Computer-aided design tools for labs-on-a-chips

Alexi Bonament

UNIVERSITÉ DE STRASBOURG

ÉCOLE DOCTORALE MATHEMATIQUE, SCIENCE DE L'INFORMATION ET DE L'INGENIEUR

Laboratoire des sciences de l'ingénieur, de l'informatique et de l'imagerie (ICube, UMR 7357)

THÈSE présentée par :

Alexi BONAMENT

soutenue le : **23 Avril 2021**

pour obtenir le grade de : **Docteur de l'université de Strasbourg**

Discipline/ Spécialité : Science de l'ingénieur

Computer-aided design tools for labs-on-a-chips

THÈSE dirigée par :

Mr LALLEMENT Christophe	Professeur, Université de Strasbourg

RAPPORTEURS :

Mr ERRACHID EL SALHI Abdelhamid	Professeur, Université Claude Bernard Lyon 1
Mr PECHEUX François (président)	Professeur, université Paris Sorbonne

---

AUTRES MEMBRES DU JURY :

Mr MADEC Morgan	Maitre de conférences, Université de Strasbourg
Mr RENAUD Louis	Maitre de conférences, Université Claude Bernard Lyon 1
Mr SALLESE Jean-Michel	Maître d'Enseignement et de Recherche, Ecole polytechnique Fédérale de Lausanne EPFL

Table des matières

Glossaire

DNA	Acide désoxyribonucléique
CAD	Computer-aided design
DMF	Digital MicroFluidic
IDT	InterDigital Transducer
HDL-AMS	Hardware Description Languages for Analog and Mixed Signals
ISFET	Ion-Sensitive Field-Effect transistor
Lab-on-a-chip	Miniaturized molecular analysis tool
Microfluidic	Fluid mechanics in a microscopic scale environment
MSE	Mean Square Error
SAW	Surface Acoustic Wave
PCR	Polymerase Chain Reaction
SPICE	Simulation Program with Integrated Circuit Emphasis
SNP	Single Nucleotide Polymorphism
SPR	Surface Plasmon Resonance
Verilog	Hardware description language (HDL) used to model electronic systems.

Introduction

Since the 2000's, the demand for reliable and portable biological and chemical analysis tools has increased in several areas, including:

- the health sector:

  - to diversify the number of screening tools
  - for the continuous monitoring of a patient's state of health

- in the field of the environment:

  - for monitoring water, air or soil pollution levels

- and even in the food industry:

  - for monitoring food contamination

To meet this demand, miniaturized analysis tools called "lab-on-a-chip" devices have been developed, particularly from a technical viewpoint, over the past decade. However, large-scale industrial developments of labs on chips have yet to be discovered and exploited.

The development of labs on chips follows from the development of microfluidics. The manufacturing technique developed by George WHITSIDE [1] using polydimethylsiloxane (PDMS) paved the way for the development of microfluidics in academic research. The number of scientific articles has been increasing substantially since the beginning of the 2000's, following publications about this technique allowing for the rapid prototyping of microfluidic circuits.

However, to date the PDMS-based technique has only led to the production of an efficient prototype, not to industrial development in the field of microfluidic circuits.

Moreover, most publications on microfluidics currently concern complex technical challenges taken up by microfluidics specialists [2],[3]. However, most of the circuits are designed for use by researchers in other fields like chemistry and / or biology.

The current issue stems from the fact that researchers outside the microfluidics field have to collaborate with microfluidics specialists to develop their ideas. This is an obvious obstacle to development, as if computer scientists were needed to use a computer.

Thus, labs on chips will have to meet two major requirements to be developed in the industrial environment:

- the development of reliable, reproducible and standardized mass manufacturing tools

- the development of virtual prototyping tools accessible to all.

These two points explain the success of the industrialization of microelectronics. Our basic postulate is that the development of microelectronics, which has been dazzling these last 50

years thanks to the separation of manufacturing technologies from modeling skills, can be mimicked. This separation resulted in the free creation of specialized industries, either in manufacturing or in modeling, independent from each other.

The specialty of our team oriented us towards the development of virtual prototyping tools for labs on chips. As a result, we became interested in the development of design tools for labs on chips, but found no design support tool taking the lab-on-a-chip as a whole into account.

Labs on chips are complex tools relying on many fields (physics, chemistry, biology, etc.), so that a prototyping tool would require many models for the different parts of the lab-on-a-chip to quickly communicate with each other.

To this end, we considered the development of a tool for lab-on-a-chip prototyping in a microelectronics simulation environment. To build our tool, we needed many model libraries from different areas so as to take all the features of a lab-on-a-chip into account, *i.e.*,:

- fluid mechanics
- biological or chemical reactions
- the response of biosensors
- the electronic control of all ancillary tools.

By exploring the literature, we realized that models designed for these different fields already existed, but none of them allowed for the global modeling of a lab-on-a-chip. Therefore, we were interested in the interconnection between these different models and in the development of new models – especially microfluidics models – to best meet the needs for the development of our virtual prototyping tool.

This thesis manuscript contains 5 chapters:

- an introductory chapter about existing labs on chips and models
- a chapter presenting the results of current literature on the modeling of microfluidics using equivalent electrical diagrams
- a chapter about the development of a compact model for modeling a passive mixer
- a chapter about the development of a 2D / 3D simulation tool, compatible with our other models, and about microfluidics allowing more precise simulation for the prototyping of a lab-on-a-chip
- finally, a more experimental chapter about the development of a microfluidic PCR prototype to develop associated thermal and biological models. The end of this chapter also presents a virtual prototype image of a complete lab-on-a-chip using all the tools developed during this thesis.

Chapter 1

Introduction to labs on chips

This first chapter deals with the reflections that led to the design of this thesis subject. First, we are going to explain what a lab-on-a-chip is: its advantages, its composition (microfluidics, biological reaction, biosensor and electronic control devices), and detail its two main elements:

- the microfluidic circuit
- the biosensor(s)

Then, we are going to explain why the industrialization of a lab-on-a-chip is a complex issue, and this will bring us to the need for the development of a design support tool taking all the different physico-chemical interactions of a complete lab-on-a-chip into account.

We will review the different modeling tools currently used for labs on chips, and then we will see that many tools are used to model the different parts of a lab-on-a-chip. However generalized design support tools considering all the different components of a lab-on-a-chip at the same time are still lacking.

1. General principles and examples of applications

1.1. What is a lab-on-a-chip?

The lab-on-a-chip technology focuses on the development of hybrid devices that combine fluidic and electronic components on a same chip [4]. They are mainly devoted to the testing and handling of liquid samples. The functions of a lab-on-a-chip include sample handling, mixing, filtering, analysis, and monitoring. A typical lab-on-a-chip device contains micro-channels for liquid samples to flow inside the chip, but also includes measurement, sensing, and actuation components such as micro-valves, microfluidic mixers, microelectrodes, thermal elements or optical devices. Some commercial lab-on-a-chip devices include electronic components for the control of different parts of the lab-on-a-chip (see examples of applications below).

Labs on chips are made up of three main parts that need to be adapted to the miniaturization of component analysis:

- A microfluidic circuit drives the sample to its area of analysis, and also performs any necessary pre-analysis step such as mixing or preconcentration.
- The analysis system is also adapted to transform a chemical or biological signal into an electrical signal. In addition, it will generally have to process small quantities of low-concentration samples. Devices called biosensors have been developed to solve this problem.
- The interfacing and control devices remain broadly identical to the tools used for conventional automated analysis techniques, but they have been adapted to their new use, whether through the use of microcontrollers or control by computer software.

1.2. Main assets

Compared to conventional benchtop instrument technologies, labs on chips reduce analytical devices, resulting in better integration, increased sensitivity, minimal reagent consumption, sterilization, better detection and efficient sample separation.

As lab-on-a-chip devices are designed to automate laboratory processes, some of their notable technical advantages are compactness, portability, modularity, on-board computing, automated sample handling, low electronic noise, limited power consumption, and the simple integration of components. In addition to the need for minimal amounts of samples, analytes or reagents, lab-on-a-chip devices are fully enclosed, reducing contamination of transported samples.

Lab-on-a-chip devices also support a wide range of processes such as sampling, routing, transporting, distribution and mixing, mainly with reduced moving or rotating components, and this increases the ease of use and the lifespan of the device. Due to their small size, lab-on-a-chip devices provide precise fluid transport through the use of electrokinetics or micro-pumping, efficient separation of liquid samples, and precision in sample measurements. Although fluid transport generally involves continuous flows, segmented droplet-based flows are also favored [5]-[7].

