



Article

Structural Optimization Design of Microfluidic Chips Based on Fast Sequence Pair Algorithm

Chuang Wu ^{1,2,3,*}, Jiju Sun ¹, Haithm Yahya Mohammed Almuaalemi ¹, A. S. M. Muhtasim Fuad Sohan ⁴ and Binfeng Yin ^{1,*}

¹ School of Mechanical Engineering, Yangzhou University, Yangzhou 225127, China; chuangstu@yzu.edu.cn (J.S.); mh22012@stu.yzu.edu.cn (H.Y.M.A.)

² Nantong Fuleda Vehicle Accessory Component Co., Ltd., Nantong 226300, China

³ Jiangsu Tongshun Power Technology Co., Ltd., Nantong 226300, China

⁴ Faculty of Engineering, Department of Mechanical Engineering, University of Adelaide, Adelaide, SA 5000, Australia; asmmuhtasimfuad.sohan@student.adelaide.edu.au

* Correspondence: 007511@yzu.edu.cn (C.W.); binfengyin@yzu.edu.cn (B.Y.)

Abstract: The market for microfluidic chips is experiencing significant growth; however, their development is hindered by a complex design process and low efficiency. Enhancing microfluidic chips' design quality and efficiency has emerged as an integral approach to foster their advancement. Currently, the existing structural design schemes lack careful consideration regarding the impact of chip area, microchannel length, and the number of intersections on chip design. This inadequacy leads to redundant chip structures resulting from the separation of layout and wiring design. This study proposes a structural optimization method for microfluidic chips to address these issues utilizing a simulated annealing algorithm. The simulated annealing algorithm generates an initial solution in advance using the fast sequence pair algorithm. Subsequently, an improved simulated annealing algorithm is employed to obtain the optimal solution for the device layout. During the wiring stage, an advanced wiring method is used to designate the high wiring area, thereby increasing the success rate of microfluidic chip wiring. Furthermore, the connection between layout and routing is reinforced through an improved layout adjustment method, which reduces the length of microchannels and the number of intersections. Finally, the effectiveness of the structural optimization approach is validated through six sets of test cases, successfully achieving the objective of enhancing the design quality of microfluidic chips.

Keywords: microfluidic chip; simulated annealing algorithm; fast sequence pair algorithm; structural design; optimization algorithm



Citation: Wu, C.; Sun, J.; Almuaalemi, H.Y.M.; Sohan, A.S.M.M.F.; Yin, B. Structural Optimization Design of Microfluidic Chips Based on Fast Sequence Pair Algorithm. *Micromachines* **2023**, *14*, 1577. <https://doi.org/10.3390/mi14081577>

Academic Editor: Xiangsong Feng

Received: 3 July 2023

Revised: 3 August 2023

Accepted: 8 August 2023

Published: 10 August 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Microfluidic technology is a scientific and technological system that integrates fundamental operational units, such as sample preparation, reaction, separation, and detection, into a chip, automating the analysis process [1–3]. Microfluidic chips serve as the primary platform for microfluidic technology, utilizing microfluidic channels and reaction chambers to process and manipulate small liquid volumes. Compared with traditional technology platforms, microfluidic chips enable faster completion of tasks that originally took hours, such as separation, isolation, and chemical and biological reactions, resulting in significantly improved detection efficiency [4–6].

The structure of a microfluidic chip primarily consists of a fluid layer and a control layer [7,8]. The fluid layer facilitates the reaction process of the experimental reagent. In contrast, the control layer connects to external pressure sources to regulate the flow and stoppage of the reagent within the fluid layer. The connection between the fluid layer and the control layer is established through microvalves, which serve as connectors and can also form devices within the microfluidic chip [9,10]. Figure 1 shows the chip reaction device

and a hybrid device containing microvalves. Microvalves will be placed at the crossing point when microchannels cross due to the need for experiments. A typical microfluidics chip can be integrated into hundreds of microvalves. Wiring between microvalves and control ports is required. A large number of microvalves will make wiring difficult. Due to the fact that channel crossing is not allowed in the control layer, the number of microvalves is huge. Once the position of the arrangement is incorrect, it will lead to malfunctions in the control layer. Therefore, it is necessary to reduce the number of microvalves and place them in appropriate positions [11–13]. Meanwhile, when the microchannel length is too long, the strong output of an external mechanical pump will affect the execution efficiency and time accuracy of biochemical experiments [14–16]. Secondly, when the area of microfluidic chips is too large, it will also cause a waste of raw materials and increase manufacturing costs [17–19]. Hence, the number of microchannel intersections, the microchannel length, and the chip area are critical indicators for evaluating the quality of chip structural design.

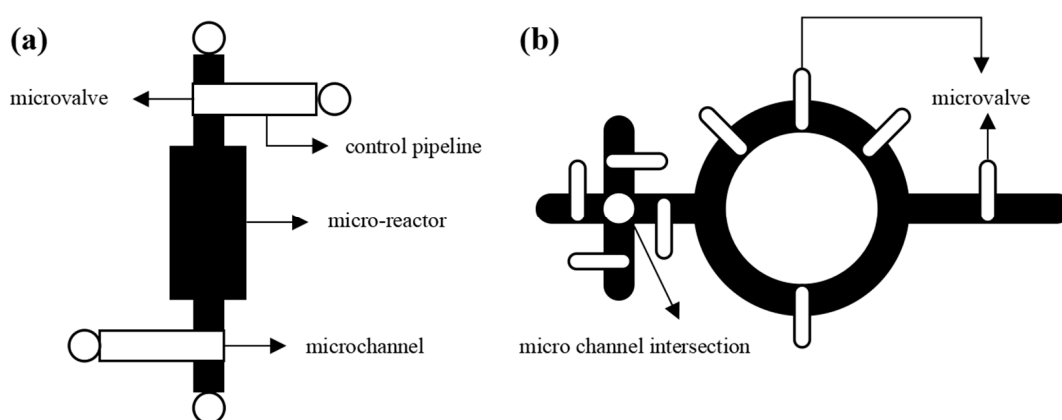


Figure 1. A device with microvalve structure: (a) schematic diagram of a chip reaction device; (b) schematic diagram of hybrid device.

Reference [20] proposed a component extension method that introduces heuristic diagonals in component extension. References [21,22] proposed an arbitrary angle routing algorithm. In response to storage issues, References [23,24] proposed a distributed channel storage method. References [25–27] proposed a weighted sum algorithm for the total length of microchannels. References [28,29] proposed a chip synthesis method for minimizing microvalves. However, in the above research, the layout of components and the routing of microchannels were considered separately, ignoring the interaction between the two, resulting in a decrease in quality and execution efficiency. To consider the interaction between the component layout and runner routing stage, Reference [30] proposed an effective layout and routing algorithm which allows iterative layout adjustment based on feedback information during micro runner routing. It uses a simulated annealing algorithm in the component layout stage, a negotiation-based routing algorithm in the routing stage, and iterative layout adjustment based on micro runner routing information. However, this method's convergence rate is slow, and the convergence quality is uneven.

Based on a Monte Carlo iterative solution strategy, the simulated annealing algorithm is a stochastic optimization algorithm commonly employed for combinatorial optimization problems [31,32]. Drawing inspiration from the annealing process of solid materials in physics, this algorithm provides an effective approximate solution for issues with non-deterministic polynomial (NP) complexity [33–35]. The traditional sequence pair algorithm is a classical layout representation method in the field of electronic design automation. It provides a coding method for the layout scheme of a given device set, which can speed up the enumeration and calculation of layout schemes [36,37].

