can find the best electrode to be added to SE_M . At line 16, we check whether this addition violates the limit number of compatible time steps, i.e., 2^k , or not. If it does not violate the limitation, we move the electrode (E_{Best}) from SN to SE_M and update $CurrentNum$ (line 17 to 19). Otherwise, we give up increasing the size of SE_M , and construct a new set, i.e., SE_{M+1} . To do so, we set the flag to be false at line 22 to go outside the while loop from line 7.

For example, consider the set of electrodes in Fig. 2 (a) and set k to 2, at the first iteration of the while loop at line 7, we have $SE_1 = \{E_1\}$, $CurrentNum = 2$ since there are two groups of compatible time steps in SE_1 now. Adding E_2 to SE_1 does not increase $CurrentNum$, and thus E_2 is currently the “ E_{Best} ” electrode for SE_1 . Therefore, at the second iteration of the while loop, we add E_2 to SE_1 ($SE_1 = \{E_1, E_2\}$, $CurrentNum = 2$). At the third iteration of the while loop, we have to choose an electrode from the remaining seven ones (E_3, \dots, E_9) to be added to SE_1 . By calling $Try(E_i, SE_1)$, we can check that adding E_9 to SE_1 does not increase the number of the groups of compatible time steps, while adding one of the others to SE_1 makes that number increased by 1 (by adding either of E_5, E_6, E_7 or E_8) or 2 (by adding E_3 or E_4). Therefore, we add E_9 to SE_1 , and have $SE_1 = \{E_1, E_2, E_9\}$, $CurrentNum = 2$. The procedure is repeated until all of the nine electrodes are assigned to new sets. Actually, in this example, all electrodes are assigned to SE_1 , and thus $M = 1$; a total of $k \times M = 2 \times 1 = 2$ control pins are enough to handle all electrodes. The heuristic procedure can obtain the same result as the optimal solution does.

The time complexity in the worst case of the heuristic procedure is $O(N^2)$, where N is the total number of electrodes. As k represents the number of input variables of each logic function, a large number of k will lead to the increase of the complexity of the functions. Therefore, we experiment the heuristic for $k = 2, 3, 4$ which imply simple and practical logic functions in general.

D. On-chip logic integration

As discussed in Section I-B, there exist several applications which integrate semiconductor circuits into digital microfluidic biochips [13]–[15]. Compared to the integrated systems used in the previous works, e.g., sensing circuits, the proposed scheme utilizes a much simpler combinational logic circuit. It is also reported in [11], [12] that the current fabrication process of digital microfluidic biochips is compatible to that of CMOS technology and that the typical droplets driving voltage is less than 15V which is considered to be the lowest driving voltage for material systems compatible with integrated circuits. Furthermore, the feature size of a typical CMOS gate is much smaller than that of an electrode (500 – 1000 μm). Therefore, the fabrication cost as well as the overhead of the whole chip area can be negligible.

Moreover, in our scheme, as will be demonstrated later, the logic function assigned to each electrode is only a simple logic function with a few number of logic variables, thus the latency and the power consumption of the integrated circuit is also not considerable compared to the whole system.

The above discussion demonstrates the practical applicability of the proposed scheme. Although a combinational logic

TABLE I: Comparison between the broadcast-addressing scheme [4], the proposed optimal and heuristic algorithms

