Journal of Electronic Testing (2024) 40:573–587
https://doi.org/10.1007/s10836-024-06137-3

RESEARCH

Pebble Traversal-Based Fault Detection and Advanced Reconfiguration Technique for Digital Microfluidic Biochips

Basudev Saha[1] · Bidyut Das[2] · Vineeta Shukla[3] · Mukta Majumder[4]

Received: 16 April 2024 / Accepted: 20 August 2024 / Published online: 4 September 2024

Abstract
Digital Microfluidic Biochips (DMFBs) are rapidly replacing conventional biomedical analyzers by incorporating diverse bioassay operations with better throughput and precision at a negligible cost. In the last decade, these microfluidic devices have been well anticipated in miscellaneous healthcare applications such as DNA sequencing, drug discovery, drug screening, clinical diagnosis, etc., and other safety-critical fields like air quality monitoring, food safety testing, etc. In view of the application areas, these devices must incorporate the attributes like reliability, accuracy, and robustness. The correctness of a microfluidic device must be ensured through a superior testing technique before it is accepted for use in various applications. In this paper, an optimized fault modelling strategy to detect multiple faults in a digital microfluidic biochip has been introduced by embedding clockwise and anticlockwise movements of droplets using Pebble Traversal (based on Pebble Motion of Graph Theory). The suggested method also calculates traversal time for a fault-free biochip. In addition, this work presents an Advanced Module Sequence Graph-based reconfiguration technique to reinstate the microfluidic device for regular bioassays.

Keywords Digital microfluidic biochip · Pebble motion · Droplet traversal · Fault detection and reconfiguration · Manhattan distance · Advanced module sequence graph

1 Introduction

In recent times, microfluidic biochips are observed in numerous miniaturized pharmacological and biochemical laboratory applications; such as protein detection, protein and glucose analysis, drug delivery and discovery, drug screening, tissue engineering, cancer cell detection and diagnosis, food safety [1–5], etc. As these composite microsystems are replacing traditional benchtop laboratory equipment with better throughput and efficiency, they are also called lab-on-a-chip (LOC) [6]. The biochip device can be segregated into two categories based on the sample fluid transportation within the device. The microfluidic devices that use the unremitting flow of sample are termed Continuous-Flow Microfluidic Biochips (CFMBs) [6]. On the other hand, Digital Microfluidic Biochips (DMFBs) [7] are the devices that apply electro-wetting-on-dielectric (EWOD) [8] to manipulate nano or picoliter volume discrete droplets between two parallel electrode arrays [9, 10] as shown in Fig. 1. The advantages of DMFB over CFMB have been elaborated in the literature, such as easy integration, dynamic reconfiguration, and low cost [11, 12].

Since the application areas of biochip are observed diversely in various safety critical domains, the complexity of the device is also increased to achieve the higher precision and robustness. Due to the critical nature of the numerous clinical diagnosis and pharmacological applications of these

Responsible Editor: K. Chakrabarty

✉ Mukta Majumder
mukta_jgec_it_4@yahoo.co.in

Basudev Saha
mailtobasudev@gmail.com

Bidyut Das
bidyut2002in@gmail.com

Vineeta Shukla
vineeta.banasthali@gmail.com

[1] Department of Computer Science and Engineering, Sister Nivedita University, Kolkata 700156, West Bengal, India

[2] Department of Information Technology, Haldia Institute of Technology, Haldia 721657, West Bengal, India

[3] MathWorks India Private Limited, Bengaluru, Karnataka, India

[4] Department of Computer Science & Technology, University of North Bengal, Darjeeling 734013, West Bengal, India

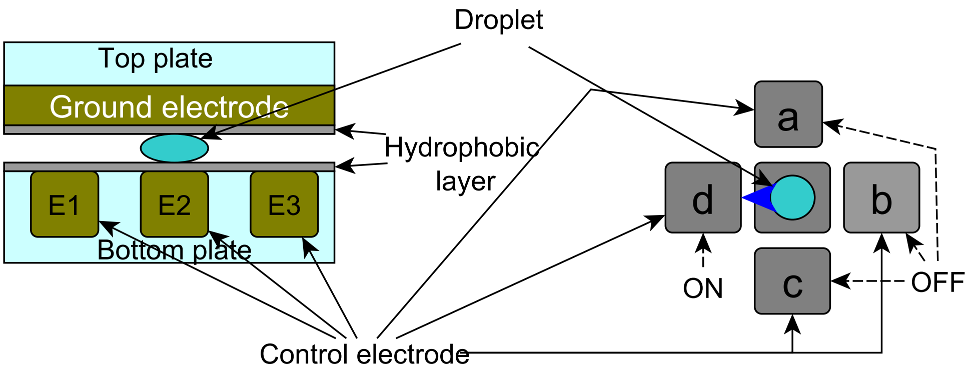

Fig. 1 Cross section view of a DMFB and move of droplet using control electrode

devices, it is essential to ensure their reliability through extensive testing strategies. Once a fault is detected, such device turns impractical to use. Consequently, the device could either be discarded or rectified using a suitable reconfiguration strategy to maintain its value for the low-cost market sector. Faults in a digital microfluidic biochip are categorized into two types: catastrophic and parametric [11, 13]. A catastrophic fault occurs when a test droplet is ceased at the faulty position during its transportation, leads to malfunction. The device performance suffers when there is a parametric or soft fault. The source and sink are connected with capacitive sensing circuits to identify the presence of test droplet.

The main contributions of this work are listed below:

1. This article introduces a novel Pebble Motion-based droplet traversal technique for examining the electrodes of a digital microfluidic biochip by transporting multiple test droplets.
2. The summarized outcome indicates that the proposed methodology optimizes the traversal time for a fault-free biochip.
3. If a fault is identified in a path, the proposed approach calculates the Manhattan distance to determine the best possible path to place the test droplet very next to faulty position and continue droplet traversal to address the unvisited electrodes in the path.
4. This article also introduces an Advanced Module Sequence Graph-based reconfiguration technique to reinstate the defective biochip.

A set of test droplets is transported by controlling the voltage of electrodes to traverse all of the edges and cells of the biochip to ensure that the device is fault-free. The initial task is to verify whether the device is defective or fault-free. If a fault is detected, it is necessary to identify the defective location. As soon as the faulty position is spotted, a reconfiguration technique is applied to reinstate the normal bioassay. Since a large biochip contains many spare cells (those are not involved in regular bioassay operation), the reconfiguration can be done by swapping the defective cell (swapping the functionalities of the defective cell) with one of the possible spare cell without deterring the droplet transportation. The testing and reconfiguration process should be prompt and economical to maintain the overall price of DMFBs [14]. After reconfiguration, a biochip can perform like a fresh one.

Here, we have introduced an optimized testing process using multiple droplets for digital microfluidic biochips. The approach allows offline testing by routing parallel droplets on a two-dimensional electrode array. The technique optimizes the droplet traversal time and scrutinizes faulty electrodes by incorporating Borderline Traversal with Inner Traversal. The Inner Traversal is conducted by routing of test droplets based on Pebble Motion [15]. If the device is fault-free all the test droplets can be observed at the sink using capacitive sensing circuit. After time out, if any of the droplets do not reach the sink, rollback is applied to retrieve the test droplet at the source via the same path [11, 16–19]. By calculating the rollback time and considering each time unit is equivalent to one edge traversal, location of the fault is estimated from the source electrode. If a fault is identified, an Advanced Module Sequence Graph (AMSG) based reconfiguration is applied to reintegrate the device for a typical bioassay.

The rest of this article is arranged as follows: In Section 2, we review the related literature. Section 3 describes the proposed fault detection technique. Section 4 summarizes the outcomes of the proposed method. Section 5 presents the feasible reconfiguration techniques for faulty biochips. Finally, Section 6 concludes the work.

2 Literature Review

A digital microfluidic biochip applies the electrowetting phenomenon to move nanoliter or picoliter droplets containing biochemical samples by applying the control voltage on a two-dimensional electrode array [20]. Some well-pronounced biochip models were described in the literature [21–23]. Since the biochip device is used in safety-critical and clinical applications, it should be error-free and robust. To ensure correctness, this device must pass a suitable testing methodology and reconfiguration policy. Subsequently, several researchers have attempted various testing strategies to reduce the traversal time for locating errors in a DMFB. Some of the prior works are described below.