Focusing on the goal of microfluidics chip structure optimization, this paper uses the improved Fast Sequence Pair (FAST-SP) algorithm to generate the initial solution of the simulated annealing algorithm, improves the cooling speed function of the simulated

annealing algorithm, and improves the Rate of convergence of the simulated annealing algorithm. Subsequently, the layout of devices and the routing of microchannels are integrated through layout adjustment using a chip layout quality evaluation function, which reduces the length of microchannels and the number of intersections. Finally, the comparison of data from six experiments demonstrates that the optimization algorithm proposed in this paper effectively enhances the design quality of microfluidic chips.

2. Structure Optimization of a Microfluidic Chip

2.1. Structural Modeling of Microfluidic Chips

1. Definition of a structural optimization problem

Problem input: experimental process, device summary, device connection relationship, and design constraints.

Problem output: design results, including device layout and microchannel routing results.

Optimization objects: chip area, microchannel length, and number of microchannel intersections.

Design goal: minimize the weighted sum of the chip area, the length of microchannels, and the number of microchannel intersections.

2. Structural design modeling

The structural design of microfluidic chips mainly includes device layout, channel routing, and device layout adjustment [38,39]. Figure 2 is a structural design modeling diagram of the microfluidic chip, and Figure 2a is an experimental sequence diagram showing the observed reaction's specific flow and the device's connection relationship. Each node represents a particular operation, such as detection, mixing, etc. Figure 2b summarizes the experimental apparatus. Figure 2c is a schematic diagram of the device layout and channel routing of the microfluidic chip, where the gray channel is the microchannel of the chip.

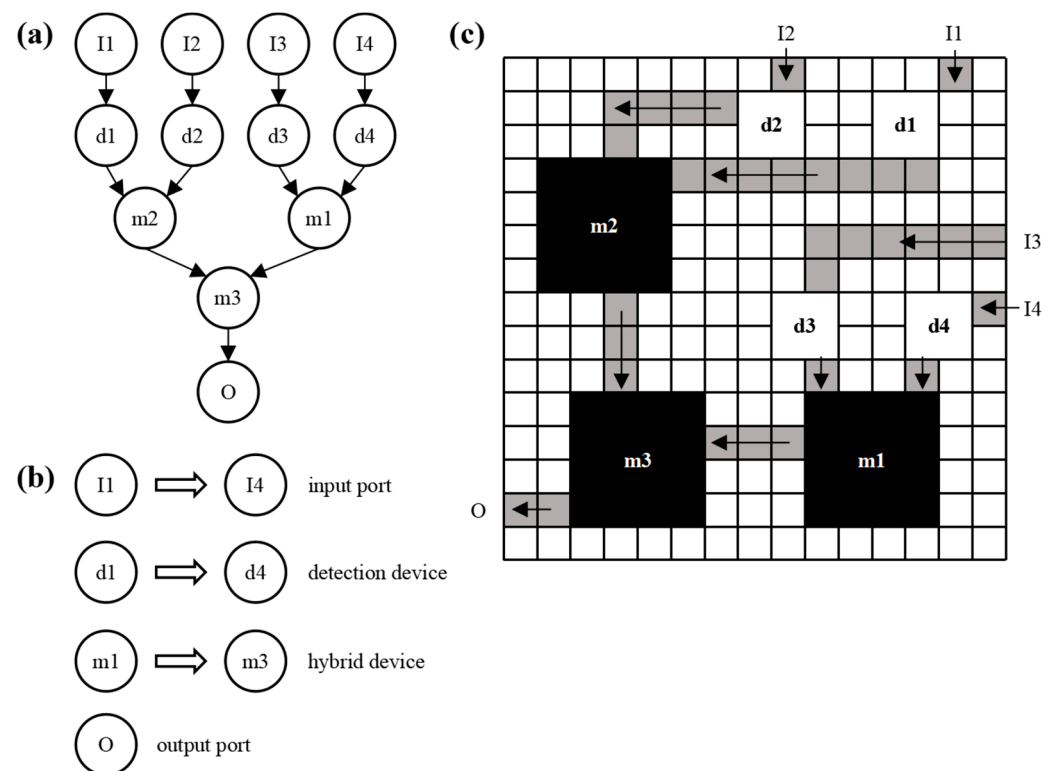


Figure 2. Structural design modeling of microfluidic chips: (a) experimental sequence diagram; (b) device summary; (c) channel wiring diagram of chip device layout.

2.2. Algorithm Design

In this paper, the FAST-SP algorithm [40–42] represents the initial setting scheme of the device layout. The exact position of each device on the microfluidics chip is calculated through the improved simulated annealing algorithm. Then, the chip layout design scheme is output according to the change process of heating, isothermal, cooling, and search strategies. As shown in Figure 3, the algorithm design process is as follows.

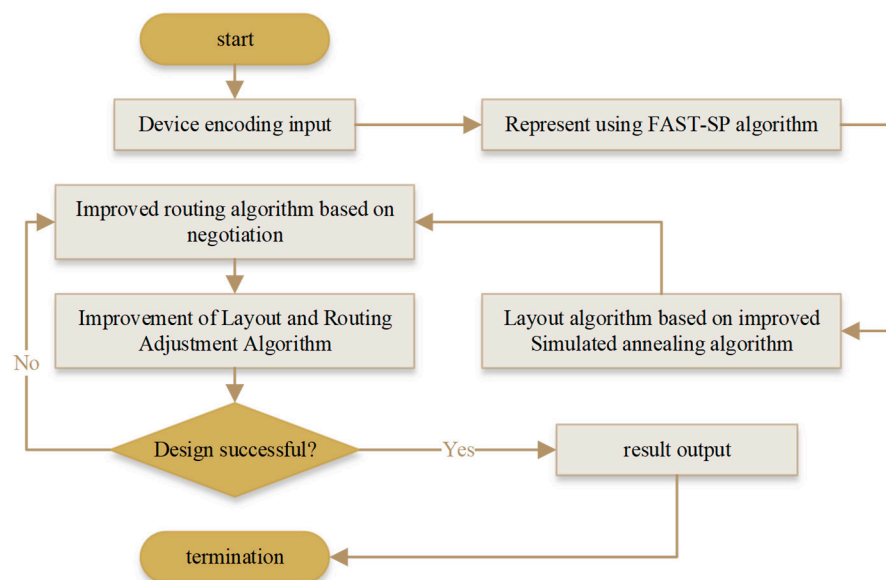


Figure 3. Algorithm design flow chart.

Upon completion of the layout, wiring the microchannels directly would result in experimental redundancy. To address this issue, the devices intended for connection are sequentially linked end to end, utilizing line segments as microchannel lengths in accordance with the experimental sequence diagram. The number of microchannel intersections can be determined by counting the intersection points between these line segments. Once the routing phase is finalized, the optimized layout adjustment algorithm can be employed to rectify the device layout and microchannel routing. The optimized layout adjustment algorithm preserves the initial sequence generated by FAST-SP, but adjusts the component spacing to allow for more routing possibilities. The objective is to eliminate unnecessary microchannel intersections, minimize the number of microvalves, and reduce the overall length of microchannels. Compared with the traditional sequence pair algorithm, the FAST-SP algorithm reduces the complexity of decoding time. After the longest common subsequence (LCS) algorithm is integrated, it speeds up the time to evaluate the layout of sequence pairs, making it more practical. Given a sequence pair, the structure's starting point, the device's width and height, and the layout direction can be obtained, and then a configuration can be generated. The starting point refers to the position where the first device is placed. When calculating the relative position between devices, the device's size is required, and the layout direction is conducive to minimizing the area of the chip. This paper adopts the layout scheme from bottom left to top right.