Because of the low volumes of fluid they handle, lab-on-a-chip devices can reduce product synthesis time and sample analysis time. They can measure samples with greater accuracy (provided that they have enough data and volume), but most essential is their ability to control chemical reactions through effective control of the concentrations of reagents. Labs on chips can provide for cascade or parallel sample processing. The advantage of parallel processing is that different samples can be tested simultaneously with different reagents, so that the efficacy of the product can be characterized efficiently.

1.3. Scope

Biochemists, synthetic chemists and physicians are currently evaluating the potential of lab-on-a-chip devices in the context of biochemical synthesis, analysis, and screening. Specific areas of application have emerged, from the detection and diagnostics of infectious diseases to food quality assessment.

In pharmacy, lab-on-a-chip devices are gradually becoming valuable in drug research, with a focus on cell targeting [8],[9] and drug synthesis [10],[11]. Lab-on-a-chip devices are promising for drug analysis and for determining optimal dosages. Microfluidic on-chip networks provide unique opportunities to mimic natural veins, test nanoparticles as drug vectors, target cells, and also examine the *in vitro* metabolism of biological cultures.

Technical limitations such as size reduction, sample entry rates, power consumption, but also chip reliability and biocompatibility all need to be considered in the design of lab-on-a-chip devices.

Labs on chips allow for the development of next-generation portable and implantable bioelectronic devices. Due to their capacity for biological detection and their integration concept, lab-on-a-chip systems are attractive platforms for the development of implantable bio-inspired sensors [12].

Lab-on-a-chip technology has adopted most of the benchmark analytical chemistry methods across different fields of physics: electrochemistry, optics, mass physics, and acoustics physics.

1.4. A few concrete examples

Our first example of a lab-on-a-chip is taken from the work of Zeynep Altintas *et al*. [13]. This lab-on-a-chip detects the bacterium *Escherichia coli* within a range of 1.99×10^4 cfu / mL down to 50 cfu / mL. A description of the lab-on-a-chip is given in Figure 1. Figure 1A shows the complete device, composed of a sample holder and the reagents necessary for the measurement. These are collected and injected into a cassette made up of the microfluidic system and the analysis system, using an electrical device (Figure 1B). Figure 1B is an exploded-view diagram of the cassette with an inlet / outlet of fluid in a microfluidic cavity where the different reagents are mixed and led to the biosensor. The biosensor detects the bacteria by an electrochemical measurement whose principle is developed in section 2.2 [13]. Then, the signal from the biosensor is processed by the computer to be viewed on a monitor (Figure 1C).

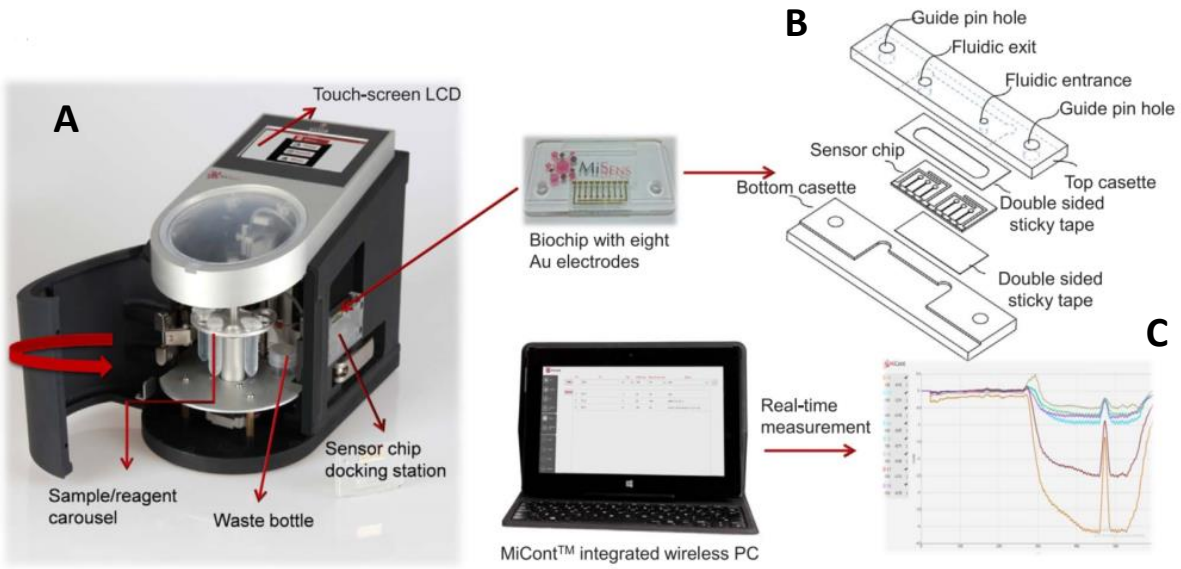


Figure 1. Description of the lab-on-a-chip designed by Z. Altintas et al. [1] to detect bacteria.

A second example is a system developed by Chao-Wei Huang *et al*. [14], which optimizes DNA sequencing of animal genetically modified organisms (GMOs). It has a complex microfluidic circuit with several mixing zones (Figure 2):

- The extraction zone (in yellow) is used to mix a blood sample with a lysis buffer, *i.e.,* a solution that breaks up cell membranes to release cell DNA
- The amplification zone (in red) is used to mix DNA extracted from blood with reagents to perform a polymerase chain reaction (PCR) [15]. PCR is used in biology to multiply DNA strands. It is based on successive biological reactions and requires temperature control. This step immobilizes the DNA and the reagents in a cavity that is heated and cooled several times to increase the DNA concentration of the sample and amplify the signal for the biosensor [16]. The PCR is detailed in Chapter 5.
- The detection area (in green), where DNA strands are mixed with probes. A probe is a DNA fragment whose sequence is complementary to the sequence we want to detect (target DNA), and is labeled with radioactive or fluorescent compounds or an antibody. If the PCR solution contains the target DNA, the probes will bind to it, and this binding will be detected by the biosensor. In this application, the probe is specific to a single nucleotide, and the binding with the target DNA is detected by fluorimetry. The method is called single nucleotide polymorphism (SNP) [17], and reveals genetic changes in DNA.

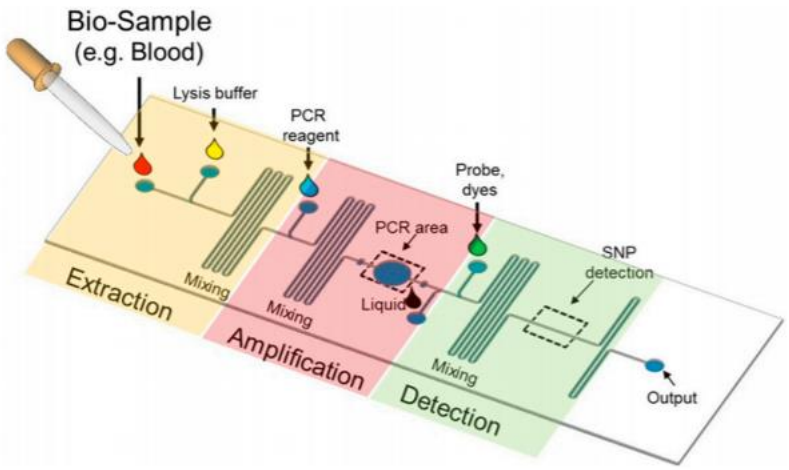


Figure 2. Example of a lab-on-a-chip for detecting DNA in a blood sample.

2. General presentation of biosensors

2.1. General architecture

As seen in the previous paragraph, labs on chips are mostly intended to measure biological signals. This measurement is done using biosensors, *i.e.,* analytical devices providing quantitative and / or qualitative results.

Figure 3 gives a synoptic view of a biosensor, which consists of several parts. Left to right:

- The analyte is the biological signal to be measured. This signal can be a molecule, a single cell, or a population of cells.
- The bio-receptor recognizes the analyte through a biochemical reaction. It is usually composed of enzymes or binding proteins, such as antibodies immobilized on the surface of physicochemical transducers. The term immunosensor is often used to describe biosensors that use antibodies as a biorecognition system. Enzymes and antibodies in these systems can also target nucleic acids, bacteria and other unicellular organisms, or even whole tissues of higher organisms.
- The transducer is sensitive to specific interactions between the target analyte and the bio-receptor. The reaction induced when these two elements are brought together produces physicochemical changes detectable and measurable by the transducer, which translates them into an electrical signal. The transducer can be based on several physicochemical principles: electrochemical, optical, measurements of mass or acoustic wave variations.
- The signal processing unit transforms the electrical signals measured by the transducer into data that can be used by the user. This part consists almost exclusively of electronic components performing simple analog functions (amplification, filtering, demodulation, etc.)
- The control unit notably includes the user interface that controls the measurement process, drives actuators, collects measurements, and displays results.