Hu et al. proposed a checkpoint-based error recovery technique to address electrode-open faults in biochip [18]. They divided the microarray into multiple regions with a checkpoint in each region. And a capacitive sensing circuit was connected with each checkpoint. These sensor circuits were applied to detect faults on the droplet routing path; though these sensors could identify the faulty regions but unable to pinpoint multiple defective electrodes within that region. The proposed approach detoured the test droplet using neighboring electrodes towards the target, bypassing the defective path, but the author did not clearly mention how to identify this detour path. The installation of sensors in each region also increased the cost of the biochip. Majumder et al. introduced a novel method for detecting defective electrodes on a digital microfluidic biochip [19]. They employed a testing strategy by moving droplets through the neighbouring electrodes of a microarray in multiple loops. In the work, the authors carefully handled the undesirable mixing caused by neighbouring droplets at the adjacent electrodes. This approach involved substantial numbers of test droplets to navigate through the microarray. This incurs extended processing times. They applied the backtracking method to identify the faulty points.

Li et al. introduced a pipelined scan-like testing approach to decrease the overall measurement time [24]. The method selected ideal parameters, including voltage and frequency, for activating the electrodes which were used to distribute the electric field and evaluate the dielectric degradation. The challenge of managing a large number of pins required to activate the electrodes was resolved by pin-constrained chips which reduced testing time. In this approach, multiple modules were addressed by identical pin numbers. A fault diagnosis technique was proposed by Mukherjee et al. which involved distributed dispensing and scheduling of multiple test droplets in a synchronized manner [25]. The main objective of their method was to prevent routing conflicts in biochips, leading to improved fault detection capabilities.

A particle swarm optimization-based meta-heuristic fault identification technique was proposed by Mukherjee et al. [26]. A set of droplets was considered as particles and started circulating from the source reservoir to sink. In this approach, multiple droplets were passed through the same path as swarm particles. Another fault detection approach was proposed by Majumder et al. [27] where several droplets traversed through the left diagonal and right diagonal electrodes of the microarray to detect a faulty location. Ghosh et al. presented a fault detection method based on image processing [28]. They employed a CCD camera to capture an image of the biochip when all the test droplets didn't reach the sink. Saha and Majumder described various types of catastrophic and parametric faults and their reasons and circumstances [29]. They proposed a fault detection technique using two droplets to uncover double faults. The study took a considerable amount of time to navigate through a large microarray and did not tackle the issue of unvisited sites in case a fault was encountered in the path.

Ghosh et al. proposed a testing technique, namely, Multiple Electrodes Actuation Test (MEAT) using peripheral test, row-wise path test, multiple electrodes actuation test, and column-wise path test; with a huge number of test droplets [30]. Another testing strategy was proposed by Huang et al. where the authors demonstrated both offline and online testing [31]. They used a technique based on the Euler circuit for offline testing and a priority-based Genetic Algorithm approach for online testing. In this online testing, they addressed the fluidic constraints using backoff operation. Saha and Majumder introduced a multiple faults detection method based on Knight traversal approach by utilizing multiple droplets [32]. The authors identified the faulty locations by backtracking method. However, they did not address the issue of electrodes that remained unvisited due to the presence of fault in the path.

3 Proposed Fault Detection Technique

In this article, the proposed work is segmented into three sections. The first section introduces droplet movement patterns. The second section illustrates the multiple droplets traversal strategy from source reservoir to sink reservoir using several pseudo sources (base nodes) and pseudo sinks of a two-dimensional microarray to detect whether there exists any defective electrode. The third section identifies the location of the defective electrode in the microfluidic biochip.

3.1 Movement Patterns

Each cell and edge of a microarray must be visited at least once by any of the stimuli droplets to ensure the correctness of a biochip. To complete the traversal process, we have introduced a unique movement pattern for test droplets, named Pebble Traversal, based on Pebble Motion [15], where a droplet starts from an electrode and rotates once clockwise and then once anticlockwise through its adjacent electrodes. A clockwise rotation is executed when a test droplet starts from a cell and performs a left move followed by up, right, and down moves sequentially through the adjacent electrodes, as shown in Fig. 2(a). Similarly, the anticlockwise rotation is executed using one right, up, left, and down moves, as depicted in Fig. 2(b).

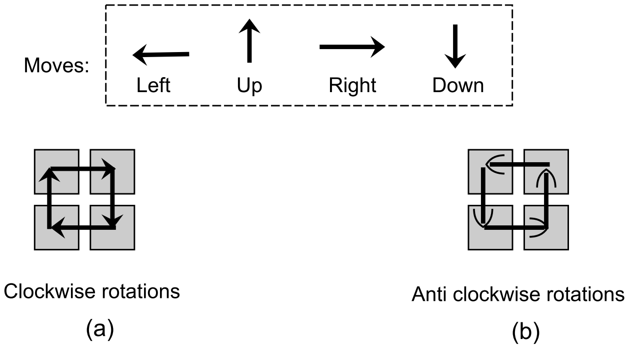

Fig. 2 Droplet movement patterns (a) clockwise rotation (b) anticlockwise rotation

If number of row is greater than the number of column of a microarray, then row-wise pebble traversal is executed and if column number is higher than the row number, then column-wise pebble traversal is carried out. The row-wise pebble traversal is termed as HPT (Horizontal Pebble Traversal) and column-wise pebble traversal is called VPT (Vertical Pebble Traversal). Figure 3 (a) shows HPT and its graph representation with visited nodes and edges of a 2D microarray. Similarly, Fig. 3(b) depicts VPT and its graph representation with the visited edges and cells. In case of square microarray, both approaches can be used and here, we are considering VPT (Vertical Pebble Traversal) for further calculation.

3.2 Proposed Traversal Procedure

In this subsection, we describe the traversal of a M×N microarray by applying multiple droplets, where M defines the number of columns and N signifies the number of rows. The Pebble Traversal-based technique combines the clockwise and anticlockwise rotations, which looks like slanted numerical digit '8'. Borderline Traversal and Inner Traversal (carried out by HPT or VPT) are used to ensure that each edge is traversed at least once.

For simplification of the M×N microarray, we have differentiated the columns as C_1, C_2, C_3,..., C_M and the rows as R_1, R_2, R_3,..., R_N, where M and N are the total numbers of columns and rows, respectively. In this work, we have considered that the source reservoir and sink reservoir are connected at (1,1) and (M,N), respectively, of the microarray. The journey of each droplet starts from (1,1) position and terminates at (M,N) position of the microarray. We have calculated one unit of time for each edge movement of the droplet.

Borderline Traversal is performed to visit the electrodes at the boundary of the microarray. Two droplets are dispatched from the source reservoir and routed parallelly in opposite directions through the boundary to complete the Borderline Traversal process. Therefore, these droplets finish the journey process after T_1 times which is calculated using Eq. 1. The Borderline Traversal procedure is depicted in Fig. 4.

$$T_1 = M + N \tag{1}$$

After completing Borderline Traversal, we have selected the pseudo sources for Inner Traversal based on the dimension of the microarray. Here, we consider the square microarray (6×6) for illustrating the process. The Inner Traversal is segmented into two passes. In the first pass, the test droplets are placed on the pseudo sources as C_2, C_6, C_{10},.., and so on. Figure 5 (a) shows C_2 and C_6 for the microarray of dimension 6×6. Then the Pebble Traversal (here, VPT is applied, since it is an square microarray) is applied repeatedly until the test droplets are steady at the bottom boundary. Next, the droplets are flowed out through the sink. One additional down move (shown blue arrows in Fig. 5) is required to carry out the Vertical Pebble Traversal of the test droplet. For example, Fig. 5 (a) shows the first pass where two test