It is relevant to generate a layout from sequence pairs and find the longest weighted common subsequence in two sequence pairs. Determining the x coordinate of each device is equivalent to calculating LCS (S_X, S_Y), choosing the y coordinate of each device is equivalent to calculating LCS (S_X^R, S_Y^R), and S_X^R is the inverse sequence of S_X (Tables 1 and 2). For the device constraint relationship, each sequence unit in the sequence pair (S_X, S_Y) represents each device. Given two devices, a and b, in the sequences S_X and S_Y , if a is before b, then a is on the left of b; in sequence S_X , if a is before b, and in sequence S_Y , if a is after b, then a is above b.

Table 1. Sequence pair algorithm.

Input: sequence pair (S_X, S_Y) , width (length) of n devices, width $[n]$ (heights $[n]$).
 Output: x (y) coordinates x_coords (y_coords), the W (H) dimension of the layout structure.

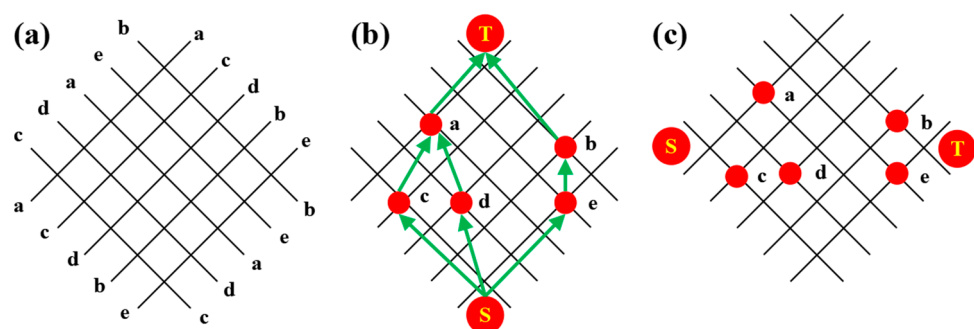
1. for($i = 1$ to n)
2. $weights[i] = widths[i]$ //Weight of device-width
3. $(x_coords, W) = LCS(S_X, S_Y, weights)$ //X coordinate, total width W
4. for($i = 1$ to n)
5. $weights[i] = heights[i]$ //Weight of device height
6. $S_X^R[i] = S_X[n + 1 - i]$ //Reverse S_X
7. $(y_coords, H) = LCS(S_X^R, S_Y, weights)$ //Y coordinate, total height H

Table 2. LCS algorithm.

Input: sequences S_1 and S_2 , weights of n devices $[n]$
 Output: position of each module, total length L

1. for($i = 1$ to n)
2. $block_order[S_2[i]] = i$ //Index of each device in S_2
3. $lengths[i] = 0$ //Total length initialization of all devices
4. for($i = 1$ to n)
5. $block = S_1[i]$ //Current device
6. $index = block_order[block]$ //Index of current device in S_2
7. $positions[block] = lengths[index]$ //Calculate the position of the device
8. $t_span = positions[block] + weights[block]$ //Determine the current fast length
9. for($j = index$ to n)
10. if($t_span > lengths[j]$)
11. $lengths[j] = t_span$ //The length of the current device replaces the former
12. else break
13. $L = lengths[n]$ //Total length

As shown in Figure 4, given a set of sequence pairs, the grid represents their corresponding constraint relationships. Draw a grid graph of $n \times n$, mark the grid lines from top to bottom with a positive sequence, and mark the grid lines from left to right with an inverse arrangement. Rotate -45° to obtain the oblique grid of sequence pairs $\langle a\ c\ d\ b\ e \rangle$ and $\langle c\ d\ a\ c\ b \rangle$. Given this group of sequence pairs, the layout vertical constraint graph (VCG) and layout horizontal constraint graph (HCG) can be obtained according to the start and endpoints.

**Figure 4.** Sequence pair layout: (a) skew network of sequence pairs; (b) layout vertical constraint diagram; (c) layout horizontal constraint diagram.

Lines 1–2 are used to initialize the width of n devices. Line 3 searches for the longest weighted common subsequence of S_X and S_Y and calculate the x coordinate of each device. Lines 4–5 initialize the height of n devices. Row 6 is used to obtain the reverse-order column S_X^R of S_X . Line 7 calculates the y -coordinate of each device according to the longest weighted common subsequence of S_X^R and S_Y .

Lines 1–2 represent vector `blocks_order`, which records the index of each device in S_2 . Line 3 initializes the vector length to 0, saving the maximum length of each device. Lines 4–5 indicate that the variable block is equivalent to the current device in S_1 . Line 6 indicates that the index is the index of the current device in S_2 . The block's position in Line 7 is defined as the position not occupied by other devices at first. Lines 9–12 update the lengths. The last quantity of Line 13 stores the total determined length.

- (1) Calculate the weighted LCS width of two sequences, S_1 and S_2 , and use the vector block order to record the index of each device in S_2 .
- (2) Initialize vector lengths to 0 to store each device's maximum quantity (length or width).
- (3) The variable block is defined as the current device in S_1 , and the index is the index of the current device in S_2 . All the items on the left of the block are arranged into intervals with the length of the block, and the block is then placed.
- (4) The total length of the updated layout is length, and length [n] indicates the full length after n devices are determined.
- (5) It is updated if the length [j] exceeds the current length.

Extract the LCS algorithm when $S_1 = S_X$, $S_2 = S_Y$, and weights = width, which can define the x coordinate of the layout. Extract the LCS algorithm when $S_1 = S_X^R$, $S_2 = S_Y$, and weights = heights, which can define the y coordinate of the layout.

2.3. Layout Calculation Optimization

The traditional simulated annealing algorithm has the problem that the parameters greatly impact the experimental results, and the convergence quality is not high when dealing with layout problems [43,44]. For this reason, this paper improves the simulated annealing algorithm, and the main improvement strategies are divided into two types: ① changing the factors of the algorithm itself; ② changing the search strategy, speeding up the search, and improving the search quality.

- (1) Change of algorithm cooling function

The control methods of simulated annealing in temperature cooling are divided into rapid cooling mode (RSA) and general cooling mode (CSA).

RSA: $T = T_0 / \log(1 + N)$; CSA: $T = qT_0 + k$, where q is the cooling rate and k is a constant.

Since the initial cooling rate of RSA and CSA is too fast to obtain the global optimal solution, this paper improves the algorithm from the cooling rate and proposes an improved simulated annealing algorithm (ISA), as shown in Formula 1:

$$T = T_0 \times e^{(-N^3/T_0^2)} \quad (1)$$

T_0 is the temperature at the initial time, N is the number of iterations required by the algorithm in the external cycle, and T is the current temperature.

As shown in Figure 5, compared with RSA and CSA, the ISA algorithm proposed in this paper decreases slowly in the high-temperature stage, achieving a global search of the algorithm, which is more conducive to generating the optimal solution.

The slow cooling rate can easily cause the slow global Rate of convergence. The convergence rate can be accelerated by combining the priority scheme of the FAST-SP algorithm with the ISA algorithm. The improvement methods are as follows:

- (1) Use reverse transformation to select two device units and reverse all units between the two units.
- (2) Select three device units and switch the unit between the two device units to the back of the third unit.