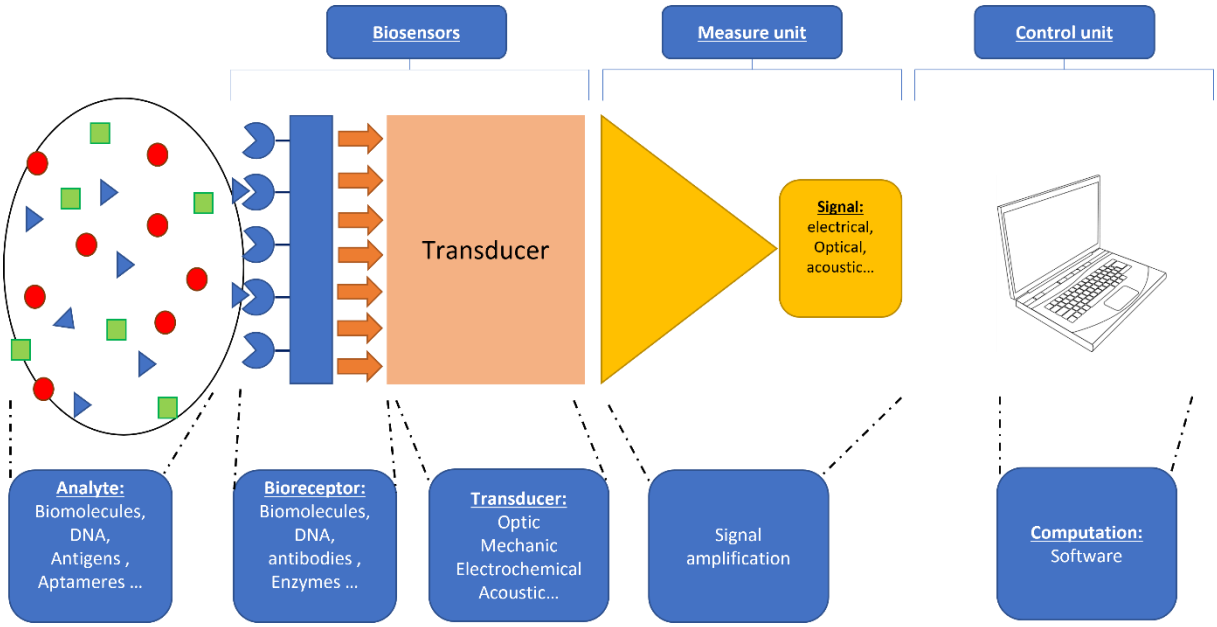


Figure 3. Synoptic diagram of the main components of a biosensor and its environment.

Biosensors combine the existing selectivity of biology, the data processing power of modern microelectronics and optoelectronics to offer new high-performance analysis tools dedicated to medicine, the environment and the food industry.

In the remainder of the chapter, we further describe the different types of transducers likely to be found in a lab-on-a-chip; this list is of course not exhaustive.

2.2. Electrochemical transducers

The principle of an electrochemical transducer is to measure changes in electrical properties caused by the presence of specific biological targets. These changes can be caused by interference in electric fields [18], chemical reactions [19], or conductive probes attached to the analyte [20].

Electrodes (usually made of gold) are used as interfaces to apply electric fields to the samples being tested and measure the electrical signals. The types of electrical signals also vary depending on the application: they can be simple impedance or resistance measurements [18], capacitance measurements [20] or impedance spectroscopy performing measurements within a range of frequencies [19].

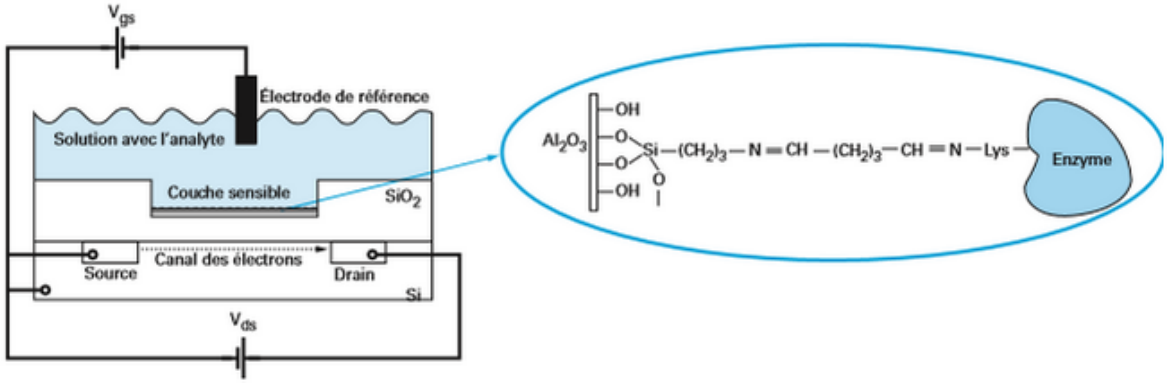


Figure 4. Diagram representing a potentiometric biosensor.

The electrodes used in electrical and electrochemical biosensors are not just a simple interface: their properties can have a significant impact on the function and efficiency of the biosensor. A common example is the use of interdigitated electrodes (IDEs) in biosensors. As showed in Figure 5b, IDEs consist of two nested but separate metal plates bearing each a number of individual fingers that overlap those of the other electrode. Applying static or alternating voltage to IDEs creates an electric field between the fingers. This electric field can be disturbed and altered by the presence of specific target biomolecules, cells or electrically active markers, and the resulting change can be measured. Almost all IDE biosensor applications use gold as the electrode material because it is highly biocompatible [21]. Biocompatibility is the biological compatibility of a material that does not cause toxicity or alter biological compounds (proteins, DNA, cells, etc.).

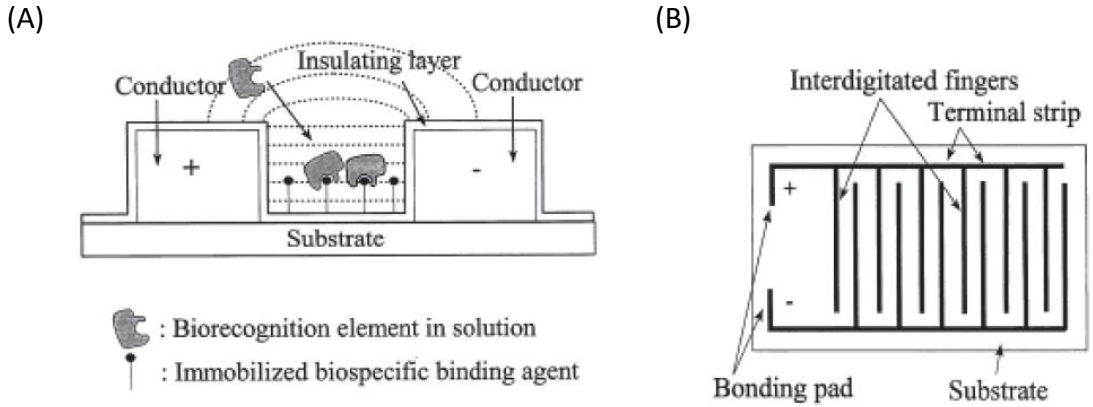


Figure 5. (A) Affinity biosensor with interdigitated electrodes. (B) Capacitance affinity biosensor using two metal plates. The changes in the dielectric properties of the field located between the electrodes following the binding or the release of an analyte within or out of the field located between the electrodes induce a change in capacitance [103].

The application of the biosensor dictates the dimensions of the IDEs, which can have a great impact on their performance. The width of the fingers and their spacing out are particularly important. A smaller spacing between the fingers generally results in a higher detection sensitivity. Sizes of several micrometers (for finger width and spacing) have been showed effective in detecting cells and bacteria [9], while electrodes smaller than one micron have been manufactured to be sensitive enough for detecting DNA hybridization [22].

IDEs are used for different types of electrochemical biosensors, but for other purposes too. They are also used in acoustic biosensors, as explained below.

