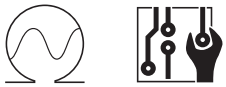
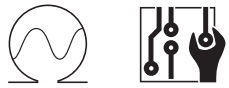


Technische Universität München
Department of Electrical Engineering and Information Technology
Institute for Electronic Design Automation

High-level Synthesis for Continuous-flow Microfluidics

Bachelor Thesis

Mengchu Li



Technische Universität München
Department of Electrical Engineering and Information Technology
Institute for Electronic Design Automation

High-level Synthesis for Continuous-flow Microfluidics

Bachelor Thesis

Mengchu Li

Supervisor :	M.Sc. Tsun-Ming Tseng
Supervising Professor :	Prof. Dr. Ulf Schlichtmann
Topic issued :	09.10.2015
Date of submission :	01.02.2016

Mengchu Li
Matrikelnr.: 10854691
Ludwig-Maximilians Universität München
Informatik plus Mathematik

Abstract

The rapid development of continuous-flow microfluidics produces an increasing demand for design automation. In previous design automation work, biochemical operations are classified into specific types, and the interactions between operations and devices are oversimplified as a one-to-one type-matching process. This simplification cannot support all kinds of operations, and also overlooks important characteristics of individual operations, such as *exclusive* execution and *indeterminate* execution, which may result in unrealistic designs. In this work, we briefly review important microfluidic components, some of which have not been discussed in previous work. And we analyse the interactions between operations and devices further, based on which we propose a *general device* concept that removes the fence between devices, and introduce a *component-oriented* operation definition, which enables our modelling method to synthesize scheduling and binding solutions from complex bioassay protocols which cannot be supported by previous work.

Acknowledgements

I would like to thank Mr. Tsun-Ming Tseng and Mr. Ulf Schlichtmann for supervising my Bachelor thesis.

The development of microfluidic biochips is one of the most amazing achievement in this new century. Currently, microfluidics are mainly designed manually, which is time-consuming and lacks of practicability for very-large-scale design. Biologists spend a lot of time doing work that should be substituted by computers. This work aims to propose a high-level synthesis method which produces scheduling and binding results for complex assay protocols. I would feel very honourable if this work could contribute to the design automation for microfluidics.

Contents

List of Figures

List of Tables

1. Introduction

The advent of continuous-flow microfluidics contributes greatly to the miniaturization, integration, automation and parallelization of biochemical assays, since it offers benefits in numerous aspects including high throughput, rapid results, better reproducibility, accurate volume control, and cost saving. With the rapid development of lab-on-a-chip technology, continuous-flow microfluidics provide a platform for ever more complex assays consisting of different delicate operations, which involves lots of design efforts and thus results in the increasing demand for design automation.

Most continuous-flow microfluidics comprise an easily combinable set of devices, which enables sophisticated bioassays to be performed in a single chip within mature fabrication technology (Mark et al. 2010). The first automatic synthesis work (Amin et al. 2007) therefore proposes a fluidic instruction set, where each device is dedicated to a specified type of operations. For example, a mixer is supposed to be a device only for mixing operations, and a mixing operation is supposed to be bound only to a mixer. This concept is inherited and strengthened by later research (Minhass et al. 2011) (Minhass et al. 2012) (Tseng et al. 2013) (Tseng et al. 2015) and has become the accepted standard. However, as fabrication technology evolves and bioassay protocol innovates, ever more assays include operations that do not fit into this assumption.

Figure 1.1 illustrates the protocol of a gene expression profiling assay from (Zhong et al. 2008), in which mRNA is extracted from single human embryonic stem cells (hSEC) and then converted to cDNA for the measurement of gene expression. We take this assay as an example to introduce some devices and operations, which cannot easily be defined as any conventional types.

As shown in Figure 1.1, after a few preparation steps, single hSEC is captured from a single-cell supspension by a cell-trap module integrated in a $10nL$ ring (o_5), and then observed by microscope to ensure that only modules with one cell will be used for further experimentation (o_6). The captured cell is then lysed with the lysis buffer in this ring by executing a peristaltic

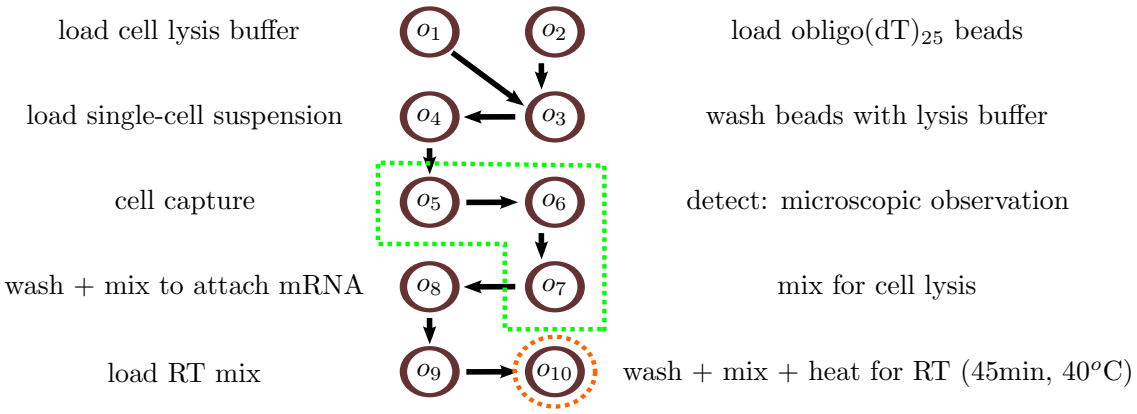


Figure 1.1.: Protocol of a gene expression profiling assay.