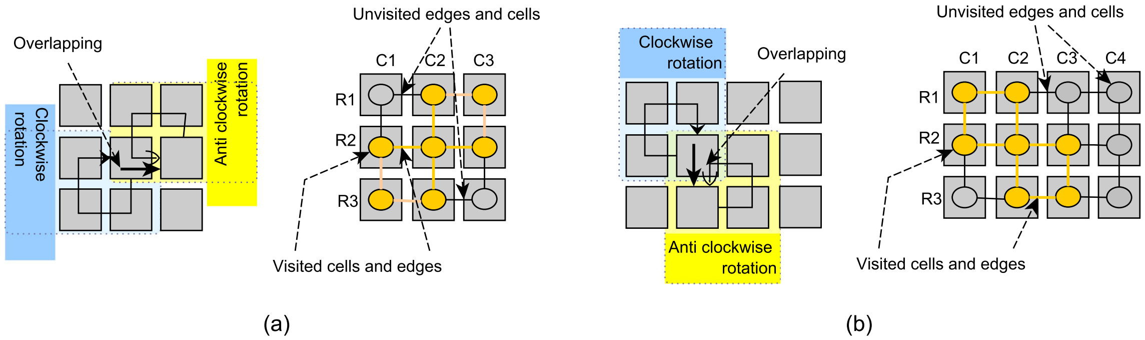

Fig. 3 (a) HPT (Horizontal Pebble Traversal) and graph representation with visited cells and edges (b) VPT (Vertical Pebble Traversal) and graph representation

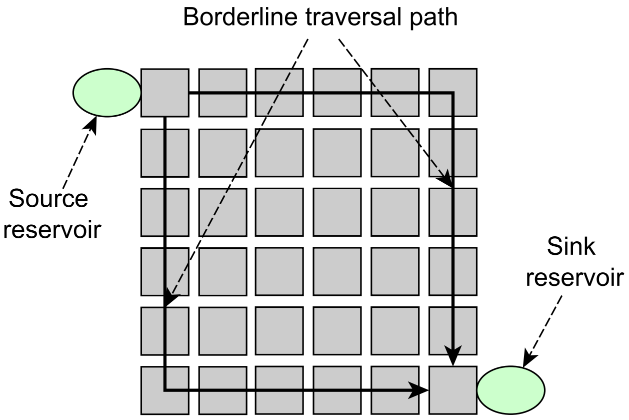

Fig. 4 Borderline Traversal process

droplets commence their journey from pseudo sources C_2 and C_6, and reach their respective pseudo sinks of the bottom boundary by adhering the VPT, indicated by bold arrows, along with an extra downward movement, as illustrated by the dotted arrows. If all the test droplets are seen at the sink reservoir within the stipulated time, a second pass of Inner Traversal will commence. In this pass, the test droplets are placed at new pseudo sources such as C_4, C_8,...so on (C_4 for 6×6 microarray as shown in Fig. 5(b)), and repeat the same. Figures 6 and 7 depict the Inner Traversal technique for 7×6 (using VPT) and 6×7 (using HPT) microarrays. If any test droplet does not reach the sink within the pre-calculated time, it indicates a defective electrode in the path. The proposed traversal approach is summarized in Algorithm 1, which uses Algorithm 2 to select the specific Pebble Traversal (HPT or VPT) based on the dimension of the microarray. The HPT and VPT procedures are abridged in Algorithms 3 and 4, respectively. Total execution time (T) is calculated by Eq. 2, where time for Horizontal Pebble Traversal ($T_2(HPT)$) and Vertical Pebble Traversal ($T_2(VPT)$) are computed using Eqs. 3 and 4, respectively. As Borderline Traversal or Inner Traversal individually does not cover each edge of the microarray, both of these traversals are obligatory to ensure that every edge of the microarray is routed at least once.

$$T = \begin{cases} T_1 + T_2(HPT) & \text{if N>M} \\ T_1 + T_2(VPT) & \text{otherwise} \end{cases} \tag{2}$$

$$T_2(HPT) = \begin{cases} 2(5M + 2N - 9) & \text{if } N \mod 4 = 3 \\ 2(5M + 2N - 8) & \text{otherwise} \end{cases} \tag{3}$$

$$T_2(VPT) = \begin{cases} 2(5N + 2M - 9) & \text{if } M \mod 4 = 3 \\ 2(5N + 2M - 8) & \text{otherwise} \end{cases} \tag{4}$$

Algorithm 1 Proposed Traversal Strategy

```
Require: 1. M×N microarray, where M is the number of columns and N is the number of rows.
         2. Source reservoir at $R_1C_1$ and sink reservoir at $R_NC_M$ locations of the microarray, where R and C represent row and column numbers.
Ensure: Traverse the entire microarray.
Two droplets are ejected from the source reservoir.
Start moving in opposite directions (clockwise and anticlockwise) along the boundary electrodes in the direction to the sink.
if (all droplets reached to sink reservoir) then                 ▷ Inner Traversal starts
    Calculate $\rho = \lfloor \frac{M}{2} \rfloor$ ▷ $\rho$ is the total number of droplets for Inner Traversal
    Compute $\alpha = \lceil \frac{\rho}{2} \rceil$                ▷ droplets for first pass
    Dispense $\alpha$ test droplets from source reservoir with four unit gaps.
    Place the droplets to their pseudo source using ($R_i$) for $N > M$, otherwise ($C_i$), where i=2, 6, 10, ..., N or M respectively.
    while (droplet have not steady to pseudo sink) do
        Call $Pebble\_Selection$                  ▷ Pebble Traversal: pass-1
    end while
    Move the droplets through the boundary electrode towards the sink reservoir.
    if (all droplets reached to sink reservoir) then
        Compute $\beta = \lfloor \frac{\rho}{2} \rfloor$          ▷ droplets for second pass
        Dispense $\beta$ test droplets from source reservoir with four unit gaps.
        Place the droplets to their pseudo source using ($R_j$) for $N > M$, otherwise ($C_j$), where j=4, 8, 12, ..., N or M respectively.
        while (droplet have not steady to end electrode) do
            Call $Pebble\_Selection$              ▷ Pebble Traversal: pass 2
        end while
        Move the droplets through the boundary electrode towards the sink reservoir.
    end if
end if
```

Algorithm 2 *Pebble_Selection*

```
if $N > M$ then
    Apply one Right move.
    if ($HPT$=true) then            ▷ Call Horizontal Pebble Traversal procedure
        Break.
    end if
else
    Apply one Down move.
    if ($VPT$=true) then ▷ Call Vertical Pebble Traversal procedure
        Break.
    end if
end if
```

3.3 Identify Fault Location

The proposed approach traverses the entire biochip using multiple test droplets. In addition to traversing the microarray, the proposed technique also identifies and locates the defective electrode by applying the roll-back method. If any test droplet does not reach the sink reservoir within the stip-

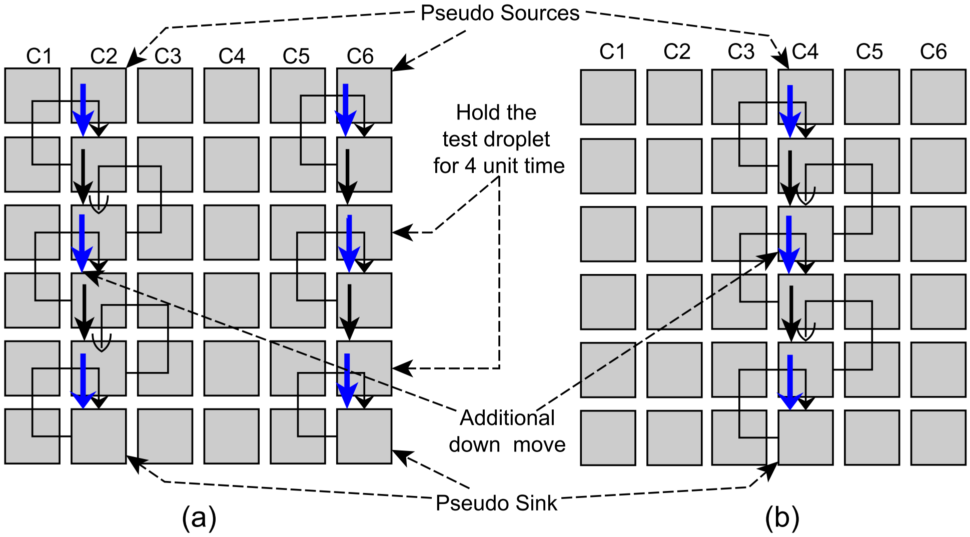

Fig. 5 Inner Traversal of a square microarray using VPT (a) first pass (b) second pass