2.3. Acoustic transducers

The fundamental principle of acoustic transducers is to measure the variations in the properties of waves propagating through a material when it is in contact with the analytes. As showed in Figure 6, the transducer consists of two series of IDEs: one of them is used to generate surface acoustic waves (SAWs), while the other serves as a SAW detector. In-between, the wave crosses a delay line whose properties are measured. Two types of SAWs are frequently used, namely Rayleigh waves and Love waves. Rayleigh waves are perpendicular to the surface between IDEs (Figure 6a), while Love waves are in the plane of the surface between IDEs (Figure 6b). The input IDEs generate a SAW at a particular resonant frequency.

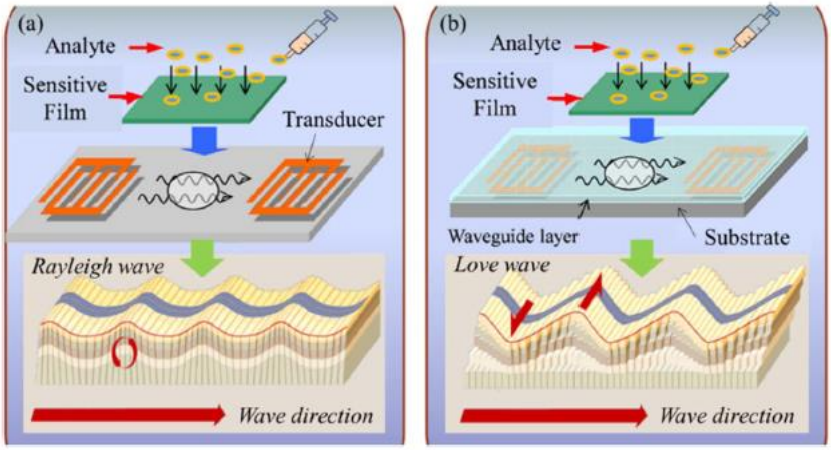


Figure 6. Propagation of surface acoustic waves (SAWs). (a) Diagram of Raleigh waves, that display both components of a perpendicular surface and of a parallel surface to the wave direction. (b) Diagram of Love's horizontal shearing waves. A waveguide layer maintains most of the vibrations near the surface.

The propagation of SAWs across the delay line is disturbed when a mass is applied to its surface, so that antibodies are immobilized on the delay line; then, they can capture the analyte and induce mass changes [23].

2.4. Optical transducer

As their name suggests, optical transducers detect changes in the optical properties of a solution depending on whether the analyte is present or not. They have many applications in research [24]. Generally speaking, two detection protocols can be implemented in optical detection, namely i) detection with a probe and ii) detection without a probe. In addition, different optical properties can be measured: optical density, turbidity, color, or fluorescence emission; the latter is probably the most commonly exploited one.

In fluorescence detection, target molecules or biorecognition molecules are modified with fluorescent markers also called fluorophores. The intensity of fluorescence indicates the presence of the target molecules that have reacted with the molecule marked by the fluorophore. While fluorescence detection is extremely sensitive, with the limit of detection

down to a single molecule, it requires labeling the analyte with the probes, which is a tedious process and can also interfere with the function of the biorecognition molecule by altering its efficiency in recognizing the analyte. Quantitative analysis is difficult due to the bias of the fluorescence signal, as the number of fluorophores on each molecule cannot be precisely controlled [25]. Besides measuring the intensity of the fluorescence signal, measuring its temporal decay may also prove useful, as the half-life of the signal is characteristic of a fluorophore. The half-life is half of the characteristic time during which the molecule remains in the excited state before returning to its basal state. This technique is called time-resolved fluorescence and makes it more particularly possible to distinguish several fluorophores in a single measurement [26].

In contrast, in unmarked detection, the target molecules are not labeled or modified and are detected in their natural forms. For example, certain molecules of interest are naturally fluorescent and can be detected by the techniques described above, but without requiring a probe [27]. Another untagged detection mechanism consists in measuring refractive index (RI) changes induced by molecular interactions and related to the sample concentration or the surface density [28]. Therefore, the detection signal does not decrease with the sample volume. This characteristic is particularly interesting when a detection volume in the order of the picoliter is involved, and is advantageous over detection by fluorescence whose signal generally depends on the total number of analytes present in the volume or on the surface [29]. Within this research field, we can mention time-resolved fluorescence (TRF) analysis, which is considered to be one of the main research tools in biochemistry and biophysics. One application of this method is the study of biomolecular interactions, with promising applications in biodetection [30]. Finally, we can mention detection by optical density (the opacity of a solution), which remains a reference measurement for monitoring cell populations [31].

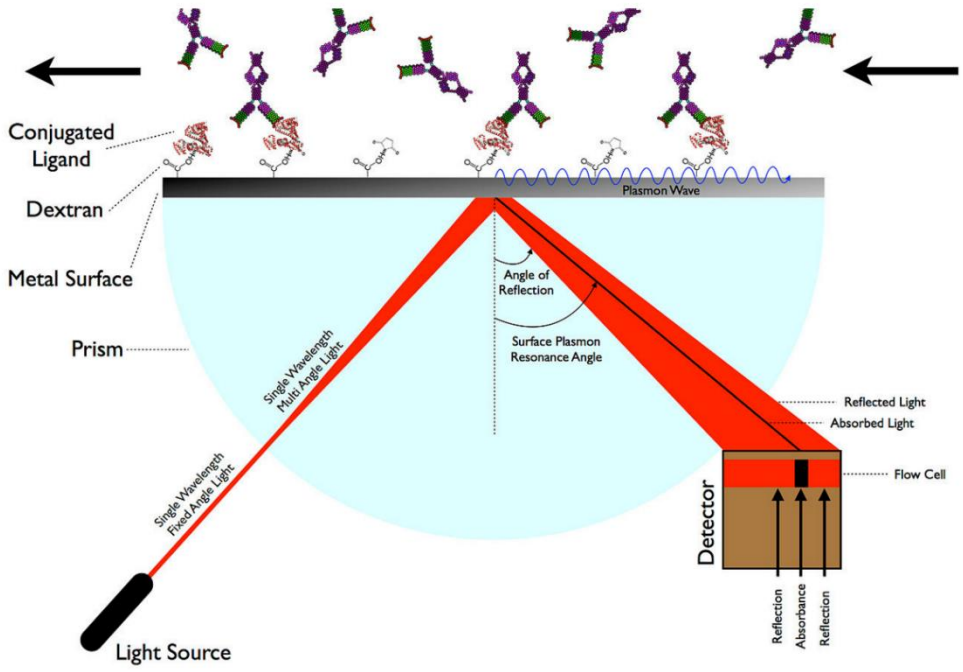


Figure 7. Overall diagram of surface plasmon resonance (SPR) detection.

Among non-labeling measurement techniques, biosensors based on the principle of surface plasmon resonance (SPR) have recently emerged (Figure 7) [32]. SPR is an optical analytical surface technique that analyzes interference variation on a given conductive surface. Surface plasmon is a non-radiative surface electromagnetic wave that propagates in a direction parallel to the interface of a dielectric material of negative permittivity. Since the wave is located at the border of the conductor and the external environment (*e.g.,* air, water, or vacuum), these oscillations are very sensitive to any variation of this border, for example the adsorption of molecules on the conductive surface. Thus, bio-recognition of an antibody previously fixed on the sensitive metal layer and tagged with the analyte modifies the plasmon wave. A monochromatic light source is reflected on the back of the metal layer. The modification of the plasmon wave at the level of the surface antibodies that have reacted with the analyte prevents the reflection of the monochromatic wave by absorbing its wavelength. Thus, we can provide imaging of surface interactions [32].

3. General presentation of microfluidics

As previously seen, the sample of interest has to be handled a certain number of times before it can be analyzed by the biosensor. Whether it is simple transport or a chemical reaction, these handlings should ideally occur without requiring any action from the operator. In a lab-on-a-chip, these actions are performed by microfluidic chips.

Microfluidics studies the mechanics of fluids when low volumes are used. This field has been growing since the early 2000's with the development and popularization of manufacturing techniques for microfluidic circuits made of polydimethylsiloxane (PDMS), a particularly malleable, pressure-resistant and biocompatible silica-based polymer. George Whitesides was one of the pioneers in the use of PDMS for microfluidics. He defined microfluidics as "the science and technology of systems that manipulate small volumes of fluids (10^{-9} to 10^{-18} liters), using channels the size of a few tens of micrometers" [33]. According to Patrick Tabeling, "we can define microfluidics as a discipline dealing with the flows of simple or complex fluids, mono or multiphase, in artificial microsystems, that is to say, manufactured using new techniques" [34]. P. Tabeling specified that by "new techniques" he essentially meant micro-manufacturing techniques inherited from microelectronics. We can more broadly define microfluidics as the science and technology of systems handling fluids and having at least one of their characteristic dimensions in the order of a micrometer.