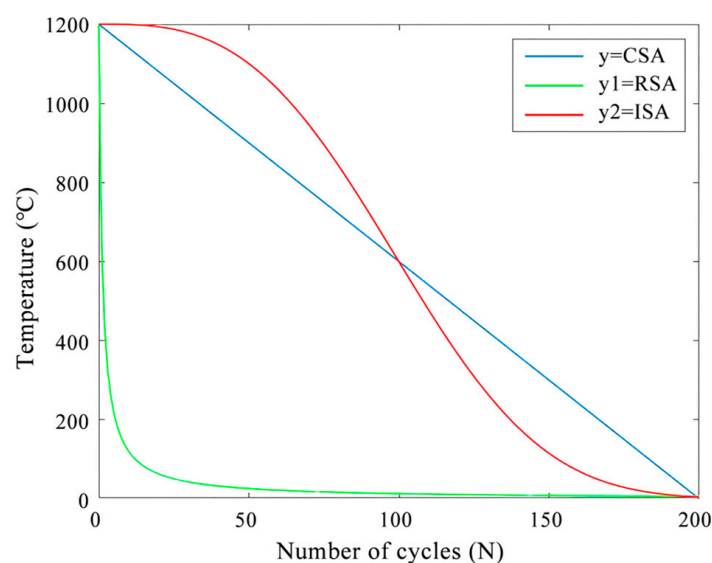


Figure 5. Cooling function curve, where CSA is the function curve of the available cooling mode, RSA is the function curve of the rapid cooling mode, and ISA is the function curve of the improved simulated annealing algorithm.

(2) Change of algorithmic search strategy

(1) Expand the recording function. During the implementation of the algorithm, the particularity of its selection probability may cause the problem that even if the optimal solution is generated, it may be abandoned. The extended recording function can save the optimal solution in this cycle for comparison with the results generated later.

Add a memory matrix (I) and a function (F) in the simulated annealing algorithm. Initially, there is an element i_0 in I , and $F = f(i_0)$. When generating a new solution, each time a new solution (j) is obtained, compare F with $f(j)$. If $f(j) < F$, let $F = f(j)$, and store j in I . After the algorithm is completed, the optimal solution is compared with the solution recorded in I to select the optimal solution as the final solution of the algorithm.

(2) For the solution of the initial solution, the simulated annealing algorithm with the recording function is used first. After the trial run of the entire algorithm is completed, the final result obtained is searched locally until the local algorithm search is conducted. Then, the final solution, namely the optimal solution, is output.

2.4. Routing Algorithm Optimization

The wiring of microfluidic chips can be divided into two stages. In the first stage, the A^* algorithm is mainly used for routing because there is no microchannel intersection [45–47]. If the wiring is successful, the result will be output. If the wiring fails, the second stage will be carried out. In the second stage, the microchannels are allowed to generate intersections during routing. Then, the routing is carried out through the improved A^* algorithm, adding additional generation value to the grid. Whether it is the first or second stage, the existing algorithms use routing based on the experimental response order [48,49]. Although the extra generation value of the routing grid can be iterated to obtain the routing scheme, the routing quality is not high.

As shown in Figure 6a, the blue box is a device, and the black line is a microchannel. When six devices are wired, a microchannel intersection will be generated, and the experimental reaction sequence is af, bf, and dcef. As shown in Figure 6b, the intersection disappears when the microchannel of connector f is wired first. It can be seen that key devices are essential to the results of microchannel wiring, and the probability that key devices can be wired in this area is higher than in other areas.

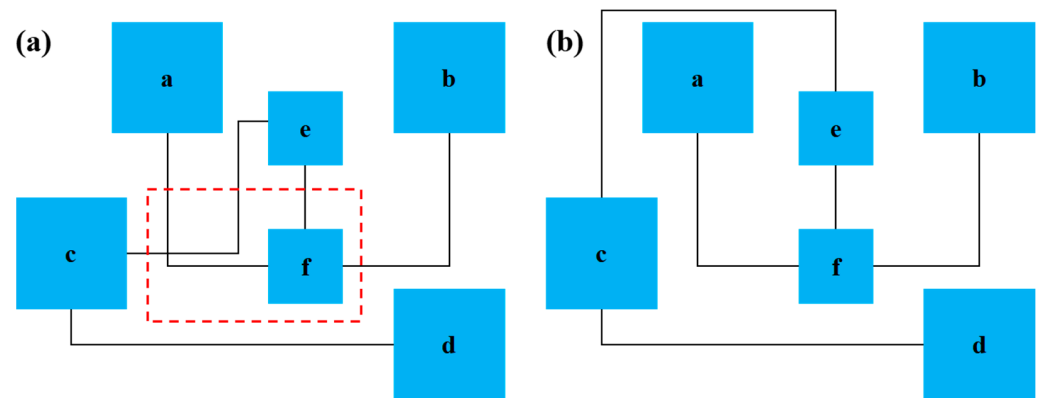


Figure 6. A schematic diagram of two types of wiring results: (a) there is an intersection; (b) no intersection exists.

For this reason, in the wiring stage of a microfluidic chip, this paper gives priority to the wiring of devices with multiple microchannels, defines the high wiring area (in the red dotted box) to distinguish it from other regions, and then wires the chip microchannels based on the improved wiring algorithm. In addition, the main purpose of wiring is to shorten the length of the microchannel and speed up the biochemical reaction time in the microchannel. To this end, three types of wiring modes, as shown in Figure 7, can also be selected to reduce the number of cross-points in microchannels.

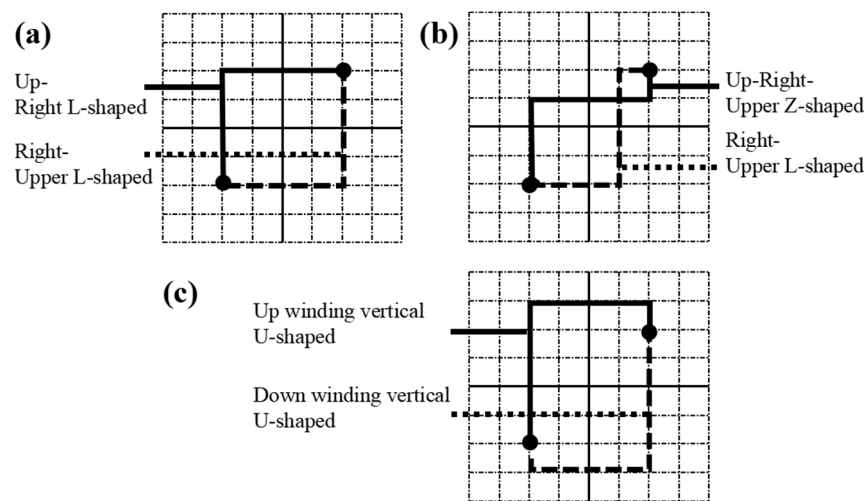


Figure 7. Wiring mode: (a) l-shaped wiring; (b) z-wiring; (c) u-shaped wiring.

2.5. Adjustment Algorithm Optimization

The previous device layout and microchannel routing results must be adjusted in the layout adjustment phase. Fine-tune the devices while keeping the relative position of the devices unchanged in the congested area; that is, keep the sequence pair order intact. The congestion area contains most devices, microchannels, and microchannel intersections. After adjustment, the next iteration will be carried out if the layout and wiring results are better than the previous work. They will be discarded if they are not as good as the earlier work [50–52]. However, this method only adjusts the congestion area, and the adjustment spacing is too small, which affects the overall layout quality to a certain extent and may result in the failure of layout adjustment.

For this reason, when adjusting the device layout and microchannel routing, this paper adjusts the devices in all areas where there are microchannel intersections and multiple channels, moving two units of length to the left, down, or right while maintaining the same relative position and moving one unit of measurement for devices in areas where

there is no microchannel intersection. The purpose is to provide space for machine and microchannel adjustment in other congested areas and ensure the algorithm can optimize the layout results without defects. If the result after adjustment is better than the previous scheme, the optimal solution will be output.