pump sequence with control channels (o_7). It is remarkable that instead of being distributed to three different devices which are dedicated to specified operation types, the cell-capturing (o_5), detecting (o_6), and mixing (o_7) operations are executed in the same ring integrated with a cell-trap module and a peristaltic pump, which we call a *multi-functional device*.

Moreover, in the reverse transcription (RT) process, after mRNAs are captured by a oligo $(dT)_{25}$ bead column stacked against a sieve valve, they will be mixed with RT master mix, while the chip is heated to 40°C (o_{10}). This RT operation is a combination of washing, mixing and heating operations, and thus cannot be specified into a pre-defined operation type. We call this kind of operation *compound operation*.

Both multi-functional devices and compound operations are beyond the capability of existing binding methods. And there are also some commonly seen characteristics of biochemical operations, which can barely be supported by existing scheduling methods. These characteristics include *exclusive* execution, and *indeterminate* execution.

Some operations require exclusive execution, since they need to be executed under special conditions that may bring about side-effect to other operations. For example, in the assay protocol from (Zhong et al. 2008), the above mentioned RT operation is executed by heating the whole chip to 40°C on a thermal microscope stage, which means that other temperature-sensitive operations cannot be performed in parallel in the same chip, and other temperature-sensitive reagents must be removed from the chip before the RT process.

The execution duration of operations are sometimes indeterminate. For example, in single-cell capturing operations, the chance that a cell trap captures exactly one cell is about 53% (Carlo

et al. 2006). Therefore, most of the time it is necessary to check the number of cells. In (Marcy et al. 2007), cells can be detected by fluorescent signals. When a signal comes, an image will be taken and analyzed to count the number of cells. If the number is not equal to one, this cell capturing operation needs to be rerun. Therefore, the exact duration of this operation cannot be confirmed until its completion. We call operations with indeterminate execution duration *indeterminate operations*.

2. Background and Formulation

2.1. Microfluidic Components

In order to propose a binding and scheduling method for complex assays involving multi-functional devices and operations with different characteristics, we start with a brief review of important microfluidic components, some of which have never been discussed in previous design automation work. Based on the area cost and manufacturing cost of integrating these components in a chip, we classify them into three categories: containers, accessories, and off-chip instruments.

2.1.1. Container

Containers are microfluidic components, the integration of which require both manufacturing costs and exclusive chip areas.

Chamber is a segment of a flow channel separated by two valves. Chambers can vary in length and width according to different operation protocols. Diverse operations can be performed in chambers, such as mixing (White et al. 2011), amplification (Wang et al. 2012), heating (Zhong et al. 2008), neutralization (Marcy et al. 2007), and cell culturing (Gomez-Sjoeberg et al. 2007).

Ring is a specialized chamber which is connected end to end and thus enables circulation flow. It is mainly used to perform highly efficient mixing operations.

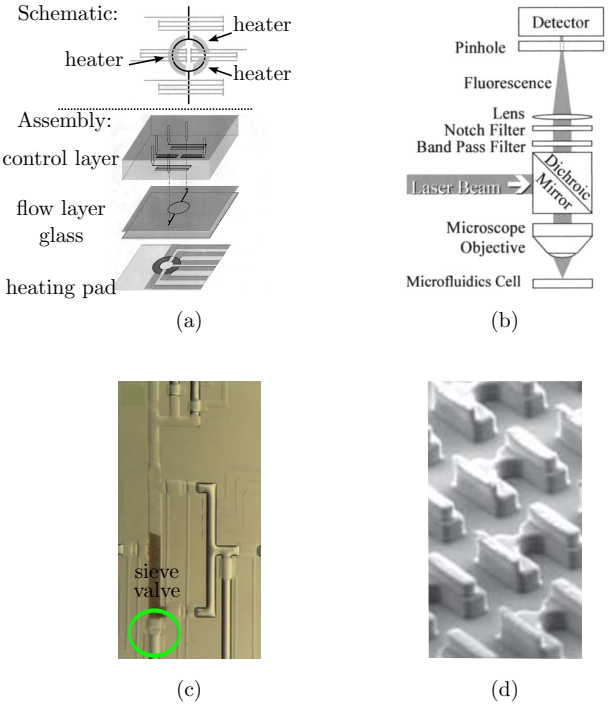


Figure 2.1.: (a) Heating pad. (b) Optical system. (c) Sieve valve. (d) Cell trap.

2.1.2. Accessory

Accessories are microfluidic components with functional specialization. They can be integrated into containers and thus requiring no area cost. However, the integration of accessories involves additional control efforts, such as chip ports and control channels, thus requires extra manufacturing cost.