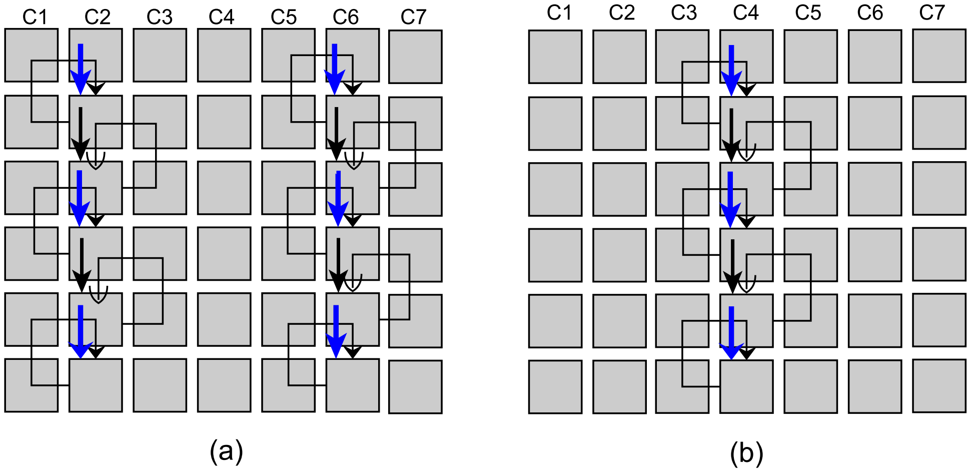

Fig. 6 Inner Traversal of a 7×6 microarray using VPT (a) first pass (b) second pass

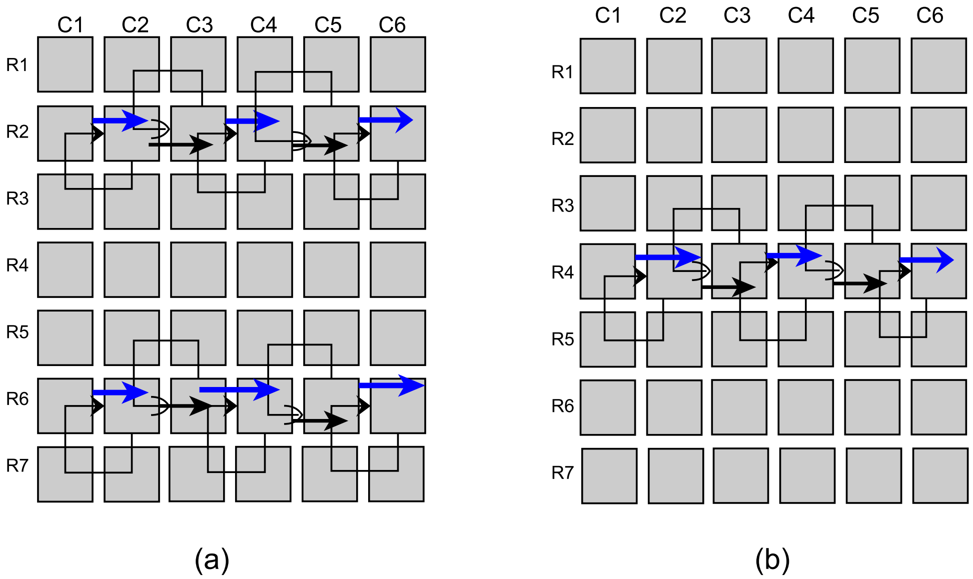

Fig. 7 Inner Traversal of a 6×7 microarray using HPT (a) first pass (b) second pass

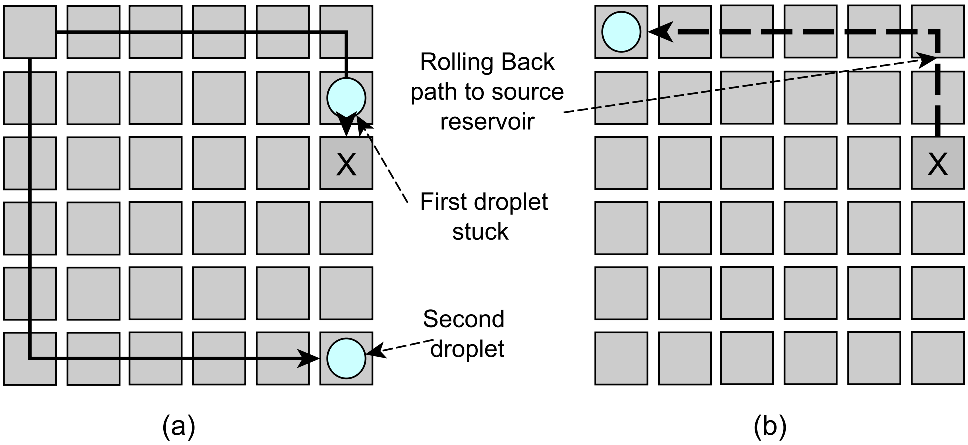

Fig. 8 (a) Droplet stopped on boundary electrode due to a fault (b) roll-back path from faulty location to source

Algorithm 3 HPT

```
Apply Down Left Up Right moves each.
if (reached pseudo sink) then
    Return true.
end if
Apply one Right move.
Apply Up Left Down Right moves each.
if (reached pseudo sink) then
    Return true.
end if
Return false.
```

Algorithm 4 VPT

```
Apply Left Up Right Down moves each.
if (reached pseudo sink) then
    Return true.
end if
Apply one Down move.
Apply Right Up Left Down moves each.
if (reached pseudo sink) then
    Return true.
end if
Return false.
```

ulated time, a fault is assumed to be present on the path as shown in Fig. 8(a). In this erroneous instance, the halted droplet is rolled-back to the source along its original journey path by applying the control voltage in the reverse direction [11, 16–19] (shown in Fig. 8(b)), and the roll-back time is calculated. In this process, initially, the roll-back starts from the first electrode of the path. If the droplet does not return to the source, then roll-back is initiated from the second electrode of the path. If this rolling back also fails, it will start again from the third electrode, and so on. In this iterative way, if the droplet takes $\mathcal{E}$ units of time for rolling back to its

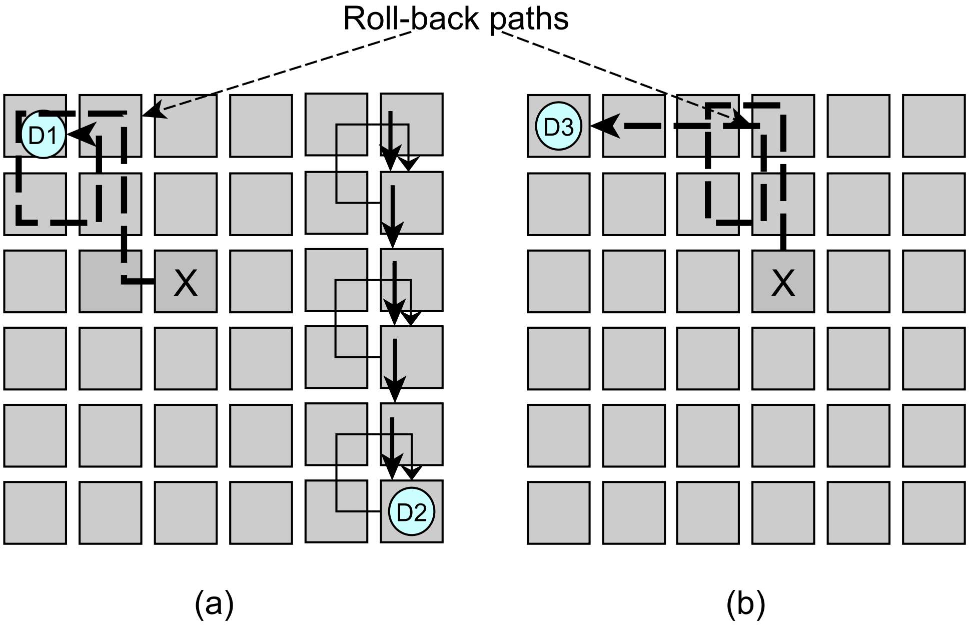

Fig. 9 Faults identification on internal electrodes (a) roll-back path in first pass (b) roll-back path in second pass

source electrode, this implies the fault is identified at $\mathcal{E}$ edges distance from the source electrode in the respective path.