2.6. Improved Algorithm Flow

On the premise of device set and device connection relationships and with the device layout scheme obtained by the FAST-SP algorithm as the input, the following conditions are defined:

1. The distance between device m_i and device m_j on its right or above is defined as rx and axe .
2. The width and height of all device set spacing are defined as WX and WY . The constraint condition of elements between WX and WY is $[e_{\min}, e_{\max}]$. WX and WY form the initial solution S .
3. In the simulated annealing algorithm, the initial temperature is defined as T , the number of external cycles is N , the end temperature is T_{end} , the current temperature is T_0 , the chain length is L , and the quality function for evaluating the chip layout is $E(S)$.

The algorithm flow is:

- (1) Initialize WX and HY : $e_{\min} < rx, rx < e_{\max}$.
- (2) Set the state variables $S = (S_X, S_Y, WX, HY)$ and the initial temperature T . When the initial test temperature exceeds the minimum temperature, the iteration starts.
- (3) Adjust the state variable $S \rightarrow S_1$, randomly generate the variables rx^1 and ax^1 , and compare them with $e_{\min}$ and $e_{\max}$. When $rx^1 < e_{\min}$, let $e_{\min} = rx^1$; when $rx^1 > e_{\max}$, let $e_{\min} = rx^1$. The ax^1 is obtained in the same way.
- (4) Utilize Metropolis guidelines [53–55]:

$$df = E(S') - E(S) \quad (2)$$

$$P = \begin{cases} 1, & df < 0 \\ \exp(-df/T), & df \geq 0 \end{cases} \quad (3)$$

$$P' = \begin{cases} 1, & df < 0 \\ \exp\left(-df/\left(T_0 \times e^{(-N^3/T_0^2)}\right)\right), & df \geq 0 \end{cases} \quad (4)$$

If $df < 0$, the newly generated layout result will be accepted; otherwise, the new layout result will be obtained with probability $\exp(-df/T)$.

- (5) Cool down. Use the new cooling rate function to cool down. Stop iterating and output the current result if T_0 exceeds the end temperature.

Formula (4) is obtained by combining Formulas (2) and (3). The improved simulated annealing algorithm proposed in this paper converges quickly and can obtain a better solution when facing more devices.

For example, the parameters are $e_{\min} = 3$, $e_{\max} = 5$, $T = 10,000$, $T_{\text{end}} = 10^{-4}$, and $L = 200$, and the cooling rate is 0.95.

Evaluate the chip layout quality functions:

$$E(S) = \alpha A + \beta B + \gamma C + \theta C^2 \quad (5)$$

where A is the chip layout area, B is the number of microchannel intersections, C is the length of the microchannel segment, and C^2 is the square of the total segment. The main purpose is to minimize and enhance the length of the microchannel. Set the weight values of α to 1, β to 300, γ to 20, and θ to 0.001.

After the layout is completed, the A* algorithm can be used to find the shortest path after the layout. The input of the algorithm includes the following:

- ① Non-negative edge weight graph $G(WX, HY)$.
- ② A starting source node s .
- ③ One target end node t .

Then, two sets, M and N , are introduced, where M contains the point of the shortest path that has been found and the length of the shortest route, and N is the point of the shortest path that has not been found and the distance from the point to the source node.

Initialize the two sets M and N , find the shortest path point from the N set, add it to the M set, then update the devices in the N set, iterate circularly until the end of the traversal, and find the best routing result of the microfluidic chip.

3. Experimental Results and Analysis

We compared the proposed comprehensive optimization algorithm with the manual layout and existing algorithms [56] to verify the effectiveness of the proposed algorithm. Among them, the existing algorithm uses the basic simulated annealing algorithm, and this paper optimizes on this basis. Manual layout refers to a layout that has not undergone algorithm-optimization adjustments. The algorithm in this article was implemented using C++ programming language. The experimental platform was a 64-bit Windows server, configured with a 2.40 GHz Intel processor and 32 GB of memory to implement.

As shown in Table 3, in order to better compare the performance of algorithms, this paper selects six groups of test examples that can be completed without control layer iteration. Among them, the number of devices in the polymerase Chain reaction (PCR) group is 16, the number of devices in the InVitro-1~InVitro-3 groups is 30, 45, and 60, and the number of devices in the ProteinSplit-1 and ProteinSplit-2 groups is 30 and 66, respectively. The experimental results show that the integrated optimization algorithm in this paper reduced the chip area by 17.6%, the microchannel length by 20.9%, the number of microchannel intersections by 24.0%, and the convergence time was reduced by 16.9% on average, with the optimization ratio of PCR reaching 47.3%, compared with existing algorithms. From the improvement percentage, it can be concluded that when the number of devices was 45, the comprehensive optimization algorithm in this paper was optimal.

Table 3. Comparison of experimental results.

Test Case	Chip Area (mm ²) Existing Algo- rithm/Optimization Algorithm	Microchannel Length (mm) Existing Algo- rithm/Optimization Algorithm	Microchannel Intersection (pcs) Existing Algo- rithm/Optimization Algorithm	CPU Time (s) Existing Algo- rithm/Optimization Algorithm	Percent Improvement (%)
PCR	2958/2850	522/509	1/1	42.7/22.5	3.6
InVitro-1	5110/3906	802/765	1/1	84.1/63.7	23.5
InVitro-2	8232/5688	1485/1203	8/5	179.7/170.9	30.9
InVitro-3	11,187/8460	1864/1568	6/3	301.3/245.6	24.3
ProteinSplit-1	4914/4422	1162/713	5/3	114.0/106.9	10.0
ProteinSplit-2	17,030/14,690	3247/1749	42/35	528.0/527.7	13.7

Figures 8–10 compare the integrated optimization algorithm, the existing algorithm, and manual layout in the total chip area, the number of microchannel intersections, and the total length of microchannels. The figure shows the effectiveness of the integrated optimization algorithm in the structural design of microfluidic chips and the necessity of the mechanical structural optimization design of microfluidic chips.

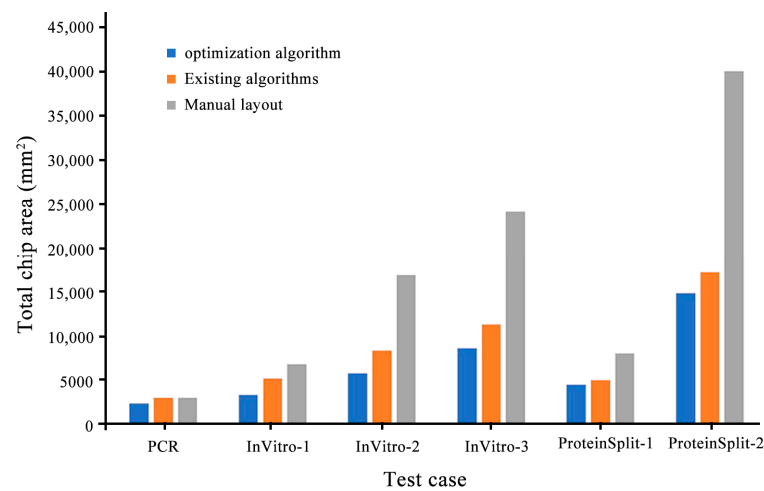


Figure 8. Comparison of three types of algorithms in total chip area.