Pump is a group of valves providing pressure for fluid movement. Each valve can be assigned to an individual pressure source or sequentially connected with other valves driven by the same pressure source.

Heating pad consists of a heating layer and a heating circuit, and has not been discussed in depth in previous work. A heating pad is usually integrated under the flow layer. Figure. 2.1(a) (Liu et al. 2002) shows the schematic and the assembly of a rotary device integrated with a heating pad, the heating circuit of which is divided into three independent parts (denoted as heaters), thus enabling independent heating operations requiring different temperatures to be executed in parallel.

Optical system is a general term that refers to detection components consisting of a light source and a receiver (detector). The light source may be weak as a simple LED (Adams et al. 2002) or strong as a laser beam (Filippova et al. 2003) as shown in Figure. 2.1(b); and the receiver may be an imager under the chip (Adams et al. 2002) or a camera hanging above the chip (Filippova et al. 2003).

Sieve valve is a specialized valve as shown in Figure. 2.1(c) (Lee et al. 2005), which leaves a gap when it is closed. It is mentioned in (Li et al. 2016) for the first time in the design automation field. A closed sieve valve can halt large particles while allowing small particles and fluids to flow, thus enabling operations that increase sample concentration by forming solid-phase support (Zhong et al. 2008), which are called *washing* operations in bioassay protocols, and should be distinguished from rinsing operations for cleaning channels or devices.

Cell trap is a passive microfluidic component used to capture a single cell. It has not been mentioned in previous work. Cell traps vary in shapes and sizes: some U-shaped PDMS traps are shown in Figure. 2.1(d) (Gupta et al. 2010). Single-cell assays are one of the most important categories of microfluidic applications, and contribute strongly for understanding the stochastic variation of gene expression.

2.1.3. Off-chip Instrument

Off-chip instruments are external apparatuses which are usually driven by custom software. They are not integrated in chips and thus demand neither chip area nor chip manufacturing cost.

Thermocycler is a laboratory instrument for executing heating operations. The chip inserted in a thermocycler will be heated as a whole by the thermal block.

Microscope is an instrument for observing small objects, which is involved in microfluidic assays to track assay process.

2.2. General Device and Component-oriented Operation Definition

With the above categories of microfluidic components, instead of building fences between devices to distribute them to dedicated types, we formulate a *general device* concept to synthesize devices that are adaptable for a variety type of operations. A general device is a general platform for all kinds of operations. It consists of a container and a variety number of accessories and off-chip instruments, which can be adjusted according to different bioassay protocols, thus providing high flexibility for design automation.

A general device can be a conventional specialized device, or a multi-functional device. For example, a conventional rotary mixer is a general device with a ring as its container and a pump as its accessory; and the multi-functional device mentioned in Section 1 for three sequential operations is a general device with a ring as its container, a cell-trap and a pump as its accessories, and a microscope as its off-chip instrument. Similarly, compound operations can easily be bound to general devices with corresponding settings. For example, the above mentioned RT operation can be bound to a general device with a chamber as its container, a sieve valve as its accessory, and an off-chip thermocycler.

Under this general device concept, instead of classifying biochemical operations into different types, we introduce a *component-oriented* definition method to accurately describe the characteristics of operations.

A component-oriented operation definition shall include following attributes:

- required container (with specified capacity), accessories, off-chip instruments;
- execution duration, which can be an accurate value, or be specified as indeterminate with a minimum duration;
- dependency relationship: parent-child specification (if an operation receives the outputs of other operations as its inputs, corresponding operations need to be specified);
- sensitivity and influences;
- execution limitations.

2.3. Problem Formulation

Therefore, the high-level-synthesis problem that we are dealing with can be formulated as follows:

Input :

a bioassay protocol consisting of component-oriented operation definitions.

Output :

a synthesis result indicating scheduling and binding solutions, considering assay execution time, chip area and manufacturing cost.

3. Synthesis Using the General Device Concept

We build an integer-linear-programming (ILP) model to synthesize binding and scheduling solutions from bioassay protocols, all operations thereof are specified with component-oriented definitions. In this model, we have a set D of general devices, the cardinality of which can be given by the user, and represents the number of available devices.

3.1. General Device Configuration

According to our concept, each general device consists of exactly one container, which can be a ring or a chamber. We introduce two binary variables $d_{j,r}$ and $d_{j,ch}$ to indicate the container type of a device d_j, and formulate their relation as:

$$\forall d_j \in D, \quad d_{j,r} + d_{j,ch} = 1; \tag{3.1}$$

Correspondingly, the binary variables indicating the existence of accessories and off-chip instruments are also represented with their initials as: $d_{j,p}$, $d_{j,h}$, $d_{j,o}$, $d_{j,s}$, $d_{j,c}$, $d_{j,t}$ and $d_{j,m}$.