Similarly, the roll-back process is applied to an erroneous situation during Inner Traversal. In case of multiple faults, the rolled-back process is applied identically to retrieve the test droplets at the source reservoir. Figure 9 illustrates the multiple faults identification process. Here, Fig. 9 (a) shows the roll-back path for droplet D_1 in the first pass. D_1 is rolled-back after the stipulated time, within that stipulated time D_2 already reached the sink by covering its journey path. Next, Fig. 9 (b) shows the roll-back path for D_3 in the second pass. When multiple droplets are ceased during the same pass, the roll-back procedure is initiated separately for each droplet, following their dispatch order one by one ($i.e.$, D_1, D_2, D_3...). Figure 10 shows the halted droplets in the first pass. Figure 11 depicts the rolled-back paths of halted droplets according to their order (first D_1, and then D_2 as shown in Fig. 11(a) & (b) respectively).

After detecting a fault, the halted droplet is rolled-back to the source reservoir, as a result, all the other electrodes along its path remain unvisited. Since the goal of the traversal process is to visit every edge and cell at least once, any site that remains unvisited must be traversed by another test droplet. To reach the next position of the faulty electrode, the test droplet follows the best possible path (shortest) using Manhattan distance [33], bypassing the defective location. Starting from this position, the traversal continues to cover the unvisited sites. The droplet placement and remaining traversal path are illustrated in Fig. 12.

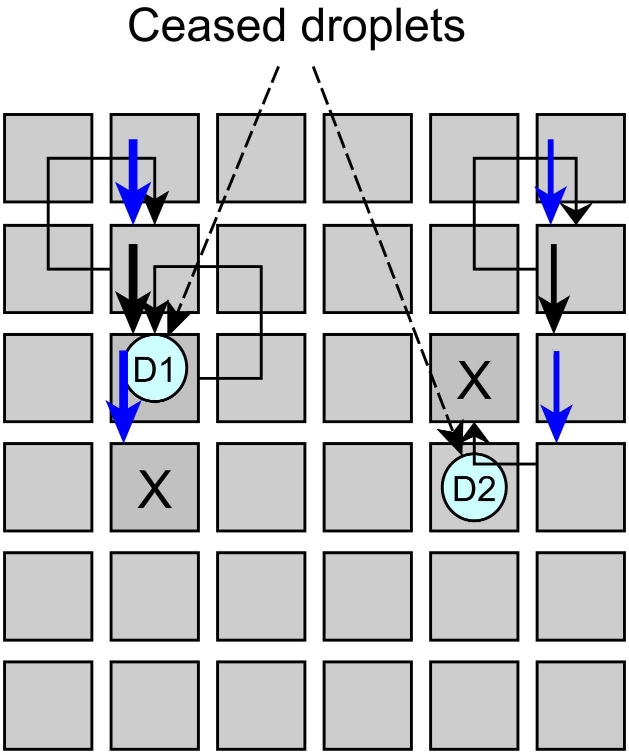

Fig. 10 Ceased droplets in an identical pass

4 Simulated Result

In this article, we have demonstrated the droplet traversal technique to detect multiple faulty locations in a microarray. The test droplets start from the source reservoir, traverse the entire microarray through the predefined path and finally remove via sink. In this process, the defective location will be uncovered if any test droplet stops its journey and does not reach the sink within the timeout duration. The proposed technique not only recognizes the faulty electrode but also counts the entire traversal time for a fault-free biochip.

We have routed the test droplets in parallel to optimize the traversal time. Parallelism is achieved by actuating the row electrodes (for vertical movement) or column electrodes (for horizontal movement) [14, 17, 34]. Due to the possibility that two drops may mix and violate the fluidic constraints [3] during concurrent execution, we have to be very meticulous while calculating the pseudo sources. In each traversal phase, droplets start their journey from (1,1) location of the microarray and end at (M,N) location. Those locations are connected with the source and sink reservoirs, respectively. As mentioned earlier, one unit of time has been considered for each edge movement of the test droplet, and according to this, we have computed the traversal time in the proposed approach.

The proposed traversal technique is simulated in OpenMP-C++ on a Linux multicore environment. We have considered a large number of microarrays from 5×5 to 48×48.

Table 1 shows the proposed time (T) using Eq. 2 to traverse the fault-free biochip, where T_1 is the time to complete Borderline Traversal and calculated using Eq. 1; $T_2(HPT)$ and $T_2(VPT)$ represent Horizontal Pebble Traversal time and Vertical Pebble Traversal time which are computed in Eqs. 3 and 4, respectively, as described earlier. In all cases, the droplet placement time from source reservoir to pseudo sources through (1,1) electrode, the traversal time from pseudo sources to pseudo sinks (end electrodes), and finally the droplet placement time from pseudo sinks to sink reservoir through (M,N) electrode have been considered. We have compared the result of our proposed technique with Parallel scan-like test [14] and Knight traversal [32] and listed them in Table 2. We have also computed the percentage of improvement using the formula given in Eq. 5. It is noticeable

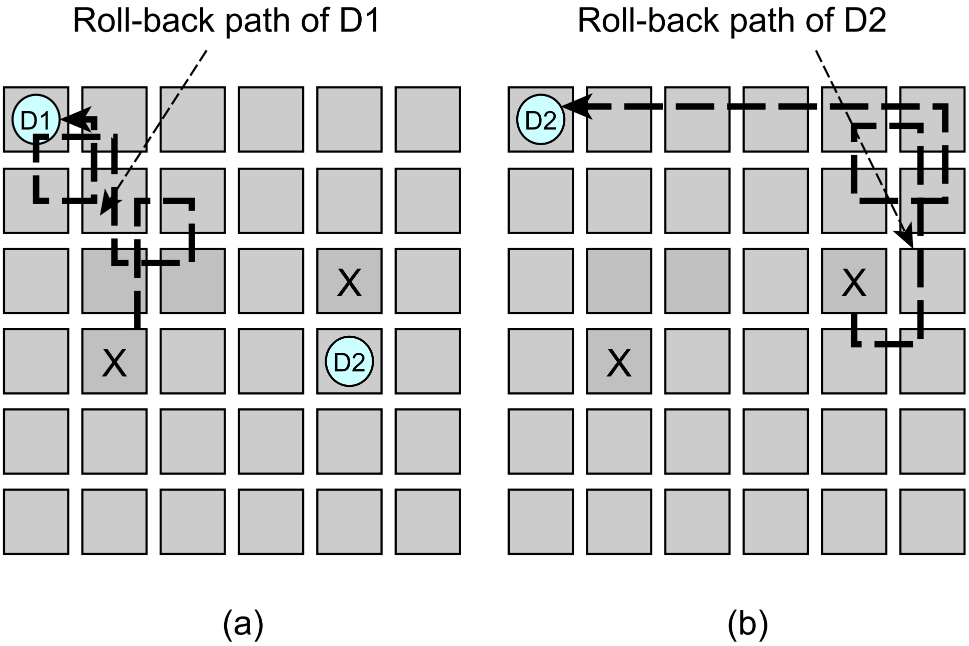

Fig. 11 (a) Roll-back path for ceased droplet D_1 (b) roll-back path for ceased droplet D_2 from their faulty locations

that the proposed technique shows far better result compared to Parallel scan-like test and moderate improvement over the Knight Traversal approach.

$$Improvement(\%) = \frac{existing - proposed}{existing} \times 100 \quad (5)$$

The proposed technique also requires a reduced number of test droplets to visit each electrode of the microarray. Here, we have considered n as the number of test droplets and T as the total traversal time. D^{TT} [32] is the metric, which defines the ratio between the number of used test droplets (n) and the traversal time (T), and it is calculated by Eq. 6. Table 3 shows a comparative study of the proposed technique with some of the contemporary works [11, 32, 35] in terms of D^{TT}.