Microfluidics is a fairly large, multi-faceted field. In this thesis, only the fluidics of liquids, which corresponds to a large majority of labs on chips, is discussed. Liquid microfluidics is divided into three main areas.

3.1. Droplet microfluidics: biphasic microfluidics

Biphasic microfluidics consists in creating microscopic drops by mixing two immiscible liquids (*e.g.*, hydrophilic and hydrophobic ones) so as to isolate the samples to be analyzed from each other. As showed in Figure 8, three main techniques are available for making drops. In each of them, the flow rate Q_C (external droplet fluid flow rate) is always greater than the flow rate Q_D (internal droplet flow rate), and variation in the ratio of the two flow rates influences drop size. Drop formation is due to the breakdown of surface tension that occurs with the continuous growth of the droplet in an immiscible medium.

This technique is extensively studied in the field of drug design to create chemical microreactors [35] or in the field of microorganism detection through their incorporation into droplets that can themselves be immobilized for analysis [9]. A more detailed example of one of these devices is presented in Chapter 2. The optimization and uniformization of the droplet shape with a flow rate suitable for use remains an open field of exploration in microfluidics.

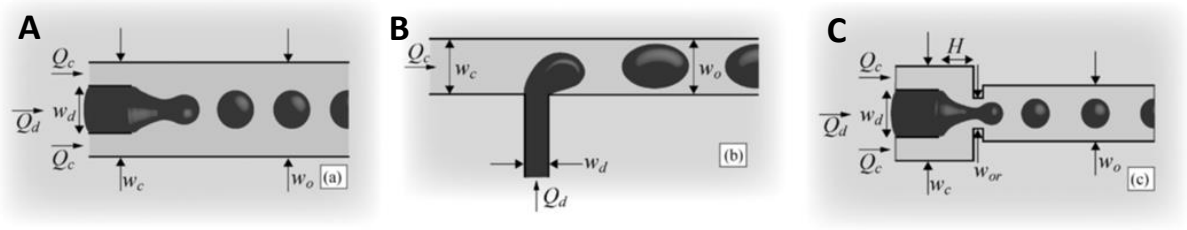


Figure 8. Illustration of the three main microfluidic geometries used for forming droplets. (A) Streams flowing in the same direction; (B) cross-cutting streams in a T-junction;(C) an extended flow within a geometry that focuses the flow [2].

3.2. Digital microfluidics

Digital microfluidics (DMF) is an emerging liquid-handling technology that turns liquids into discrete droplets within integrated microfluidic devices. In the present thesis, the term "digital microfluidics" is used to depict integrated systems in which the droplets are handled on an array of electrodes, rather than on systems in which they are handled in closed microchannels as seen in the previous section. In DMF, droplets from a picoliter to a microliter in size are independently addressed onto an open array of electrodes coated with a hydrophobic insulator. By applying a series of potentials to these electrodes, the droplets can be handled individually and merge, mix, or divide (Figure 9). As with the channel-based microfluidic format (seen earlier), DMF offers the advantages of low reagent consumption and rapid heat transfer, and it can be easily integrated with other analytical techniques. Please note that DMF has interesting characteristics that are not found in channel-based microfluidics. In DMF, each droplet is individually controlled without needing channel networks, pumps, valves or mechanical mixers. Thus, various processes can be carried out simultaneously, with a simple and compact design. The droplets are handled by generic arrays of electrodes, and the droplet operations are reconfigurable from one experiment to another.

Taking samples after preliminary steps is also straightforward, as the droplet volumes are relatively large and DMF devices are often open onto the atmosphere. In addition, solid samples in DMF systems can be handled and used without the risk of clogging. [36]

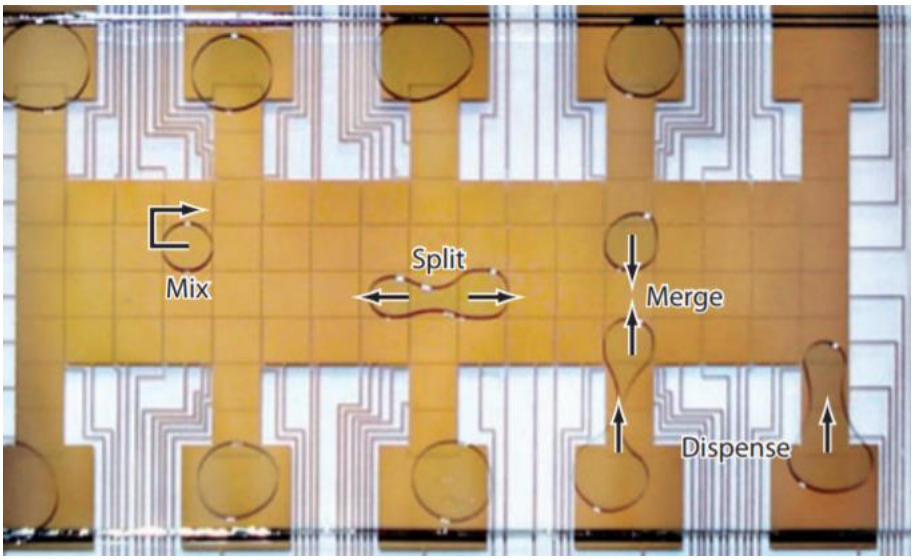


Figure 9. Example of a network for handling droplets for digital microfluidics.

3.3. Continuous-flow microfluidics: monophasic microfluidics

This field of study works with fluids containing the sample(s) to be analyzed in continuous flow [14], [37] - [39]. At small dimensions, the effects of macroscopic physical phenomena do not just decrease linearly. Certain negligible phenomena such as capillarity become predominant. Conversely, other forces such as gravity become negligible [40]. The present thesis manuscript is focused on this type of microfluidics, which is detailed in Chapter 2.

4. Lab-on-a-chip design tools

In this final section, we discuss lab-on-a-chip design from the perspective of computational tools and computer-aided design.

4.1. The issue

As seen above, lab-on-a-chip design involves modeling several aspects, namely microfluidics, the biosensor, and potential biological / chemical reactions. As each of these aspects has to be considered, it is essential that modeling be multi-domain, including fluid mechanics, biology and electronics.

Despite all the advantages that labs on chips can bring, they face great difficulties in their industrial development. The technical demands for manufacturing are very high, and the parameters to be considered to simulate these devices are just as important. Therefore, both efficient design tools and state-of-the-art manufacturing techniques are required.

However, current research has focused a great deal on manufacturing techniques and the optimization of each of the lab-on-a-chip components, but has barely investigated design support tools.

Many tools are available for each of the fields. However, designing each domain individually and then linking them back together can lead to a lot of malfunctioning as well as a waste of time and money for a complete lab-on-a-chip model.

A parallel with industrial development can be drawn with microelectronics, whose industrial development occurred once design and manufacturing were separated; this allowed

manufacturers to specialize in one of these branches and experience the success that we are witnessing today. Therefore, by following the path of microelectronics, we can easily think that the future of labs on chips will depend on task specialization. Two professions would be involved, namely designer and technology expert. The technology expert will provide the designer with a library of know-hows (*e.g.,* the minimum distance required between the mixer coils, the maximum width of the channels, the maximum depth of the channels, gluing, design rules in electronics) as a cornerstone, with an associated model. The designer will assemble these building blocks within a software program, carry out simulations, optimize them, and send a design file back to the technology expert once everything seems to be working. Finally, the technology expert will manufacture the lab-on-a-chip for the designer.

The industrialization of labs on chips will most likely follow in this wake. While microfluidics is indeed booming, many works relate to its industrialization using lithography [41] or even 3D printing [42], but design support software for labs on chips is scarce. These programs are generally incomplete and only take some of the components of a lab-on-a-chip into account.

No universally adopted language or tool for labs on chips is yet available. Some are too specific to a given domain, others are too complex and cannot be used at the level of the complete system, others still are too simplistic to accompany the design process. This is what we are going to develop in the next section.