To support operations with different reagent volumes, we define containers with four different capacity: $large$, $medium$, $small$ and $tiny$, which can be represented by binary variables $d_{j,cap,l}$, $d_{j,cap,m}$, $d_{j,cap,s}$ and $d_{j,cap,t}$. We introduce the following constraint to ensure that a device d_j has exactly one capacity:

$$\forall d_j \in D, \quad d_{j,cap,l} + d_{j,cap,m} + d_{j,cap,s} + d_{j,cap,t} = 1. \tag{3.2}$$

Since the capacity of a ring is usually larger than the capacity of a chamber, we define that the capacity of a ring may vary among large, medium and small, and the capacity of a chamber

may vary among medium, small and tiny, which can be formulated as:

$$\forall d_j \in D, \quad d_{j,cap,l} + d_{j,cap,m} + d_{j,cap,s} \geq d_{j,r}, \tag{3.3}$$

$$d_{j,cap,m} + d_{j,cap,s} + d_{j,cap,t} \geq d_{j,ch}. \tag{3.4}$$

If a device d_j has a ring as its container, $d_{j,r}$ will be set to 1, and constraint (3.2) and (3.3) ensures that exactly one of the elements in $\{d_{j,cap,l}, d_{j,cap,m}, d_{j_{cap},s}\}$ will be set to 1, too. Analogously, if d_j has a chamber as its container, exactly one of the elements $\{d_{j,cap,m}, d_{j,cap,s}, d_{j_{cap},t}\}$ must be set to 1 correspondingly.

3.2. Operation Configuration

With the component-oriented definitions from bioassay protocols, the scheduling and binding relations among operations and devices can be modelled as follows.

3.2.1. Component Consistence

An operation must be bound to exactly one general device that fulfills the component requirements specified in the operation definitions. To model the binding relations among operations and devices, we introduce a binary variable $o\_d_{i,j}$ for each operation o_i and device d_j to represent whether o_i is bound to d_j, and we introduce the following constraint to ensure that o_i is bound to exactly one device:

$$\sum_{d_j \in D} o\_d_{i,j} = 1, \tag{3.5}$$

We then represent the requirements of an operation o_i with binary constants analogous to the device configurations: $o_{i,r}$, $o_{i,ch}$, $o_{i,p}$, $o_{i,h}$, $o_{i,o}$, $o_{i,s}$, $o_{i,c}$, $o_{i,t}$ and $o_{i,m}$ indicate whether the execution of o_i requires corresponding components; and $o_{i,cap,l}$, $o_{i,cap,m}$, $o_{i,cap,s}$ and $o_{i,cap,t}$ indicate the volume of reagents. For example, if an operation o_i with a large reagent volume requires a ring as the container and a pump as an accessory for its execution, $o_{i,cap,l}$, $o_{i,r}$ and

$o_{i,p}$ are set to 1, and the corresponding constraints can be formulated as:

$$\forall o_i \in O, d_j \in D, \quad d_{j,r} - o\_d_{i,j} + 1 \geq o_{i,r}, \tag{3.6}$$

$$d_{j,p} - o\_d_{i,j} + 1 \geq o_{i,p}, \tag{3.7}$$

$$d_{j,cap,l} - o\_d_{i,j} + 1 \geq o_{i,cap,l}, \tag{3.8}$$

where O is the set of all operations. If o_i is not bound to d_j ($o\_d_{i,j} = 0$), above constraints become tautology. If o_i is bound to d_j, above constraints can be regarded as the following:

$$d_{j,r} \geq o_{i,r}, \tag{3.9}$$

$$d_{j,p} \geq o_{i,p}, \tag{3.10}$$

$$d_{j,cap,l} \geq o_{i,cap,l}, \tag{3.11}$$

which ensure that d_j has a container of required capacity, and integrated with required accessories.

3.2.2. Execution Duration

In our model, the duration of a determinate operation o_i is represented as $o_{i,dur}$. If o_i is indeterminate, its duration is represent as $o_{i,dur} + o_{i,ind}$, where $o_{i,dur}$ represents its minimum duration, and $o_{i,ind}$ represents its rest duration. Assays including indeterminate operations will be discussed in Section 3.2.6.

The execution time of an operation o_i can therefore be specified by introducing a variable $o_{i,st}$ that represents the start time of o_i, since the completion time of o_i can be calculated as $o_{i,st} + o_{i,dur}$. If the execution times of two operations o_a and o_b overlap each other, o_a and o_b cannot be bound to the same device, since a device cannot support multiple operations simultaneously. This can be formulated as follows:

$$o_{a,st} + q_0 \cdot M \geq o_{b,st} + o_{b,dur} + t, \tag{3.12}$$

$$o_{a,st} + o_{a,dur} + t - q_1 \cdot M \leq o_{b,st}, \tag{3.13}$$

$$\forall d_j \in D, \quad o\_d_{a,j} + o\_d_{b,j} - q_2 \leq 1, \tag{3.14}$$

$$q_0 + q_1 + q_2 \leq 2, \tag{3.15}$$

where t is a constant representing the transportation time, M is an extremely large auxiliary

constant, $\{q_0, q_1, q_2\}$ are binary variables, one of which has to be set to 0 according to (3.15). Therefore, if o_a and o_b are bound to the same device d_j ($o\_d_{a,j} = 1$ and $o\_d_{b,j} = 1$), it follows that $q_2 = 1$, and thus $q_0 = 0$ or $q_1 = 0$. Therefore, the above constraint can be transformed as the following:

$$o_{a,st} \geq o_{b,st} + o_{b,dur} + t, \tag{3.16}$$

or

$$o_{a,st} + o_{a,dur} + t \leq o_{b,st}, \tag{3.17}$$

which means that o_a either starts after the completion of o_b, or ends before the execution of o_b.