$$D^{TT} = \frac{n}{T} \quad (6)$$

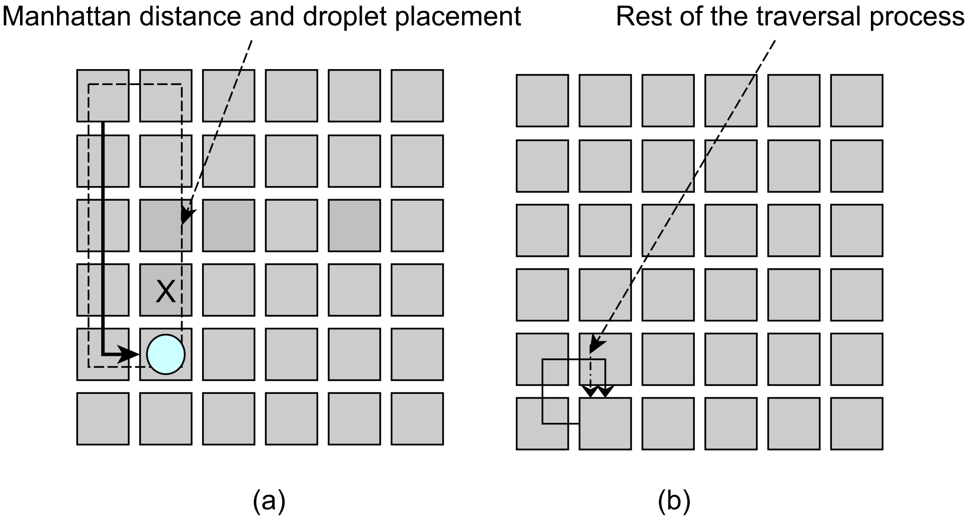
Manhattan distance and droplet placement
Rest of the traversal process
X
(a)
(b)

Fig. 12 (a) Droplet placement to cover unvisited sites (b) traversal process from new starting position

Table 1 Time to traverse of error-free microarrays

Matrix Size (M×N)	(T_1)	$T_2(HPT)$	$T_2(VPT)$	Proposed time (T)
5x5	10	-	54	64
7x8	15	86	-	101
9x9	18	-	110	128
10x8	18	-	104	122
11x9	20	-	116	136
14x16	30	188	-	218
15x15	30	-	192	222
17x18	35	226	-	261
23x22	45	-	294	339
25x25	50	-	334	384
28x28	56	-	376	432
32x32	64	-	432	496
36x36	72	-	488	560
42x42	84	-	572	656
48x48	96	-	656	752

5 Reconfiguration Technique

Reconfiguration is the prerequisite task if the proposed technique locates any faulty electrode in the testing phase. Prior to deployment of this device for safety-critical analysis, the reconfiguration process detours the droplet flow by avoiding the defective electrode in the microarray by changing the control voltages of some electrodes. Here, we consider the feasibility of reconfiguring a faulty biochip and propose a suitable online reconfiguration strategy.

5.1 Feasibility of Reconfiguration

The partial reconfiguration process relocates the defective module to a set of spare locations [36]. The tiny dimension of the microfluidic biochip is one of the major issues when attempting to relocate it. In case, insufficient availability of spare cells, the partial reconfiguration is more challenging. In this scenario, the local reconfiguration provides the way out by relocating the module to a significant spare location of that module [37]. However, local reconfiguration is not feasible

Table 2 Comparative study of proposed traversal time with Parallel scan-like test [14] and Knight traversal [32] approaches

Matrix Size	Existing [14]	Improvement(%)	Existing [32]	Improvement(%)	Proposed Time
5x5	80	20.00	62	-3.23	64
7x8	135	25.19	102	0.98	101
9x9	164	21.95	130	1.54	128
10x8	169	27.81	135	9.63	122
11x9	185	26.49	152	10.53	136
14x16	295	26.10	227	3.96	218
15x15	290	23.45	232	4.31	222
17x18	345	24.35	272	4.04	261
23x22	450	24.67	362	6.35	339
25x25	500	23.20	402	4.48	384
28x28	568	23.94	453	4.64	432
32x32	652	23.93	521	4.80	496
36x36	736	23.91	589	4.92	560
42x42	862	23.90	691	5.07	656
48x48	988	23.89	793	5.17	752

Table 3 Comparative study with existing D^{TT}

Matrix Size	Existing D^{TT} [11]	Existing D^{TT} [35]	Existing D^{TT} [32]	Proposed D^{TT}
5x5	0.2046	0.2703	0.129	0.0625
7x8	0.1892	0.3269	0.1078	0.0594
9x9	0.3607	0.1848	0.1	0.0469
10x8	0.3771	0.1848	0.1259	0.0574
11x9	0.3881	0.1827	0.1053	0.0515
14x16	0.4021	0.1768	0.1278	0.0459
15x15	0.4124	0.1768	0.0948	0.0405
17x18	0.4196	0.1753	0.0956	0.0421
23x22	0.4437	0.1732	0.0912	0.0383
25x25	0.4459	0.1725	0.092	0.0365
28x28	0.1719	0.2477	0.0762	0.0370
32x32	0.1712	0.2274	0.0758	0.0363
36x36	0.1707	0.2174	0.0756	0.0357
42x42	0.1701	0.22	0.0753	0.0351
48x48	0.1696	0.2058	0.075	0.0346

due to a lack of adjacent spare cells. A Module Sequence Graph (MSG) based reconfiguration technique was presented to overcome these issues [32]. The MSG approach relocates the cells of the faulty module to adjacent spare electrodes. MSR (Module-Spare Ratio) is the ratio between the adjacent spare cells (ASC) of a faulty module and the number of cells (C) of that faulty module and it is computed by Eq. 7.

$$MSR(\gamma) = \frac{ASC_{\gamma}}{C_{\gamma}} \tag{7}$$

where ASC_{γ} is the number of adjacent spare cells of defective module γ, C_{γ} is the number of cells of faulty module γ, and MSR_{γ} signifies Module-Spare Ratio of γ module. The MSG-based reconfiguration does not provide a solution in cases of low MSR and fails to produce a unique solution for higher MSR. In this work, we have proposed a reconfiguration technique by rearranging the cells of the faulty module that produces a unique solution. Additionally, the technique also reconfigures the defective module having low MSR value by rearranging certain fault-free modules.

The reconfiguration process is only possible with the available spare cells. Reconfiguration with Spare (R_S) [32] is the ratio of the number of spare cells (SC) and the number of active cells (AC) of the microarray and is defined by Eq. 8.

$$R_S = \frac{SC}{AC} \tag{8}$$

Local and partial reconfiguration techniques may suffer to relocate the defective module if R_S and MSR both are less than one [32]. An example of a 6×5 microarray is illustrated in Fig. 13(a), where the droplet path is marked from the 1st column of the 1st row to the 6th column of the 1st row. In this traversal path, the droplet passes through four different functional modules (named M_1, M_2, M_3, and M_4), and the R_S

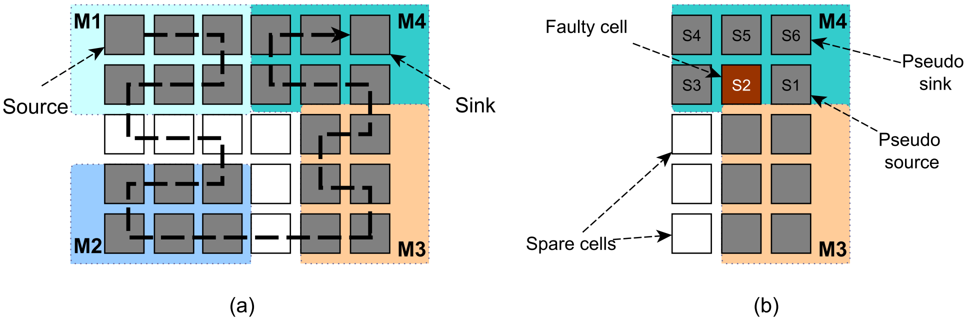

Fig. 13 (a) Droplet moving path for a 6×5 microarray (b) S_2 faulty cell in module M_4

value is 0.25 which is obtained by Eq. 8. The proposed fault detection technique detects a fault at the S_2 position of M_4 as shown in Fig. 13(b) and the MSR(M_4) is computed using Eq. 7 which is 0.17. It is observed that MSG-based reconfiguration is not feasible if the MSR value of a particular module is less than 0.25. Due to the lack of spare cells, local, partial, and MSG-based reconfigurations are not possible for M_4 as depicted in Fig. 13(b). However, local and MSG-based reconfigurations are feasible for M_3 since the MSR(M_3) is 0.5 as shown in Fig. 13(b). In this article, we have proposed an Advanced Module Sequence Graph (AMSG) based reconfiguration by combining the MSG approach with the local reconfiguration method to ensure an error-free bioassay using a pipeline of reconfiguration processes.