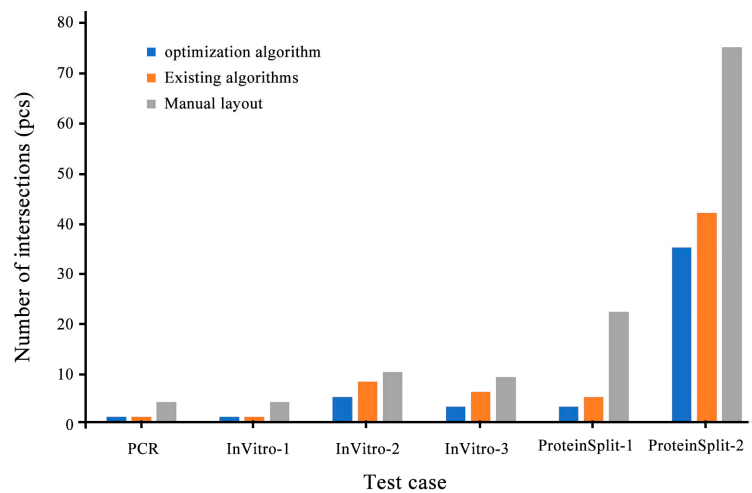


Figure 9. Comparison of three types of algorithms on the number of intersections.

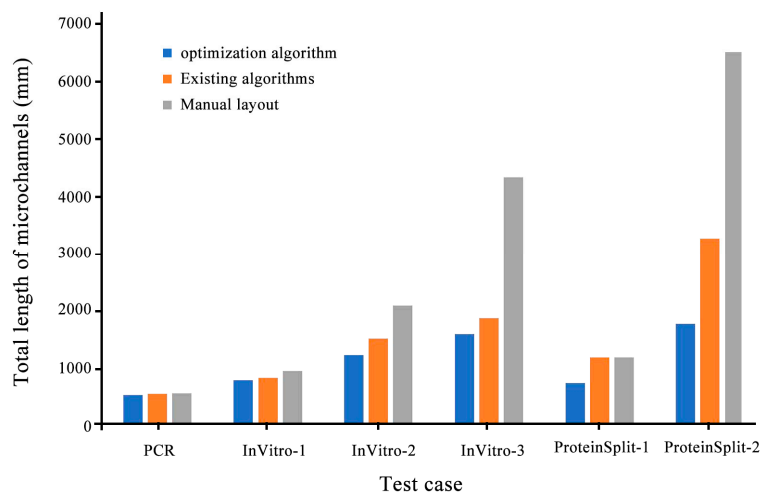


Figure 10. Comparison of 3D algorithms on the total length of microchannels.

The experimental results show that the optimization algorithm proposed in this paper can meet the automatic design requirements of microfluidic chips. The advantages of the optimization algorithm in the structural design of microfluidic chips are verified through

the reduction of the chip area, the shortening of the microfluidic channels, and the reduction of the number of intersections.

4. Conclusions

This FAST-SP algorithm addresses the challenges in chip layout design. By leveraging the FAST-SP algorithm, the proposed method accelerates the convergence rate of the simulated annealing algorithm, enhances its cooling rate and search strategy, mitigates the impact of parameters on the outcomes, and improves convergence quality. This approach targets the limitations of existing layout design methods for microfluidic chips, which often lack global optimization capabilities. Additionally, it enhances the routing method to minimize the number of intersections between microchannels. The key contribution is strengthening the interaction between microchannel routing and device layout. Six test cases were reconducted to evaluate the proposed algorithm's effectiveness, with chip area, microchannel length, and the number of intersections serving as optimization objectives. The results clearly state the superiority of the optimization algorithm presented in this paper compared with existing design methods in these three areas. Notably, the experimental data reveal that when the number of devices reaches 45, the algorithm achieves the optimal improvement percentage, highlighting its capability to obtain optimal solutions and exhibit robustness.

Through enhancements, the optimization algorithm achieved an average reduction of 17.6% in the total chip area, 20.9% in the overall microchannel length, 24.0% in the number of microchannel intersections, and 16.9% in the convergence time. These quantitative results unequivocally validate the effectiveness of the proposed optimization algorithm in optimizing the structure of microfluidic chips and enhancing their design quality paper, introducing an improved simulated annealing algorithm that builds upon the FAST-SP.

Author Contributions: Conceptualization, C.W. and B.Y.; methodology, C.W. and J.S.; software, J.S. and H.Y.M.A.; validation, C.W., H.Y.M.A., A.S.M.M.F.S. and B.Y.; formal analysis, B.Y.; investigation, J.S. and H.Y.M.A.; resources, B.Y.; data curation, C.W.; writing—original draft preparation, C.W., H.Y.M.A. and A.S.M.M.F.S.; writing—review and editing, C.W. and B.Y.; visualization, C.W.; supervision, B.Y.; project administration, B.Y.; funding acquisition, C.W. and B.Y. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Basic Science (Natural Science) Research Project of Jiangsu Province Higher Education Institutions [grant number 23KJD430014], the Innovative Science and Technology Platform Project of Cooperation between Yangzhou City and Yangzhou University, China [grant number YZ2020266], the “Green Yang Golden Phoenix Project” in Yangzhou City [grant number YZLYJFJH2021YXBS098], the National Natural Science Foundation of China [grant number 52075138], the Natural Science Foundation of the Jiangsu Higher Education Institutions of China [grant number 22KJB150050], the Jiangsu Agricultural Science and Technology Innovation Fund [grant number CX(21)3162], the Science and Technology Planning Project of Yangzhou City [grant number YZ2022180], and the Market Supervision Administration Science and Technology Fund of Jiangsu Province [grant number KJ2023076].

Data Availability Statement: Data sharing is not applicable to this article.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Yin, B.; Yue, W.; Sohan, A.; Zhou, T.; Qian, C.; Wan, X. Micromixer with Fine-Tuned Mathematical Spiral Structures. *ACS Omega* **2021**, *6*, 30779–30789. [\[CrossRef\]](#)
2. Yin, B.; Wan, X.; Qian, C.; Sohan, A.; Wang, S.; Zhou, T. Point-of-Care Testing for Multiple Cardiac Markers Based on a Snail-Shaped Microfluidic Chip. *Front. Chem.* **2021**, *9*, 741058. [\[CrossRef\]](#)
3. Li, R.; Wei, X.; Xu, J.; Chen, J.; Li, B.; Wu, Z.; Wang, Z.L. Smart Wearable Sensors Based on Triboelectric Nanogenerator for Personal Healthcare Monitoring. *Micromachines* **2021**, *12*, 352. [\[CrossRef\]](#)
4. Yin, B.F.; Wan, X.H.; Yang, M.Z.; Qian, C.C.; Sohan, A. Wave-shaped microfluidic chip assisted point-of-care testing for accurate and rapid diagnosis of infections. *Mil. Med. Res.* **2022**, *9*, 8–21. [\[CrossRef\]](#)