3.2.3. Dependency Relationship

Operation-dependency indicates the inheritance of inputs between sequential operations. If the output of an operation o_a is inherited by another operation o_b as its input, then o_a is called the parent operation of o_b, and o_b is called the child operation of o_a. Since an operation can only start after collecting all the needed inputs, a child operation can only start after the completion of its parent operations. This dependency relationship can be formulated as follows:

if (o_p *is the parent operation of* o_c), *then*:

$$o_{c,st} \geq o_{p,st} + o_{p,dur} + t, \tag{3.18}$$

where $o_{p,st} + o_{p,dur} + t$ indicates the earliest timing that the output of o_p can be ready for the execution of o_c.

3.2.4. Sensitivity and Influences

The execution of some operations may have side effect on other operations. For example, operations involving thermocyclers have an influence on the temperature of the whole chip, which means that they should be executed exclusively without other temperature-sensitive operations.

We use a pair of binary constants $(o_{i,sens,x}, o_{i',infl,x})$ in our component-oriented operation definition, to represent whether o_i and o_i' have interacting sensitivity and influence attribute x, where x indicates the type of the interacting features. For example, $o_{i,sens,h} = 1$ and $o_{i',infl,h} = 1$ indicate that o_i has an influence on temperature-sensitive operations, and $o_{i'}$ is a temperature-sensitive operation. We introduce the following constraints to ensure that the execution time of o_i and o_i' do not overlap each other:

$$o_{i,st} + q_0 \cdot M \geq o_{i',st} + o_{i',dur}, \tag{3.19}$$

$$o_{i,st} + o_{i,dur} \leq o_{i',st} + q_1 \cdot M, \tag{3.20}$$

$$q_0 + q_1 = 1, \tag{3.21}$$

where $\{q_0, q_1\}$ are auxiliary binary variables, one of which will be set to 0 and the other will be set to 1. Since M is an extremely large auxiliary constant, if q_0 is set to 1, (3.19) becomes tautology; correspondingly, if q_1 is set to 1, (3.20) becomes tautology. Therefore, the above constraints will be transformed to either

$$o_{i,st} + o_{i,dur} \leq o_{i',st}, \tag{3.22}$$

or

$$o_{i,st} \geq o_{i',st} + o_{i',dur}, \tag{3.23}$$

thus ensuring that either o_i ends before the execution of o_i', or o_i starts after the completion of o_i'.

3.2.5. Execution Limitations

Under limitation of different features of biochemical reagents and different experimental objectives, some operations must be executed under particular time and space constraints.

For example, in cDNA synthesis assays, since mRNA is very susceptible to degradation by widely existing Ribonuclease (RNase), cDNA first strand synthesis is supposed to be executed on the same device immediately after mRNA capture (Zhong et al. 2008). In our model, for operation o_a and its child operation o_b requiring immediately sequential execution, we introduce

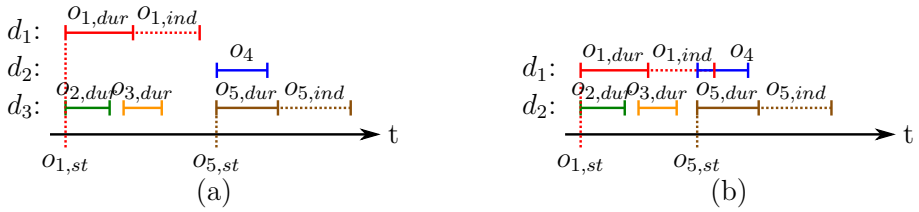


Figure 3.1.: Possible scheduling and binding solutions for an assay including indeterminate operations. (a) waste of devices. (b) conflict of devices.

the following constraints:

$$o_{b,st} \leq o_{a,st} + o_{a,dur} + t, \tag{3.24}$$

$$\forall d_j \in D, \quad o\_d_{a,j} = o\_d_{b,j}, \tag{3.25}$$

thereof (3.24) ensures that o_b will be executed immediately after the completion of o_a, and (3.25) ensures that o_a and o_b will be executed on the same device.

In many assay protocols, particular operations are executed in parallel with several duplicates for comparison. We introduce the following constraints for such operations to ensure that they start simultaneously:

if o_a and o_b require to be executed in parallel:

$$o_{a,st} = o_{b,st}. \tag{3.26}$$

3.2.6. Indeterminate Execution Duration

If o_a is indeterminate and bound to a device d_j, it is unpredictable, when d_j would be available again for the execution of another operation. This indetermination leads to either the waste or the conflict of devices, since d_j is either to be prevented from executing any posterior operation, or bound by a posterior operation arbitrarily without the guarantee of non-overlaping execution time.

Figure 3.1 shows the scheduling and binding results for an assay consisting of five operations, thereof o_1 and o_5 are indeterminate operations, o_1 is the parent operation of o_4 and o_5, o_2 is the parent operation of o_3, and all these five operations have the same component requirements.