5.2 Proposed AMSG Reconfiguration

The proposed reconfiguration technique inherits the cell sequence and adjacent spare cells as introduced in [32]. Figure 14(a) shows the Module Sequence Graph representation of the faulty microarray from the previous example (Fig. 13(a)) with the sequence number of the defective module (M_4). The reconfiguration process starts with the MSR calculation of defective module M_4. Since the value of MSR is less than 0.25 for M_4, local reconfiguration, as well as MSG-based reconfiguration processes, are not allowed. The algorithm evaluates the possibility of relocating the previous module (M_3) to create additional spare cells adjacent to the defective module and applies the local reconfiguration since the MSR(M_3) is 0.5, and there is no such defect. Figure 14(b) shows the sequencing graph after applying the local reconfiguration of M_3. In this step, if the MSR(M_3) is less than 0.5, it implies local reconfiguration is not feasible; hence, the previous modules (γ, where $\gamma \implies M_2$, M_1) are examined repeatedly until the MSR(γ) $\geqslant$ 0.5. Subsequently, local reconfiguration is performed on γ. These steps are executed repeatedly until the MSR of defective module is increased to 0.25 or the total number of modules. After that, the MSG-based reconfiguration is applied to rearrange the flawed module as shown in Fig. 14(c) and restores the biochip for typical bioassay operation. The detailed procedure of AMSG-based reconfiguration is summarized in Algorithm 5.

The proposed AMSG algorithm also establishes a single solution, in case where multiple spare cells are available just before the pseudo source (S_1) of the defective module. It computes the shortest edge distance for all paths from the spare cells (adjacent to pseudo source) to pseudo sink and selects a spare cell as pseudo source that uses more spare cells in the respective path, to explore the possibility of relocation. Then, MSG-based reconfiguration is applied to complete the reconfiguration process.

Algorithm 5 AMSG Reconfiguration

```
Require: 1. Module sequence graph ($G$) of M×N microarray, where
    $M$ is the number of columns and $N$ is the number of rows.
      2. Module $M_i$ with $S$ set of vertices where $i$ is the number of
    modules.
      3. $F$ is the spare cells and defective module ($M_D$) with defective
    cell (D).
Ensure: Fault-free path for droplet traversal.
 1: Calculate $R_S$ for graph $G$ and MSR for $M_D$.
 2: while (MSR($M_D$) $<$ 0.25) do
 3:    k=1.
 4:    Calculate the MSR for immediate module ($M_D$-k).
 5:    if (MSR($M_D$-k) $<$ 0.5) and (k $\leq$ i) then
 6:       k=k+1.
 7:       Go to step 4.
 8:    else
 9:       if k $>$ i then
10:          Exit                              ▷ Reconfiguration not possible
11:       else
12:          Relocate $M_D$-k using the adjacent spare cells.     ▷ Local
    reconfiguration.
13:       end if
14:    end if
15:    Recalculate MSR for $M_D$.
16: end while
17: if (more than one adjacent F of pseudo source exists) then
18:    Let's consider, $F_\alpha$ and $F_\beta$ are two spare cells adjacent to the
    pseudo source.
19:    Calculate the edge distance from $F_\alpha$ and $F_\beta$ to the pseudo sink
    of $M_D$ with numbers of spare cells and plot to $\mathcal{P}_\alpha$ and $\mathcal{P}_\beta$.
20:    if ($\mathcal{P}_\alpha > \mathcal{P}_\beta$) then
21:       $F_r = F_\alpha$.
22:    else
23:       $F_r = F_\beta$.
24:    end if
25:    Relocate $S_1$ to $F_r$ and mark as $S_1^{'}$.
26:    Release $S_1$ as F.
27: end if
28: k=2.
29: while (k≤j) do          ▷ $j$ is the number of cells of defective module
30:    Relocate $S_k$ to an adjacent F and mark as $S_k^{'}$.
31:    if ($S_k \neq$ D) then
32:       Release $S_k$ as F.
33:    end if
34:    k=k+1.
35: end while
```

6 Conclusion

This article presents a multi-droplet traversal technique for detecting multiple faults and identifying these faulty sites within digital microfluidic biochips. Additionally, it determines the traversal time if the device is fault-free. The outcomes of the proposed approach show comparable per-

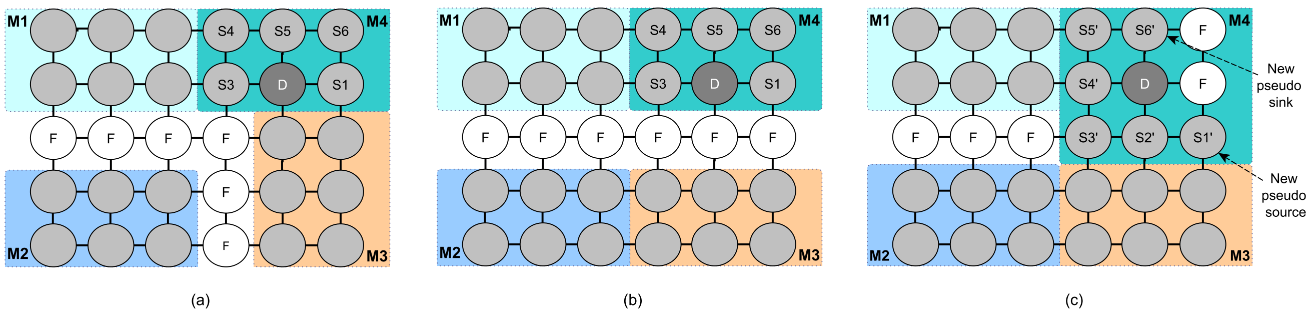

Fig. 14 (a) Module Sequence Graph representation of 6×5 microarray (b) After local reconfiguration of M_3 module (c) MSG-based reconfiguration on M_4 module

formance for calculating the journey time of the test droplet and the droplet traversal time ratio for a fault-free large microarray. It also presents an Advanced Module Sequence Graph (AMSG) based reconfiguration technique to detour the droplets by avoiding the defective electrode in this digital device. A potential future direction for this work is to develop an online fault detection method for digital microfluidic biochips.

Funding No funding was received for conducting this study.

Data Availability The article has no associated data.

Declarations

Conflicts of Interest The authors declare that there is no conflict of interest.