5. Yin, B.; Qian, C.; Wan, X.; Muhtasim Fuad Sohan, A.S.M.; Lin, X. Tape integrated self-designed microfluidic chip for point-of-care immunoassays simultaneous detection of disease biomarkers with tunable detection range. *Biosens. Bioelectron.* **2022**, *212*, 114429. [\[CrossRef\]](#)
6. Yin, B.; Wan, X.; Qian, C.; Sohan, A.; Zhou, T.; Yue, W. Enzyme Method-Based Microfluidic Chip for the Rapid Detection of Copper Ions. *Micromachines* **2021**, *12*, 1380. [\[CrossRef\]](#)
7. Shin, W.; Kim, H.J. 3D in vitro morphogenesis of human intestinal epithelium in a gut-on-a-chip or a hybrid chip with a cell culture insert. *Nat. Protoc.* **2022**, *17*, 910–939. [\[CrossRef\]](#)
8. Zhang, Y.; Tseng, T.M.; Schlichtmann, U. Portable all-in-one automated microfluidic system (PAMICON) with 3D-printed chip using novel fluid control mechanism. *Sci. Rep.* **2021**, *11*, 19189–19199. [\[CrossRef\]](#)
9. Aishan, Y.; Funano, S.I.; Sato, A.; Ito, Y.; Ota, N.; Yalikun, Y.; Tanaka, Y. Bio-actuated microvalve in microfluidics using sensing and actuating function of *Mimosa pudica*. *Sci. Rep.* **2022**, *12*, 7653–7664. [\[CrossRef\]](#)
10. Gong, J.; Wang, Q.; Liu, B.; Zhang, H.; Gui, L. A Novel On-Chip Liquid-Metal-Enabled Microvalve. *Micromachines* **2021**, *12*, 1051. [\[CrossRef\]](#)
11. Im, S.B.; Uddin, M.J.; Jin, G.J.; Shim, J.S. A disposable on-chip microvalve and pump for programmable microfluidics. *Lab. A Chip* **2018**, *18*, 1310–1319. [\[CrossRef\]](#)
12. Liu, W.; He, H.; Zheng, S.Y. Microfluidics in Single-Cell Virology: Technologies and Applications. *Trends Biotechnol.* **2020**, *38*, 1360–1372. [\[CrossRef\]](#)
13. Peshin, S.; Madou, M.; Kulinsky, L. Microvalves for Applications in Centrifugal Microfluidics. *Sensors* **2022**, *22*, 8955. [\[CrossRef\]](#)
14. Hayes, B.; Smith, L.; Kabutz, H.; Hayes, A.C.; Whiting, G.L.; Jayaram, K.; MacCurdy, R. Rapid Fabrication of Low-Cost Thermal Bubble-Driven Micro-Pumps. *Micromachines* **2022**, *13*, 1634. [\[CrossRef\]](#)
15. Gao, R.Z.; Hébert, M.; Huissoon, J.; Ren, C.L. μ Pump: An open-source pressure pump for precision fluid handling in microfluidics. *HardwareX* **2020**, *7*, e00096. [\[CrossRef\]](#)
16. Bao, Q.; Zhang, J.; Tang, M.; Huang, Z.; Lai, L.; Huang, J.; Wu, C. A Novel PZT Pump with Built-in Compliant Structures. *Sensors* **2019**, *19*, 1301. [\[CrossRef\]](#)
17. Tang, P.C.; Eriksson, O.; Sjögren, J.; Fatsis-Kavalopoulos, N.; Kreuger, J.; Andersson, D.I. A Microfluidic Chip for Studies of the Dynamics of Antibiotic Resistance Selection in Bacterial Biofilms. *Front. Cell. Infect. Microbiol.* **2022**, *12*, 896149. [\[CrossRef\]](#)
18. Reyes, D.R.; van Heeren, H. Proceedings of the First Workshop on Standards for Microfluidics. *J. Res. Natl. Inst. Stand. Technol.* **2019**, *124*, 1–22. [\[CrossRef\]](#)
19. Tian, L.L.; Li, C.H.; Ye, Q.C.; Li, Y.F.; Huang, C.Z.; Zhan, L.; Wang, D.M.; Zhen, S.J. A centrifugal microfluidic chip for point-of-care testing of staphylococcal enterotoxin B in complex matrices. *Nanoscale* **2022**, *14*, 1380–1385. [\[CrossRef\]](#)
20. Crites, B.; Kong, K.; Brisk, P. Diagonal Component Expansion for Flow-Layer Placement of Flow-Based Microfluidic Biochips. *ACM Trans. Embed. Comput. Syst.* **2017**, *16*, 1–18. [\[CrossRef\]](#)
21. Cho, S.; Lee, S.; Ahn, S.I. Design and engineering of organ-on-a-chip. *Biomed. Eng. Lett.* **2023**, *13*, 97–109. [\[CrossRef\]](#)
22. Zhao, Q.; Yuan, D.; Zhang, J.; Li, W. A Review of Secondary Flow in Inertial Microfluidics. *Micromachines* **2020**, *11*, 461. [\[CrossRef\]](#)
23. Zheng, W.; Shi, J.; Wang, A.; Fu, P.; Jiang, H. A Routing-Based Repair Method for Digital Microfluidic Biochips Based on an Improved Dijkstra and Improved Particle Swarm Optimization Algorithm. *Micromachines* **2020**, *11*, 1052. [\[CrossRef\]](#)
24. Shi, J.; Fu, P.; Zheng, W. A design method based on Bayesian decision for routing-based digital microfluidic biochips. *Analyst* **2022**, *147*, 1076–1085. [\[CrossRef\]](#)
25. Liu, W.; Ren, Y.; Tao, Y.; Yao, B.; Liu, N.; Wu, Q. A universal design of field-effect-tunable microfluidic ion diode based on a gating cation-exchange nanoporous membrane. *Phys. Fluids* **2017**, *29*, 112001. [\[CrossRef\]](#)
26. Liu, Z.; Liu, X.; Wang, Y.; Yang, D.; Li, C. Ion current rectification in asymmetric charged bilayer nanochannels. *Electrochim. Acta* **2022**, *403*, 139706. [\[CrossRef\]](#)
27. Paloschi, V.; Sabater-Lleal, M.; Middelkamp, H.; Vivas, A.; Johansson, S.; van der Meer, A.; Tenje, M.; Maegdefessel, L. Organ-on-a-chip technology: A novel approach to investigate cardiovascular diseases. *Cardiovasc. Res.* **2021**, *117*, 2742–2754. [\[CrossRef\]](#)
28. Ahmed, F.; Shimizu, M.; Wang, J.; Sakai, K.; Kiwa, T. Optimization of Microchannels and Application of Basic Activation Functions of Deep Neural Network for Accuracy Analysis of Microfluidic Parameter Data. *Micromachines* **2022**, *13*, 1352. [\[CrossRef\]](#)
29. Zhang, T.; Bai, H.; Zhao, Y.; Ren, B.; Wen, T.; Chen, L.; Song, S.; Komarneni, S. Precise Cation Recognition in Two-Dimensional Nanofluidic Channels of Clay Membranes Imparted from Intrinsic Selectivity of Clays. *ACS Nano* **2022**, *16*, 4930–4939. [\[CrossRef\]](#)
30. Wang, Q.; Ru, Y.; Yao, H.; Ho, T.Y.; Cai, Y. Sequence-pair-based placement and routing for flow-based microfluidic biochips. In Proceedings of the Asia & South Pacific Design Automation Conference, Macao, China, 25–28 January 2016; pp. 587–592.
31. Chantar, H.; Tubishat, M.; Essgaer, M.; Mirjalili, S. Hybrid Binary Dragonfly Algorithm with Simulated Annealing for Feature Selection. *SN Comput. Sci.* **2021**, *2*, 295–306. [\[CrossRef\]](#)
32. Pei, J.; Xu, L.; Huang, Y.; Jiao, Q.; Yang, M.; Ma, D.; Jiang, S.; Li, H.; Li, Y.; Liu, S.; et al. A Two-Step Simulated Annealing Algorithm for Spectral Data Feature Extraction. *Sensors* **2023**, *23*, 893. [\[CrossRef\]](#)
33. Benítez-Hidalgo, A.; Nebro, A.J.; Aldana-Montes, J.F. Sequoya: Multiobjective multiple sequence alignment in Python. *Bioinformatics* **2020**, *36*, 3892–3893. [\[CrossRef\]](#)
34. Meinecke, C.R.; Heldt, G.; Blaudeck, T.; Lindberg, F.W.; van Delft, F.; Rahman, M.A.; Salhotra, A.; Månsson, A.; Linke, H.; Korten, T.; et al. Nanolithographic Fabrication Technologies for Network-Based Biocomputation Devices. *Materials* **2023**, *16*, 1046. [\[CrossRef\]](#)