	#E	Broadcast		Optimal		Heuristics					
		#p	CPU	#p	CPU	$k = 2$		$k = 3$		$k = 4$	
						#p	CPU	#p	CPU	#p	CPU
PCR	62	14	0.01	4	0.01	7	0.00	4	0.01	4	0.01
multiplex	59	7	0.01	3	0.01	4	0.00	3	0.01	3	0.01
DNA	77	17	0.01	4	0.01	10	0.01	7	0.01	4	0.01
multifunc	91	35	0.02	5	0.01	19	0.01	13	0.01	8	0.02
n -plex	1294	17	10.40	4	0.52	9	0.01	6	0.05	4	0.18
random01	296	14	3.22	3	0.02	6	0.01	3	0.54	3	0.01
random02	478	19	6.64	4	1.12	10	0.01	6	1.12	4	2.31
random03	1200	24	14.76	5	2.28	14	0.56	9	2.34	7	5.23
random04	1200	47	15.12	6	1.52	29	0.72	21	1.67	12	4.36
random05	1200	61	18.21	N.A.	N.C.	34	2.87	20	4.65	12	6.52
random06	881	25	9.52	N.A.	N.C.	17	1.43	11	5.87	8	7.28
random07	881	251	10.06	N.A.	N.C.	167	1.78	135	3.21	102	4.31
total		530		38(*)		326		238		171	

N.A.: Not available; N.C.: Not comparable; (*) total of only the available results.

circuit is integrated into the original chip, it has a negligible impact to the whole system in every aspect.

IV. EXPERIMENTAL RESULTS

The proposed optimal and heuristic algorithms are implemented by C++ on an Intel Core i5 2.67GHz Linux machine with 16GB Memory. In the optimal algorithm, an exact method for minimum clique partition problem proposed in [17] is used in our implementation. As for the heuristic strategy, as discussed in the previous section, the value k that represents the number of control pins on which each logic function for an electrode depends is set to 2, 3 and 4. To demonstrate the robustness and the scalability of our scheme, we compare the proposed method with the best state-of-the-art previous work, broadcast-addressing scheme [4] in terms of the number of control pins. A set of chip layouts for practical biological applications including polymerase chain reaction (PCR), multiplexed assay (multiplex), DNA sample preparation (DNA), multifunctional biochip (multifunc) and n -plex immunoassay (n -plex), and seven hard test cases (random01-random07) are used to evaluate the algorithms. The numbers of electrodes on each chips are denoted by “#E” in Table I. Besides, for each implementation (broadcast-addressing, proposed optimal and heuristic methods), the resultant number of control pins for the corresponding chips (denoted by “#p”) and the CPU execution time of the algorithms (denoted by “CPU”) are also reported in Table I.

As shown in Table I, the optimal method obtains significantly fewer number of control pins compared to the broadcast-addressing scheme. On average, we observe 80% reduction on the number of control pins when compared the optimal method to the broadcast-addressing method for the first nine test cases (five practical and four generated cases). However, for the last three cases, the execution time grows exponentially, and thus we cannot obtain the exact results for such cases.

When it comes to the heuristic strategy, obviously it cannot produce as good results as the optimal solution can, e.g., when $k = 4$, compared to the available results of the optimal solution, the heuristic approach obtains more than 22% number of control pins. However, it still outperforms the broadcast-addressing method. Specifically, for the three settings $k = 2$, $k = 3$ and $k = 4$, 38%, 55% and 67% pin-count reduction are observed when compared to the broadcast-addressing method, respectively.

As for the complexity of the logic functions, as mentioned

in Section III-D, the integrated logic circuit is very simple, and thus the overhead of the circuit is negligible in every aspect. In our experiments, we observe a very small number of literals in the representation of each logic function. For example, when $k = 2, 3, 4$, the average numbers of literals used to construct each logic function for the largest test case (random07) are 1.97, 4.95, 10.98, respectively. This demonstrates the simplicity of the integrated combinational logic circuit. Moreover, the heuristic approach provides us with a better evaluation of the complexity of the logic functions. It can be seen in Table I that when k becomes larger (from 2 to 4), i.e., the logic function of an electrode depends on more number of input variables, the number of control pins becomes smaller (from 326 to 171). It implies that a less number of control pins makes the logic function of each electrode become more complex and vice versa. However, studying this trade-off is out of the scope of this paper and remains for our future work.