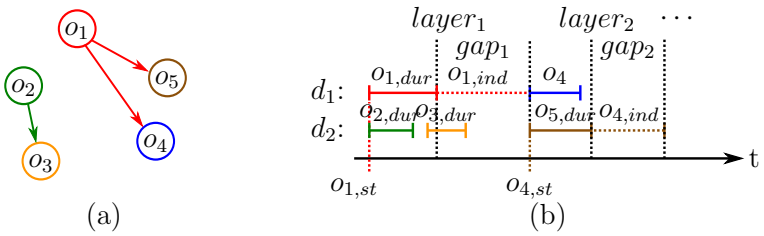


Figure 3.2.: (a) dependency graph. (b) synthesis with layer distribution

As shown in Figure 3.1(a), when o_1 is bound to d_1, if we prevent d_1 from being bound by any other posterior operation, o_4 might be bound to a new device d_2, even though d_1 is already available after the completion of o_1.

However, as shown in Figure 3.1(b), if we arbitrarily assume the execution duration of o_1 as a precise value, d_1 might be bound to o_4 even though it is still occupied by o_1, which results in an unrealistic design.

To avoid the waste and conflict of devices, if an assay includes indeterminate operations, we classify all operations in this assay into n indexed operation layers, so that each layer (except for the n-th layer) contains at least one indeterminate operation. For two layers L_a and L_b with $a < b$, we call L_a the predecessor of L_b, and operations in L_b are allowed for execution only after the completion of operations in all L_b's predecessors. In this manner, when dealing with operations in a new layer, all devices are available without execution time conflicts, and the binding problems for different layers can thus be solved independently.

For each layer L_i that includes indeterminate operations, the execution duration of an indeterminate operation o_a is regarded as $o_{a,dur}$ in the scheduling and binding process, and we introduce the following constraint to avoid potential conflicts:

$$\forall o_b \in L_i, \quad o_{b,st} \leq o_{a,st} + o_{a,dur}, \tag{3.27}$$

which means that no other operations in L_i can start after $o_{a,st} + o_{a,dur}$, thus no other operations would be bound to the device occupied by o_a in the indeterminate time interval $(o_{a,st} + o_{a,dur}, o_{a,st} + o_{a,dur} + o_{a,ind})$.

Since the scheduling result of an operation will be influenced by its predecessors, we maximize the number of operations in each layer L_i in ascending order by implementing a modified maximal independent set algorithm:

We build a graph $G = (V, E)$ for L_i, thereof V is the set of vertices representing all operations that have not been classified into any layers, and E is the set of directed edges meeting following conditions:

– if o_a is the parent operation of o_b, then there is an edge from o_a to o_b;

– if o_a and o_b are required to be executed in parallel, then there is a bidirectional edge from o_a to o_b as well as from o_b to o_a.

We initialize a set S to represent the set of operations that can be classified into L_i. If there is at least one indeterminate operation o_a in graph G, and o_a cannot be reached from any other indeterminate operations, we add o_a to S and remove o_a and all the other vertices that are reachable from o_a from G. Then we repeat the above steps until there is no indeterminate operation in G, and add all the remaining operations in G to S, which indicates the maximal set of operations in L_i.

For example, as shown in Figure. 3.2(a), o_1 and o_5 are the only two indeterminate operations in this assay. Since o_1 is not reachable from o_5, we add o_1 to S and remove o_4 and o_5 from G. After that, since G no longer contains any other indeterminate operations, we add the remaining operations o_2 and o_3 to S, and obtain the maximal set of operations in L_1 as $\{o_1, o_2, o_3\}$. Then we repeat the above steps and obtain the operation set of L_2 as $\{o_4, o_5\}$.

We then perform scheduling and binding for operations in L_1 and L_2 independently. As shown in Figure. 3.2(b), there is a gap with indeterminate length between L_1 and L_2, which indicates the completion time of all operations in L_1. The accurate length of the gap would be decided by the last completed operation in L_1 during the assay process, and can therefore either be $o_{1,ind}$ or $o_{3,st} + o_{3,dur} - (o_{1,st} + o_{1,dur})$. Since we can ensure that d_1 is available again at the end of the gap, o_4 can be bound to d_1 without any conflict concern.

3.3. Objective Configuration

Our scheduling and binding results take assay execution time, chip area and manufacturing cost of microfluidic components into consideration, which are represented as three variables sum_t, sum_a and sum_m respectively. For assays consisting of indeterminate operations, each layer is regarded as an individual assay. If L_i has at least one predecessor, the devices that are

once bound by operations in L_i's predecessors will be inherited by L_i, so that devices can be shared among different layers.