4.2. High-level description languages

The computer-aided design (CAD) of labs on chips starts with the existence of a language that describes the actions done in the lab-on-a-chip with a relatively high level of abstraction. BioStream is the pioneer in this domain. It is a software environment for describing an experimental protocol as a sequence of fluid injections into a reactor. An extension of BioStream named BioCoder [43] was launched later, in 2010. It describes the experimental protocol more specifically. At the start of the simulation, the list of reagents (solids, liquids, etc.) and the tools to be used (test tube, sterile bottles, etc.) are described. Then the user can program pre-established functions such as *centrifuge (), measure_fluid (), store ()*, etc. This description is compiled to validate the reliability of the experimental protocol, issue a textual description of the protocol and list all the necessary tools for the completion of the protocol.

BioCoder is used to describe a protocol, but does not make the link with practical implementation by a lab-on-a-chip. For this, it would be necessary to have i) a modeling and simulation environment encompassing the different aspects of the lab-on-a-chip, and ii) a library of elementary and interoperable components (microfluidics, biological reaction, biosensor) for which we would have a model and that could be assembled to perform the function described by BioCoder.

4.3. Modeling and simulation environment

Lab-on-a-chip models and simulations can be imagined at different levels of abstraction. In engineering sciences, a level of abstraction refers to the potential distance between the described behavior and the actual behavior of the described item. A high level of abstraction corresponds to a very superficial description of objects: we describe functions, their behavior, their transfer function, etc. It is usually mathematically and computationally simple but does

not ultimately describe how a system works in detail. Conversely, a low level of abstraction corresponds to a very detailed description of objects: for example, we describe the physics of a component, its internal architecture, etc. It is much more complex from a mathematical point of view and much less efficient in terms of simulation time. On the other hand, it is much more precise and integrates as many parasitic effects (non-linearities, offsets, etc.) as we want.

In the following section, we are going to see how we can model the different parts of a lab-on-a-chip at different levels of abstraction. Several elements already exist in the literature, and others are the subject of this thesis work.

4.3.1. Microfluidics

Most of the existing microfluidics models are based on fluid mechanics. Therefore, these are low-level abstraction models using partial differential equations (PDEs). Much commercial software is available to simulate this type of model. COMSOL [44] or Fluent are the most common ones. Open-source alternatives can be found, such as FreeFEM ++ or Feel ++. These tools are particularly effective to simulate and optimize certain parts of the microfluidic circuit [45]. On the other hand, the use of such models for large microfluidic circuits is prohibitive as regards computation time.

The modeling and simulation of high-abstraction microfluidic circuits recently received a lot of attention, as an analogy with electronics was demonstrated for continuous-flow microfluidics [46]. Kirchhoff's laws apply to microfluidic circuits, hence the development of microfluidics using the design tools of microelectronics, in particular for the description of complex mixer circuits to create a concentration gradient [47]. This microfluidic circuit modeling technique is the basis of our working hypothesis, so it is detailed in the following chapter.

These techniques were developed to respond to a demand for simple modeling of microfluidic circuits. As a consequence, works such as those of Elishai Ezra Tsur *et al*. recently emerged: they developed a tool to create a file suitable for 3D printing based on users request for a microfluidic circuit using this analogy system [48]. This work is detailed in Chapter 2.

4.3.2. Biological reactions

In parallel with microfluidics, the modeling and simulation of biological reactions have evolved a lot in recent years, especially with the advent of bioinformatics. These *in silico* models can take on several facets.

At the molecular level, lower-level modeling describes the interactions between the atoms forming the molecule and neighboring molecules so as to determine the geometry of the molecules and their active zones. Models are based on particle physics equations. Software programs such as HADDOCK are available [49],[50]. HADDOCK is a flexible and versatile data-driven docking approach for modeling biomolecular complexes. By providing experimental data on the interactions among the atoms of a molecule (*e.g.*, NMR results), the software program reconstructs the molecule or molecules under study in 3D. If several molecules reacting together are studied, their reaction can be modeled. By taking the least energetic

solutions into account, the software program models the different molecules in 3D and highlights the interaction zone(s).

A typical simulation from [51] is presented in Figure 10. The 3D modeling of the two proteins is showed in the upper part of the figure, in two different colors. A zooming on the interaction area is showed in the lower part of the figure; the amino acids belonging to the protein on the left are in blue, and those belonging to the protein on the right are in red. The zoom box provides the symbols of the amino acids (*e.g.,* K, R, D) and their position. Although this modeling method is particularly powerful and precise, it can consume a lot of computational power for already known reactions.

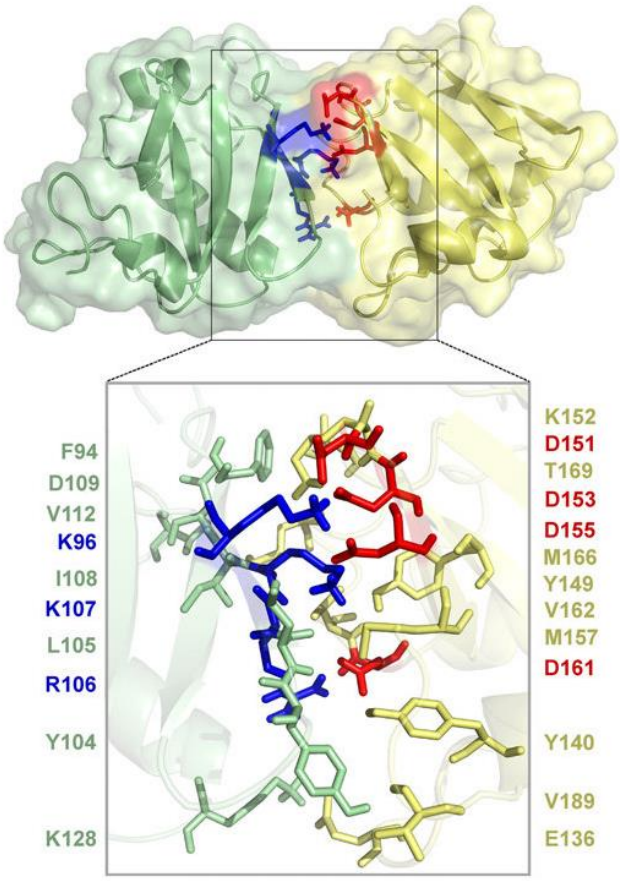


Figure 10. Model based on the HADDOCK data of the homodimer PsIAA4 PB1. A protein-protein anchored model of the homodimer PsIAA4 PB1 was created by HADDOCK, using NMR data.

A second approach at a medium level of abstraction consists in modeling the molecule itself as a particle, and studying interactions with other molecules in solution. This type of simulation can be done deterministically or stochastically. Deterministic simulation solves the diffusion-reaction equations that are also partial differential equations. Stochastic simulation involves a random walk of molecules and probabilities of interactions. HSIM is one of the benchmark software programs in this field [52]. At this level of abstraction, computation time increases exponentially with the total number of molecules involved in the reaction.

The third approach consists in considering no longer each molecule individually, but the concentration of each of the molecules in solution. This approach is only valid when each type

of molecule is present in large numbers and their concentration is homogeneous across space. In this case, the state of the system is described as a balance of flows of molecules; each flow originates from a chemical reaction and the associated kinetics. For example, in the case of an enzymatic reaction, the high-level model would be Michaelis-Menten's equation [53]. Here again, this flow equilibrium can be calculated deterministically using a system of coupled ordinary differential equations (ODEs), using a sole stochastic approach or a stochastic approach using dedicated algorithms, like Gillespie's algorithm [54]-[56].

We limited ourselves to deterministic simulations at a high level of abstraction, or at an intermediate level of abstraction when necessary. Previous studies demonstrated that biological systems could be described by analogy to electrical circuits, whether for high-level models (BB-SPICE [57]) or intermediate-level BB-SPICE 3D models.

4.3.3. Transducer and electronics

The transducer is above all an electronic circuit to which it is necessary to add a biological input. Regarding the description of "pure" electronic circuits, many tools exist and have been optimized over time and serve as benchmarks in the industry. The simulation program with integrated circuit emphasis (SPICE) was one of the pioneers in the field in the 1970's. It is composed of i) a mechanism describing how elementary components are linked together, and ii) a simulator solving Kirchhoff's laws throughout the whole network [57]. Among the components recognized by SPICE are controllable current and voltage sources whose equations are customizable. Therefore, the biological inputs necessary for describing the transducers can be integrated.