35. Vlaic, S.; Conrad, T.; Tokarski-Schnelle, C.; Gustafsson, M.; Dahmen, U.; Guthke, R.; Schuster, S. ModuleDiscoverer: Identification of regulatory modules in protein-protein interaction networks. *Sci. Rep.* **2018**, *8*, 433–444. [\[CrossRef\]](#)
36. Hadji Rasouliha, S.; Bauer, A.; Dettwiler, M.; Welle, M.M.; Leeb, T. A frameshift variant in the EDA gene in Dachshunds with X-linked hypohidrotic ectodermal dysplasia. *Anim. Genet.* **2018**, *49*, 651–654. [\[CrossRef\]](#)
37. Capuzzello, G.; Jacinto, J.G.P.; Häfliger, I.M.; Chapman, G.E.; Martin, S.S.; Viora, L.; Jonsson, N.N.; Drögemüller, C. A large deletion encompassing exon 2 of the ectodysplasin A (EDA) gene in a British blue crossbred calf with hypohidrotic ectodermal dysplasia. *Acta Vet. Scand.* **2022**, *64*, 23–31. [\[CrossRef\]](#)
38. Valverde, M.G.; Mille, L.S.; Figler, K.P.; Cervantes, E.; Li, V.Y.; Bonventre, J.V.; Masereeuw, R.; Zhang, Y.S. Biomimetic models of the glomerulus. *Nat. Rev. Nephrol.* **2022**, *18*, 241–257. [\[CrossRef\]](#)
39. Zhang, J.; Tavakoli, H.; Ma, L.; Li, X.; Han, L.; Li, X. Immunotherapy discovery on tumor organoid-on-a-chip platforms that recapitulate the tumor microenvironment. *Adv. Drug Deliv. Rev.* **2022**, *187*, 114365. [\[CrossRef\]](#)
40. Zhou, B.; Ho, S.S.; Zhang, X.; Pattni, R.; Haraksingh, R.R.; Urban, A.E. Whole-genome sequencing analysis of CNV using low-coverage and paired-end strategies is efficient and outperforms array-based CNV analysis. *J. Med. Genet.* **2018**, *55*, 735–743. [\[CrossRef\]](#)
41. Sheffler, W.; Yang, E.C.; Dowling, Q.; Hsia, Y.; Fries, C.N.; Stanislaw, J.; Langowski, M.D.; Brandys, M.; Li, Z.; Skotheim, R.; et al. Fast and versatile sequence-independent protein docking for nanomaterials design using RPXDock. *PLoS Comput. Biol.* **2023**, *19*, e1010680. [\[CrossRef\]](#)
42. Jia, L.; Jain, M.; Sun, Y. Improving antibody thermostability based on statistical analysis of sequence and structural consensus data. *Antib. Ther.* **2022**, *5*, 202–210. [\[CrossRef\]](#)
43. Pakela, J.M.; Tseng, H.H.; Matuszak, M.M.; Ten Haken, R.K.; McShan, D.L.; El Naqa, I. Quantum-inspired algorithm for radiotherapy planning optimization. *Med. Phys.* **2020**, *47*, 5–18. [\[CrossRef\]](#) [\[PubMed\]](#)
44. Dang, F.L.; Wu, C.X.; Wu, Y.; Li, R.; Zhang, S.; Jiaying, H.; Liu, Z.G. Cost-based multi-parameter logistics routing path optimization algorithm. *Math. Biosci. Eng. MBE* **2019**, *16*, 6975–6989. [\[CrossRef\]](#) [\[PubMed\]](#)
45. Leber, A.; Dong, C.; Laperrousaz, S.; Banerjee, H.; Abdelaziz, M.; Bartolomei, N.; Schyrr, B.; Temelkuran, B.; Sorin, F. Highly Integrated Multi-Material Fibers for Soft Robotics. *Adv. Sci.* **2023**, *10*, e2204016. [\[CrossRef\]](#)
46. Zeng, W.; Guo, L.; Xu, S.; Chen, J.; Zhou, J. High-Throughput Screening Technology in Industrial Biotechnology. *Trends Biotechnol.* **2020**, *38*, 888–906. [\[CrossRef\]](#)
47. Zabihhesari, A.; Hilliker, A.J.; Rezai, P. Fly-on-a-Chip: Microfluidics for Drosophila melanogaster Studies. *Integr. Biol. Quant. Biosci. Nano Macro* **2019**, *11*, 425–443. [\[CrossRef\]](#)
48. Cheng, C.; Foxworthy, G.; Fridman, G. On-chip ionic current sensor. *Appl. Phys. A* **2021**, *127*, 314–324. [\[CrossRef\]](#)
49. Wei, C.; Zong, Y.; Jiang, Y. Bioinspired Wire-on-Pillar Magneto-Responsive Superhydrophobic Arrays. *ACS Appl. Mater. Interfaces* **2023**, *15*, 24989–24998. [\[CrossRef\]](#)
50. Ducr e, J. Systematic review of centrifugal valving based on digital twin modeling towards highly integrated lab-on-a-disc systems. *Microsyst. Nanoeng.* **2021**, *7*, 104–130. [\[CrossRef\]](#) [\[PubMed\]](#)
51. Sanka, R.; Lippai, J.; Samarasekera, D.; Nemsick, S.; Densmore, D. 3D μ F—Interactive Design Environment for Continuous Flow Microfluidic Devices. *Sci. Rep.* **2019**, *9*, 9166–9176. [\[CrossRef\]](#)
52. Honrado, C.; McGrath, J.S.; Reale, R.; Bisegna, P.; Swami, N.S.; Caselli, F. A neural network approach for real-time particle/cell characterization in microfluidic impedance cytometry. *Anal. Bioanal. Chem.* **2020**, *412*, 3835–3845. [\[CrossRef\]](#) [\[PubMed\]](#)
53. Ross, G.A.; Rustenburg, A.S.; Grinaway, P.B.; Fass, J.; Chodera, J.D. Biomolecular Simulations under Realistic Macroscopic Salt Conditions. *J. Phys. Chem. B* **2018**, *122*, 5466–5486. [\[CrossRef\]](#) [\[PubMed\]](#)
54. Song, C.; Du, Q.; Yang, S.; Feng, H.; Pang, H.; Li, C. Flexible joint parameters identification method based on improved tracking differentiator and adaptive differential evolution. *Rev. Sci. Instrum.* **2022**, *93*, 084706. [\[CrossRef\]](#) [\[PubMed\]](#)
55. Suh, D.; Radak, B.K.; Chipot, C.; Roux, B. Enhanced configurational sampling with hybrid non-equilibrium molecular dynamics-Monte Carlo propagator. *J. Chem. Phys.* **2018**, *148*, 14101–14110. [\[CrossRef\]](#) [\[PubMed\]](#)
56. Wang, Q.; Zou, H.; Yao, H.; Ho, T.Y.; Wille, R.; Cai, Y. Physical Co-Design of Flow and Control Layers for Flow-Based Microfluidic Biochips. *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.* **2018**, *37*, 1157–1170. [\[CrossRef\]](#)

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.