The execution time of an assay is decided by the last completed operation in this assay, which can be formulated as following:

$$\forall o_i \in O, \quad sum_t \geq o_{i,st} + o_{i,dur}. \tag{3.28}$$

Containers of general devices require exclusive chip areas, which is decided by the type and capacity of the corresponding container. For each device d_j that has been bound by at least one operation, we decide its chip area cost according to its container type:

For d_j with a ring as its container:

$$sum_{a,r} = \sum_{d_j} A_{r,l} \cdot d_{j,cap,l} + A_{r,m} \cdot d_{j,cap,m} + A_{r,s} \cdot d_{j,cap,s};$$

For $d_{j'}$ with a chamber as its container:

$$sum_{a,ch} = \sum_{d_{j'}} A_{ch,m} \cdot d_{j',cap,m} + A_{ch,s} \cdot d_{j',cap,s} + A_{ch,t} \cdot d_{j',cap,t};$$

where $A_{r,l}$,$A_{r,m}$,$A_{r,s}$,$A_{ch,m}$,$A_{ch,s}$ and $A_{ch,t}$ are constants indicating the area cost of a ring or a chamber with different capacity. Thus, the total area cost can be formulated as:

$$sum_a = sum_{a,r} + sum_{a,ch},$$

Analogously, the manufacturing cost can be calculated as the sum of manufacturing cost of each container and accessory, thereof the manufacturing cost of containers $sum_{m,con}$ are decided by its type and capacity in a similar manner as above, and the manufacturing cost of accessories can be formulated as:

$$sum_{m,acc} = \sum_{d_j} M_p \cdot d_{j,p} + M_h \cdot d_{j,h} + M_o \cdot d_{j,o} + M_s \cdot d_{j,s} + M_c \cdot d_{j,c},$$

where M_p, M_h, M_o, M_s and M_c indicate the manufacturing cost of different accessories. Since off-chip instruments of a general device requires no manufacturing cost, the total manufacturing

cost can be formulated as:

$$sum_m = sum_{m,con} + sum_{m,acc},$$

Therefore, our model objective can be formulated as:

$$\text{Minimize:} \quad C_t \cdot sum_t + C_a \cdot sum_a + C_m \cdot sum_m,$$

where C_t, C_a and C_m are adjustable weight coefficients that can be defined by experimenters.

4. Experimental Results

We use C++ to implement our synthesis for four bioassays from (Zhong et al. 2008) (Marcy et al. 2007) (White et al. 2011) and solve our ILP model with the ILP solver Gurobi (Gurobi Optimization, Inc. n.d.) on a computer with a 2.67GHz CPU. The weight coefficients of sum_t, sum_a and sum_m are set as $1:1:1$.

Table 4.1 shows the results of our synthesis. Since all these four test cases include indeterminate operations, synthesis results for different layers are listed as well. The meaning of the abbreviations are formulated as follows:

$\#o$ **:** the number of operations.
$\#o_{in}$ **:** the number of indeterminate operations.
$\#d$ **:** the number of (general) devices.
$T_e(minute)$**:** execution time.
$\#d_{re}$**:** the number of devices, which are bound by operations in the upper layers.
$\#d_{ad}$**:** the number of devices, which are only bound by operations in current layer.
$A(mm^2)$**:** total area cost of containers.
T_r**:** program run time.

General devices are represented as $container^{\{capacity\}}_{\{accessories\}}$ $(with\ off\text{-}chip\ instruments)$. For example, ch^s_c represents a general device with a small chamber as its container, and a cell trap as its accessory.

Table 4.1.: Synthesis Results for Bioassays.

Testcase				For Each Layer					For the whole assay		
	$\#o$	$\#o_{in}$		$\#o$	$\#o_{in}$	$\#d_{re}$	$\#d_{ad}$	T_e	A	General Device	T_r
MDA	31	7	Layer L_1	10	7	\	8	651	0.84	ch^m : 7,ch^s : 7 (with th, mi)	$37.462s$
			Layer L_2	21	\	4	6	700			
cDNA	70	10	Layer L_1	30	10	\	12	97	10.02	r^s_p : 10, ch^m_c : 2, ch^m : 8, ch^s_s : 4 (with th, mi)	$1m2.670s$
			Layer L_2	40	\	2	12	228			
RTqPCR1	100	20	Layer L_1	20	20	\	20	9	3.45	ch^l_o : 13, ch^l : 6, ch^s_c : 20	$37.319s$
			Layer L_2	80	\	15	19	308			
RTqPCR2	120	20	Layer L_1	20	20	\	20	9	3.96	ch^l_o : 12, ch^l : 5, ch^m : 9, ch^s_c : 20	$29.911s$
			Layer L_2	100	\	0	26	1344			

As shown in Table 4.1, our general device concept supports test cases involving multi-functional devices, compound operations, as well as indeterminate operations:

MDA (Marcy et al. 2007) represents a multiple displacement amplification assay consisting of 31 operations, which are carried out mainly with off-chip instruments and requires no accessories. These operations are bound to proper containers for execution. cDNA (Zhong et al. 2008) and RTqPCR1 (White et al. 2011) are assays consisting of more operations with complex component requirements, which are comfortably satisfied by general devices. In these three test cases, devices that have been bound by operations in L_1 are also available for the operations in L_2, thus the waste of devices are avoided. RTqRCR2 (White et al. 2011) is a two step RTqPCR assay, which is similar to RTqPCR1, but requiring different containers and consisting of more operations. The operations in its first layer are all indeterminate cell capturing operations, which require small chambers with cell traps. Since operations in L_2 require different containers, devices cannot be shared between these operations. The synthesis results for this test case consisting of 120 operations is achieved within 30 seconds.