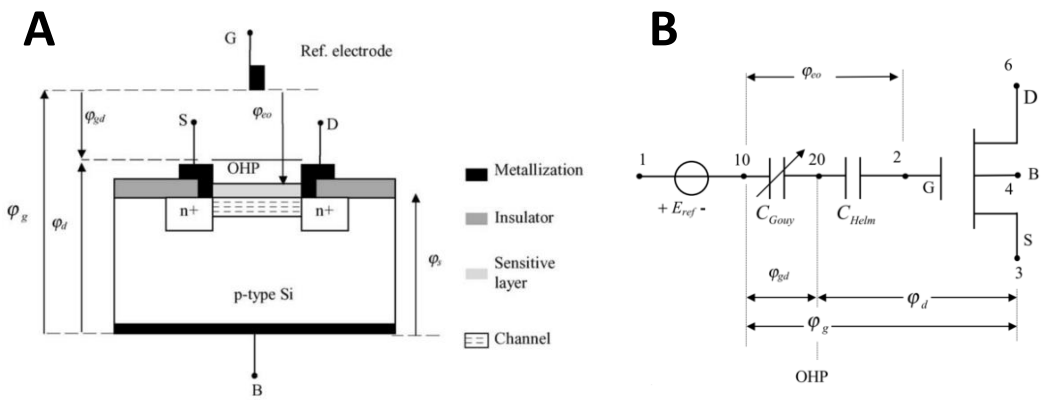


Figure 11. (A) Schematic representation of an IFSET for detecting pH changes. (B) Modeling via an IFSET-equivalent electrical diagram.

For example, many SPICE models are available for ion-sensitive field-effect transistors (ISFETs) [58]. ISFETs are potentiometric biosensors based on metal oxide semiconductor field effect transistor (MOSFET) technology, sensitive to pH changes in a solution. ISFETs are easily translatable into simulated electrical models *via* a microelectronics simulation environment (Figure 11).

The results obtained with this model can be seen in figure 12. The experimental results and those of the model simulated with SPICE are significantly similar.

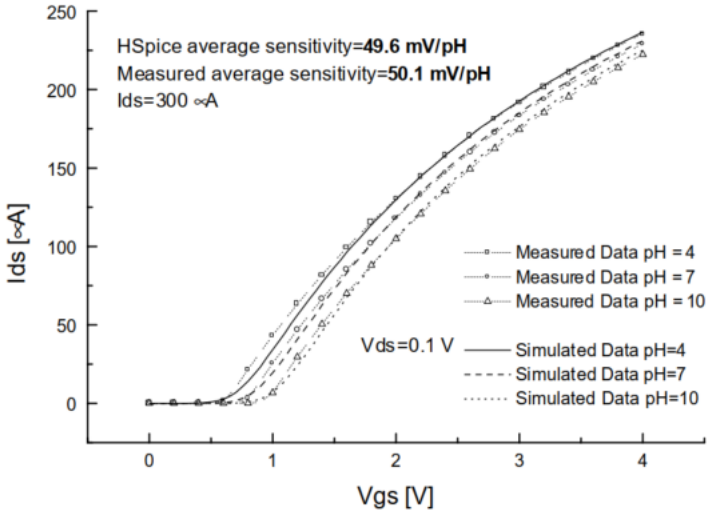


Figure 12. Comparison of SPICE simulation results and measured experimental results for an IFSET at pH 4, 7 and 10, with a gate source voltage (GSV) of 0.1.

Equivalent electrical diagram models also exist for impedancemetry biosensors [59],[60]. Figure 13A & B shows a simple diagram of the operation of an impedance biosensor where resistance to the passage of charges toward the electrode is measured. When the antibody reacts with its target, resistance to the passage of charges increases, and the reaction can be detected. Figure 13C shows the model in the form of an equivalent electrical diagram where R_{CT} is resistance to charge transfer, R_B the intrinsic resistance of the reaction medium, and C_{DL} the capacity of the double layer. Resistance to charge transfer is attributed to the electrostatic interactions of the ions in solution with the electrons in the electrodes, which produce a current. Double-layer capacitance is attributed to the spatial distribution of the ions formed near the electrode-electrolyte interface [61].

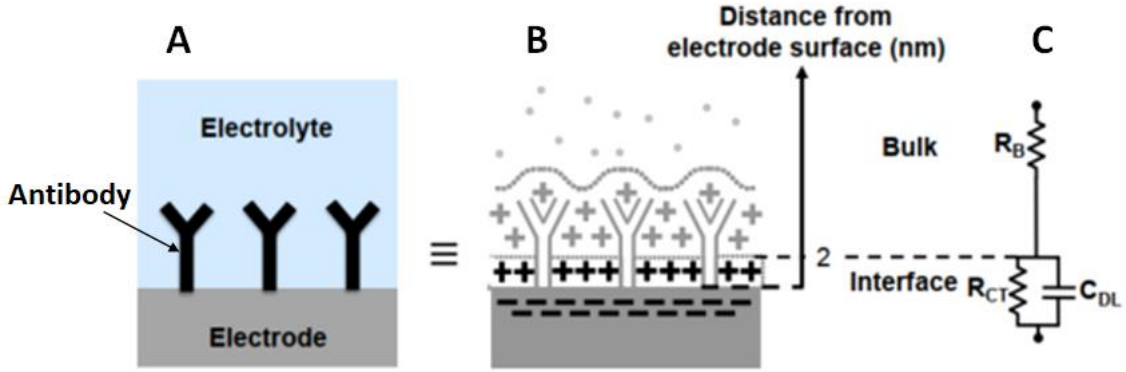


Figure 13. (A) Schematic representation of the impedancemetry transducer. (B) Schematic representation of the transducer, with charges. (C) Equivalent electrical diagram of the transducer.

The integration of other fields of physics using SPICE was facilitated in the 2000's with the emergence of hardware description languages for analog and mixed signals (HDL-AMS) [62]. Many libraries describing several branches of physics emerged through the conversion of the associated differential equations into equivalent electrical diagrams. Using dedicated libraries, this language made it possible to simulate many branches of physics interacting with each

other. The examples given in [63] show the simulations of a Pb / Fe battery and of an airbag. The battery is a coupling of electrical, chemical and thermal energies to produce a current through an oxidation-reduction reaction. For the airbag, the coupling is between an accelerometer for the sensor, an electrical control system and then an actuator that follows a cascade of electrical decisions until a chemical reaction for inflating the balloon.

4.4. Component library and assembly

As soon as we have a modeling and simulation environment common to all fields, we will be able to create a library of elementary functions for which we will have both a digital model and a procedure for practical implementation. Automating the selection of functions to be assembled in order to correspond to a high-level description, as in digital electronics, for example, seems utopian when it comes to a lab-on-a-chip. On the other hand, the creation of an environment allowing hand assembly and the validation of simulation prototypes without systematic bench validation would already constitute a major advance and seems an achievable short-term objective.

5. Conclusion

The field of study of labs on chips is very large and complex. It requires knowledge in fluid mechanics, biology and electronics.

Although this field is particularly promising, its overall complexity makes its complete modeling very long with existing simulation tools. To meet this expectation, we worked on a design support software program with several levels of abstraction to provide several possibilities for lab-on-a-chip design.

Our starting point was the ability to run simulations in an open-access simulation tool that can simulate different branches of physics at different levels of abstraction.

A work done within our team already used SPICE for multiphysics 2D simulation [64]. Therefore, we started from this working base to improve it and add functionalities to it to simulate a complete lab-on-a-chip.

The first obstacle was the simulation of microfluidics. However, we were already using SPICE, and microfluidic circuits can be simulated by an electrical diagram, as briefly explained above.

The next step was more focused on microfluidics. We developed the equations necessary to understand this field and further explain the analogy between microfluidics and electronics.

Chapter 2

Microfluidic simulation with a high level of abstraction

In this chapter, we present advances in the modeling of microfluidic circuits at a high level of abstraction, as numerous publications have showed the interest of high-abstraction modeling for microfluidics in the last decades.

This modeling goes through the simple possible analogy between analog electronics and microfluidics. As a result, the design tools used in electronics, which have been greatly optimized over the past decades, can in turn be used to design microfluidic circuits.

In this chapter, we first develop the equations necessary for understanding microfluidics. Then, we show the possible analogy with electronics, as well as the applications already developed for microfluidic circuit design.

Finally, we show how we intend to use it for our design tool.

1. Theoretical bases

The objective of this section is to recall the theoretical bases of microfluidics necessary for a good understanding of the remainder of the document. As explained in the previous chapter – 1.3 –, there exist three types of microfluidics study: continuous flow, droplet, and digital microfluidics. The models developed within the framework of this thesis more particularly concern continuous-flow microfluidics. The equations are also valid for droplet microfluidics. However, droplet microfluidics presents more complicated equations in terms of surface effects that are not presented here [65].