References

1. Vu Quoc T, Nguyen Ngoc V, Hoang B-A, Jen C-P, Duc TC, Bui TT (2023) Development of a compact electrical impedance measurement circuit for protein detection two-electrode impedance micro-sensor. IETE J Res 69(5):2478–2486
2. Raj M K, Chakraborty S (2020) Pdms microfluidics: A mini review. J Appl Polym Sci 137(27):48958. p 1–14
3. Saha B, Das B, Majumder M (2023) A deep-reinforcement learning approach for optimizing homogeneous droplet routing in digital microfluidic biochips. Nanotechnol Precis Eng 6(2):023001. p 1–12
4. Chan JY, Ahmad Kayani AB, Md Ali MA, Kok CK, Majlis BY, Hoe SLL, Marzuki M, Khoo AS-B, Ostrikov KK, Rahman MA et al (2018) Dielectrophoresis-based microfluidic platforms for cancer diagnostics. Biomicrofluidics 12(1):011503. p 1–21
5. Sajeesh P, Sen AK (2014) Particle separation and sorting in microfluidic devices: a review. Microfluid Nanofluid 17:1–52
6. Verpoorte E, De Rooij NF (2003) Microfluidics meets mems. In: Proc. of IEEE, vol 91, pp 930–953. IEEE
7. Keszocze O, Niemann P, Friedemann A, Drechsler R (2018) On the complexity of design tasks for digital microfluidic biochips. Microelectron J 78:35–45
8. Pollack MG (2001) Electrowetting-based microactuation of droplets for digital microfluidics. Dissertation. Duke University
9. Chakrabarty K, Xu T (2010) Digital Microfluidic Biochips: Design Automation and Optimization. CRC Press, USA
10. Hsieh Y-L, Ho T-Y, Chakrabarty K (2012) A reagent-saving mixing algorithm for preparing multiple-target biochemical samples using digital microfluidics. IEEE Trans Comput Aided Des Integr Circ Syst 31(11):1656–1669
11. Davids D, Datta S, Mukherjee A, Joshi B, Ravindran A (2006) Multiple fault diagnosis in digital microfluidic biochips. ACM J Emerg Technol Comput Syst 2(4):262–276
12. Li Z, Lai KY-T, Yu P-H, Chakrabarty K, Ho T-Y, Lee C-Y (2017) Structural and functional test methods for micro-electrode-dot-array digital microfluidic biochips. IEEE Trans Comput Aided Des Integr Circ Syst 37(5):968–981
13. Ho T-Y, Chakrabarty K, Pop P (2011) Digital microfluidic biochips: recent research and emerging challenges. In: Proceedings of the seventh IEEE/ACM/IFIP international conference on hardware/software codesign and system synthesis, pp 335–344
14. Xu T, Chakrabarty K (2007) Parallel scan-like testing and fault diagnosis techniques for digital microfluidic biochips. In: Proc. of 12th IEEE european test symposium (ETS'07), pp 63–68. IEEE
15. Kornhauser D, Miller G, Spirakis P (1984) Coordinating pebble motion on graphs, the diameter of permutation groups, and applications. In: 25th Annual symposium on foundations of computer science, 1984., pp 241–250. IEEE
16. Davids D, Joshi B, Mukherjee A, Ravindran A (2008) A fault detection and diagnosis technique for digital microfluidic biochips. In: Proc. of 14th international mixed-signals, sensors, and systems test workshop, pp 1–4. IEEE
17. Datta S, Joshi B, Ravindran A, Mukherjee A (2009) Efficient parallel testing and diagnosis of digital microfluidic biochips. ACM J Emerg Technol Comput Syst 5(2):1–17
18. Hu K, Hsu B-N, Madison A, Chakrabarty K, Fair R (2013) Fault detection, real-time error recovery, and experimental demonstration for digital microfluidic biochips. In: Proc. of design, automation & test in europe conference & exhibition (DATE), pp 559–564. IEEE
19. Majumder M, Das N, Saha SK (2013) A novel technique for multiple faults and their locations detection and start electrode selection in microfluidic digital biochip. J Innov Opt Health Sci 6(04):1350032. p 1–8
20. Zhao Y, Xu T, Chakrabarty K (2011) Broadcast electrode-addressing and scheduling methods for pin-constrained digital microfluidic biochips. IEEE Trans Comput Aided Des Integr Circ Syst 30(7):986–999
21. Su F, Chakrabarty K, Fair RB (2006) Microfluidics-based biochips: technology issues, implementation platforms, and design-automation challenges. IEEE Trans Comput Aided Des Integr Circ Syst 25(2):211–223

22. Shukla V, Hussin FA, Hamid NH, Ali NBBZ (2016) Investigation of capacitance dependence on droplet volume in meda based biochips. In: Proc. of 6th international conference on intelligent and advanced systems, pp 1–5. IEEE
23. Lu G-R, Kuo C-H, Chiang K-C, Banerjee A, Bhattacharya BB, Ho T-Y, Chen H-M (2018) Flexible droplet routing in active matrix–based digital microfluidic biochips. ACM Trans Des Autom Electron Syst 23(3):37. 1–25
24. Li Z, Dinh TA, Ho T-Y, Chakrabarty K (2014) Reliability-driven pipelined scan-like testing of digital microfluidic biochips. In: Proc. of 23rd asian test symposium, pp 57–62. IEEE
25. Mukherjee S, Samanta T (2015) Distributed scan like fault detection and test optimization for digital microfluidic biochips. J Electron Test 31(3):311–319
26. Mukherjee S, Pan I, Samanta T (2016) Algorithm for fault localization on a digital microfluidic biochip using particle swarm optimization technique. In: Proc. of international symposium on circuits and systems (ISCAS), pp 602–605. IEEE
27. Majumder M, Dolai U, Bhattacharya A (2017) An efficient novel single fault and its location detection technique using multiple droplets in a digital microfluidic biochip. In: Proc. of 11th international conference on intelligent systems and control (ISCO), pp 119–124. IEEE
28. Ghosh S, Rahaman H, Giri C (2018) Test diagnosis of digital microfluidic biochips using image segmentation. In: Proc. of 27th asian test symposium (ATS), pp 185–190. IEEE
29. Saha A, Majumder M (2019) An efficient technique for double faults detection and their locations identification in digital microfluidic biochip. Int J Autom Smart Technol 9(2):65–75
30. Ghosh S, Maity D, Chowdhury A, Roy SK, Giri C (2020) Efficient fault detection and diagnosis of digital microfluidic biochip using multiple electrodes actuation. In: Proc. of international test conference india, pp 1–4. IEEE
31. Huang X, Xu C, Zhang L (2020) An efficient algorithm for optimizing the test path of digital microfluidic biochips. J Electron Test 36:205–218
32. Saha B, Majumder M (2021) An optimized knight traversal technique to detect multiple faults and module sequence graph-based reconfiguration of microfluidic biochip. IET Comput Digit Tech 15(1):1–11
33. Pan I, Samanta T (2014) Weighted optimization of various parameters for droplet routing in digital microfluidic biochips. In: Recent advances in intelligent informatics: proc. of 2nd international symposium on intelligent informatics, pp 131–139. Springer
34. Bohringer KF (2006) Modeling and controlling parallel tasks in droplet-based microfluidic systems. IEEE Trans Comput Aided Des Integr Circ Syst 25(2):334–344
35. Chowdhury S, Datta P, Pal RK, Saha G (2021) An efficient multiple fault detection technique in digital microfluidic biochips. IETE J Res 67(6):899–912
36. Su F, Chakrabarty K, Pamula VK (2005) Yield enhancement of digital microfluidics-based biochips using space redundancy and local reconfiguration. In: Proc. of the conference on design, automation and test, pp 1196–1201. IEEE Computer Society
37. Su F, Chakrabarty K (2005) Reconfiguration techniques for digital microfluidic biochips. In: Proc. of the symposium on design, test, integration and packaging of MEMS/MOEMS 2005, pp 143–148. IEEE

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Basudev Saha received his BSc in Computer Science from University of North Bengal, Siliguri, India. He did his MCA and MTech in Information Technology from University of Calcutta, India. He is an Assistant Professor in the Department of Computer Science and Engineering, Sister Nivedita University, Kolkata, India. Formally he was an Assistant Professor of Department of Computational Science, Brainware University, Kolkata, India. His research interests include Microfluidic System, Digital Microfluidic-Biochips, Machine Learning and Deep Learning.

Bidyut Das completed his PhD in Computer Science & Engineering from Maulana Abul Kalam Azad University of Technology, West Bengal. He is an Associate Professor in the Department of Information Technology at Haldia Institute of Technology, located in West Bengal, India. He has received academic excellence awards such as a gold medal in his M.Sc. and a silver medal in his M.Tech. Before joining Haldia Institute of Technology, he worked as a research assistant at the Indian Institute of Technology, Kharagpur, and as a lecturer at Dr. B. C. Roy Engineering College, Durgapur. He has been honoured with prestigious fellowships such as the INSPIRE fellowship from the Department of Science & Technology, Govt. of India, and a Post-Doc fellowship from the Indian Institute of Technology, Guwahati. His research interests include ICT-based teaching and learning, natural language processing, computer vision, deep learning, and digital microfluidic biochips. Furthermore, he is an Editorial Board Member and Reviewer of numerous international journals.

Vineeta Shukla received her Ph.D from Department of Electrical & Electronic Engineering, Universiti Teknologi Petronas, Malaysia. She is a Senior Engineering Technical Writer at MathWorks India Private Limited, Bengaluru, India. Her research interests include Verification and Validation, VLSI Design and Testing, Digital Microfluidic-Biochips, and MEDA- based Biochips.

Mukta Majumder received his PhD from Department of Computer Science and Engineering, BIT Mishra, Ranchi, India. He is an Assistant Professor in the Department of Computer Science & Technology, University of North Bengal, Siliguri, India. Prior this he served Vidyasagar University as an Assistant Professor of Computer Centre from 2014 to 2017. His research interests include Natural Language Processing, Machine Learning, ICT Based Teaching-Learning and Assessment, Microfluidic System and Biochips.