5. Conclusion

Design automation for continuous-flow microfluidics should base on bioassay protocols. In this work, we raised several realistic problems led by the simplification of microfluidic components and the overlook of operation characteristics, and briefly reviewed the important micfludic components. Then we formulated complex bioassay protocols as sets of component-oriented operation definitions, and proposed a general device concept, which removed the fence between devices, thus providing a new view to deal with this high-level synthesis problem.

Bibliography

Adams, M. L., DeRose, G. A., Quake, S. R. & Scherer, A. (2002): Fundamental approach for optoelectronic and microfluidic integration for miniaturizing spectroscopic devices, Proceedings of SPIE **4647**: 1–6.

Amin, A. M., Thottethodi, M., Vijaykumar, T.N., Werely, S. & Jacobson, S. C. (2007): Aquacore: a programmable architecture for microfluidics, S. 254–265.

Carlo, D. D., Aghdam, N. & Lee, L. P. (2006): Single-cell enzyme concentrations, kinetics, and inhibition analysis using high-density hydrodynamic cell isolation arrays, Anal. Chem. **78**: 4925–4930.

Filippova, E. M., Monteleone, D. C., Trunk, J. G., Sutherland, B. M., Quake, S. R. & Sutherland, J. C. (2003): Quantifying double-strand breaks and clustered damages in dna by single-molecule laser fluorescence sizing, Biophysical Journal **84**: 1281–1290.

Gomez-Sjoeberg, R., Leyrat, A. A., Pirone, D. M., Chen, C. S. & Quake, S. R. (2007): Versatile, fully automated, microfluidic cell culture system, Anal. Chem **79**: 8557–8563.

Gupta, K., Kim, D.-H., Ellison, D., Smith, C., Kundu, A., Tuan, J., Suh, K.-Y. & Levchenko, A. (2010): Lab-on-a-chip devices as an emerging platform for stem cell biology, Lab on a Chip **10**: 2019–2031.

Gurobi Optimization, Inc. (n.d.): Gurobi Optimizer Reference Manual, http://www.gurobi.com.

Lee, C.-C., Sui, G., Elizarov, A., Shu, C. J., Shin, Y.-S., Dooley, A. N., Huang, J., Daridon, A., Wyatt, P., Stout, D., Kolb, H. C., Witte, O. N., Satyamurthy, N., Heath, J. R., Phelps, M. E., Quake, S. R. & Tseng, H.-R. (2005): Multistep synthesis of a radiolabeled imaging probe using integrated microfluidics, Science **310**(5755): 1793–1796.

Li, M., Tseng, T.-M., Li, B., Ho, T.-Y. & Schlichtmann, U. (2016): Sieve-valve-aware synthesis of flow-based microfluidic biochips considering specific biological execution limitations.

Liu, J., Enzelberger, M. & Quake, S. R. (2002): A nanoliter rotary device for polymerase chain reaction, Electrophoresis **23**: 1531–1536.

Marcy, Y., Ishoey, T., Lasken, R. S., Stockwell, T. B., Walenz, B. P., Halpern, A. L., Beeson,

K. Y., Goldberg, S. M. D. & Quake, S. R. (2007): Nanoliter reactors improve multiple displacement amplification of genomes from single cells, PLoS Genet **9**(3): e155.

Mark, D., Haeberle, S., Roth, G., v. Stetten, F. & Zengerle, R. (2010): Microfluidic lab-on-a-chip platforms: requirements, characteristics and applications, Chem. Soc. Rev. **39**: 1153–1182.

Minhass, W. H., Pop, P. & Madsen, J. (2011): System-level modeling and synthesis of flow-based microfluidic biochips, Proc. Int. Conf. Compil., Arch. and Syn. Embed. Sys., S. 225–234.

Minhass, W. H., Pop, P., Madsen, J. & Blaga, F. S. (2012): Architectural synthesis of flow-based microfluidic large-scale integration biochips, Proc. Int. Conf. Compil., Arch. and Syn. Embed. Sys., S. 181–190.

Tseng, K.-H., You, S.-Chi, Liou, J.-Y. & Ho, T.-Y. (2013): A top-down synthesis methodology for flow-based microfluidic biochips considering valve-switching minimization, S. 123–129.

Tseng, T.-M., Li, B., Ho, T.-Y. & Schlichtmann, U. (2015): Reliability-aware synthesis for flow-based microfluidic biochips by dynamic-device mapping, S. 141:1–141:6.

Wang, J., Fan, H. C., Behr, B. & Quake, S. R. (2012): Genome-wide single-cell analysis of recombination activity and *de novo* mutation rates in human sperm, Cell **150**(2): 402–412.

White, A. K., VanInsberghe, M., Petriv, O. I., Hamidi, M., Sikorski, D., Marra, M. A., Piret, J., Aparicio, S. & Hansen, C. L. (2011): High-throughput microfluidic single-cell RT-qPCR, Proc. Natl. Acad. Sci. **108**(34): 13999–14004.

Zhong, J. F., Chen, Y., Marcus, J. S., Scherer, A., Quake, S. R., Taylor, C. R. & Weiner, L. P. (2008): A microfluidic processor for gene expression profiling of single human embryonic stem cells, Lab on a Chip **8**(1): 68–74.