V. CONCLUSION

In this paper, we proposed a new pin-count minimization design scheme for digital microfluidic biochips. By integrating logic functions into the biochips at negligible cost and area overhead, we can obtain the information-theoretic minimum number of control pins to handle all on-chip electrodes. Moreover, a heuristic algorithm dealing with large-scaled biochips is also proposed. Experiments demonstrate that the proposed optimal algorithm can obtain 80% reduction, and the proposed heuristic one can obtain at most 67% reduction on the number of control pins when compared to the broadcast-addressing algorithm, the best one in literature.

REFERENCES

- [1] F. Su, K. Chakrabarty and R. B. Fair, “Microfluidics-based biochips: technology issues, implementation Platforms, and design-automation challenges,” *IEEE TCAD*, vol. 25, no. 2, pp. 211-223, Feb, 2006.
- [2] M. G. Pollack, A. D. Shenderov and R. B. Fair, “Electrowetting-based actuation of droplets for integrated microfluidics,” *Lab on a Chip*, no. 2, pp. 96-101, May, 2002.
- [3] J. Gong and C. J. Kim, “Direct-referencing two-dimensional-array digital microfluidics using multilayer printed circuit board,” *IEEE JMEMS*, vol. 17, no. 2, pp. 257-264, Apr, 2008.
- [4] T. Xu and K. Chakrabarty, “Broadcast electrode-addressing for pin-constrained multi-functional digital microfluidic biochips,” *ACM/IEEE DAC*, pp. 173-178, Jun, 2008.
- [5] W. L. Hwang, F. Su and K. Chakrabarty, “Automated design of pin-constrained digital microfluidic arrays for lab-on-a-chip applications,” *ACM/IEEE DAC*, pp. 925-930, Jul, 2006.
- [6] T. Xu and K. Chakrabarty, “Droplet-trace-based array partitioning and a pin assignment algorithm for the automated design of digital microfluidic biochips,” *IEEE/ACM CODES+ISSS*, pp. 112-117, Oct, 2006.
- [7] S. K. Phan, C. Hashi and C. J. Kim, “Manipulation of multiple droplets on nxm grid by cross-reference EWOD driving scheme and pressure-contact packing,” *IEEE MEMS*, pp. 694-697, Jan, 2003.
- [8] C. C.-Y. Lin and Y.-W. Chang, “ILP-based pin-count aware design methodology for microfluidic biochips,” *IEEE TCAD*, vol. 29, no. 9, pp. 1315-1327, Sep, 2010.
- [9] Y. Luo and K. Chakrabarty, “Design of pin-constrained general-purpose digital microfluidic biochips,” *ACM/IEEE DAC*, pp. 18-25, Jun, 2012.
- [10] D. Grissom and P. Brisk, “A field-programmable pin-constrained digital microfluidic biochip,” *ACM/IEEE DAC*, pp. 1-9, Jun, 2013.
- [11] Y. Li et al., “Building EWOD microfluidic array technology on top of foundry CMOS,” in *IET Seminar on MEMS Sensors and Actuators*, pp. 23-30, 2006.
- [12] Y. Li et al., “Anodic Ta2O5 for CMOS compatible low voltage electrowetting-on-dielectric device fabrication,” *ESSDERC*, pp. 446-449, Sep, 2007.
- [13] B. Hadwen et al., “Programmable large area digital microfluidic array with integrated droplet sensing for bioassays,” *Lab on a Chip*, no. 18, pp. 3305-3313, Apr, 2012.
- [14] Y. Luo, K. Chakrabarty, and T.-Y. Ho, “A cyberphysical synthesis approach for error recovery in digital microfluidic biochips,” *IEEE/ACM DATE*, pp. 1239-1244, Mar, 2012.
- [15] R. Evans et al., “Optical detection heterogeneously integrated with a coplanar digital microfluidic lab-on-a-chip platform,” *IEEE Sensors*, pp. 423-426, Oct, 2007.
- [16] J. L. Gros and J. Yellen, “Graph theory and its applications,” *Boca Raton, FL: CRC Press*, 1999.
- [17] D. Brelaz, “New methods to color the vertices of a graph,” *Communications of the ACM*, vol. 22, no. 4, pp. 251-256, Apr, 1979.