1.1. Reynolds number

First, we must determine the conditions of a so-called laminar microfluidic system.

One physical parameter in particular – Reynolds' number Re – is conventionally used to characterize fluid behavior in channels, *e.g.,* laminar or turbulent flow. This number is defined as the ratio of inertial forces to viscous forces.

$$\mathrm{Re} = \frac{\mathrm{F_{inertie}}}{\mathrm{F_{visqueuse}}} = \frac{\rho \cdot \mathrm{d} \cdot \mathrm{v}}{\eta} \qquad 1$$

Where ρ is density (kg / m^3), d the characteristic size of the system (m), v a characteristic velocity (m / s), and η the dynamic viscosity of the fluid (kg / m / s).

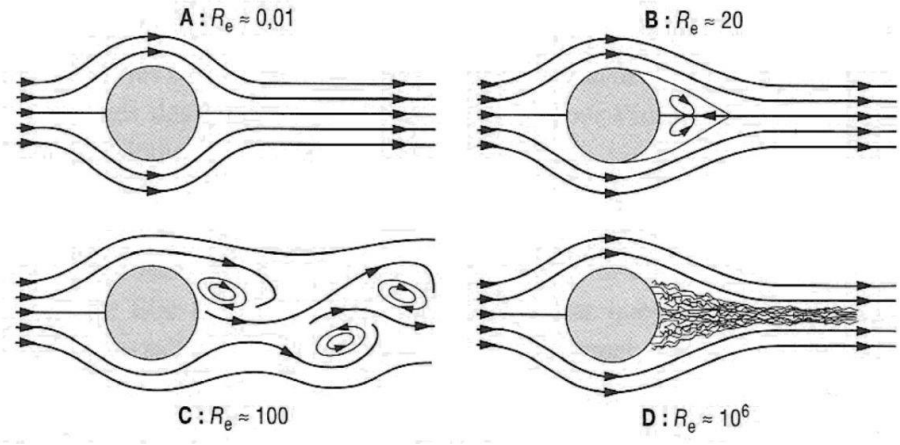


Figure 14. Different velocity fields as a function of Reynolds' number.

In order to end up with laminar flow, we need to have a low Reynolds' number (typically $Re < 10$ for straight and smooth canals [66]), where viscous forces are dominant (Figure 14).

We generally consider that we are no longer in laminar microfluidic conditions when Reynolds' number is high.

As long as the inertial forces are greater than the viscous forces, the regime is turbulent, starting from $Re > 10$ (Figure 14B, C & D).

Thus, using equation 1 describing Reynolds' number for a given fluid, ρ (density) and η (the dynamic viscosity of the fluid) are constants at constant temperature. Consequently, the only parameters that influence Reynolds' number are d (the characteristic size of the system) and v (a characteristic velocity). We should keep in mind that in a microfluidic circuit, d has an

order of magnitude of 10^{-6} m. A turbulent flow implies a very strong characteristic velocity, and in turn a very high flow in microscopic pipes. This leads to a technological hurdle due to the pressure in these pipes. This explains why microfluidics has laminar behaviors in these circuits.

1.2. Navier-Stokes equation

The conservation of mass equation is well known and can be written as follows:

$$\frac{\mathrm{d}\rho}{\mathrm{d}t} + \rho \cdot \nabla \mathbf{u} = 0 \qquad 2$$

Where ρ is density [kg/m^3], t is time [s], and $\boldsymbol{u}$ the velocity of the flow of the same fluid [m / s].

The behavior of Newtonian fluids is governed by the law of Navier Stokes [34], *i.e.*,:

$$\frac{\partial \mathbf{u}}{\partial t} = -\frac{1}{\rho} \cdot \nabla P + \nu \cdot \Delta \mathbf{u} + \frac{1}{\rho} \cdot \mathbf{F} \qquad 3$$

Where ν is the cinematic viscosity ($\nu = \frac{\eta}{\rho}$ [m²/s]), P pressure [Pa] and $\boldsymbol{F}$ is the sum of external mass forces such as gravity.

Then, $\frac{\partial \boldsymbol{u}}{\partial t}$ corresponds to acceleration, $\frac{1}{\rho} \cdot \nabla p$ to the pressure forces, $\nu \cdot \Delta \boldsymbol{u}$ to the convective forces, and $\frac{1}{\rho} \cdot \boldsymbol{F}$ to the external forces. External forces such as gravity are often neglected in laminar microfluidic conditions because the volume effects are negligible compared to the surface effects. Moreover, pressure is considered homogeneous in the width of the channel. In this case, Navier-Stokes equation can be reduced to the so-called Stokes equation in the absence of an external force:

$$\frac{\partial \mathbf{u}}{\partial t} + \frac{1}{\rho} \cdot \frac{\partial P}{\partial x} = \nu \cdot \left(\frac{\partial^2 \mathbf{u}}{\partial x^2} + \frac{\partial^2 \mathbf{u}}{\partial y^2} + \frac{\partial^2 \mathbf{u}}{\partial z^2} \right) \qquad 4$$

1.3. Poiseuille flow

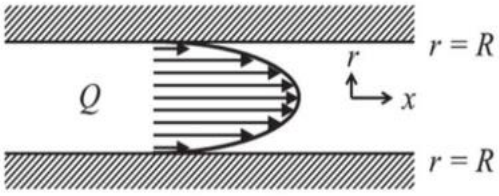


Figure 15. Representation of Poiseuille flow in a microfluidic channel.

Under laminar microfluidic conditions, let us consider a long cylindrical channel of radius R. Let us consider the system in cylindrical coordinates, with x the direction along the axis of the channel and r the radial coordinate (Figure 15). The X-axis is in the center of the channel. In steady-state fluid flow, its velocity field is unidirectional and there is no acceleration of the fluid. Thus, the unsteady terms are all zero, that is to say $\frac{\partial \boldsymbol{u}}{\partial t} = 0$. Equation 3 becomes:

$$\nabla P = \eta \cdot \Delta \mathbf{u} \quad (5)$$

Equation 5 demonstrates the balance between the net pressing force and the convection force. Considering the boundary condition such that $u(r = R) = 0$, the pressure movement, called Poiseuille flow, in the circular channel of radius R [m] is in the radial direction:

$$u = \int_0^{2\pi} \int_r^R \frac{1}{\eta} \cdot \left(-\frac{dP}{dx}\right) \cdot r \cdot dr \cdot d\theta \quad (6)$$

$$u = \frac{R^2 - r^2}{4\eta} \cdot \left(-\frac{dP}{dx}\right) = u_{max} \cdot \left(1 - \frac{r^2}{R^2}\right) \quad (7)$$

where u_{max} is the maximum speed $u_{max} = \frac{R^2}{4\eta} \cdot \left(-\frac{dP}{dx}\right)$ in the center of the canal ($r = 0$).

Poiseuille flow is characterized by a parabolic velocity profile. The flow velocity at the center of the channel is greater than that towards the outer walls [47].

To obtain the total volumetric flow Q [m^3 / s] in the circular channel, we must spatially integrate the contributions of velocity on a section of the channel (Equation 7).

$$Q = \int_0^{2\pi} \int_0^R \frac{R^2 - r^2}{4 \cdot \eta} \cdot \left(-\frac{dP}{dx}\right) \cdot r \cdot dr \cdot d\theta \quad (8)$$

As a result, the volumetric flow rate for the steady-state pressure-controlled fluid flow in the channel becomes:

$$Q = \frac{\pi \cdot R^4}{8 \cdot \eta} \cdot \left(-\frac{dP}{dx}\right) \quad (9)$$

This is Poiseuille-Hagen's law. Assuming that pressure variation is linear, Equation 9 can be simplified for the case of a finite cylinder of length L and radius R such that:

$$Q = \frac{\pi \cdot R^4}{8 \cdot \eta} \cdot \frac{\Delta P}{L} \quad (10)$$

Where ΔP is the pressure differential between the inlet and the outlet of the cylinder.

1.4. Hydraulic resistance

Equation 9 can be rewritten as follows:

$$\Delta p = Q \cdot R_H \quad (11)$$

The parameter R_H is hydraulic resistance (Pa.s/m^3). This hydraulic resistance represents the pressure drop undergone by a fluid in a microfluidic channel [34]. Based on Equations 9 and 11, we can write:

$$R_H = \frac{8 \cdot \eta \cdot L}{\pi \cdot R^4} \quad (12)$$

1.5. Case of a rectangular section channel

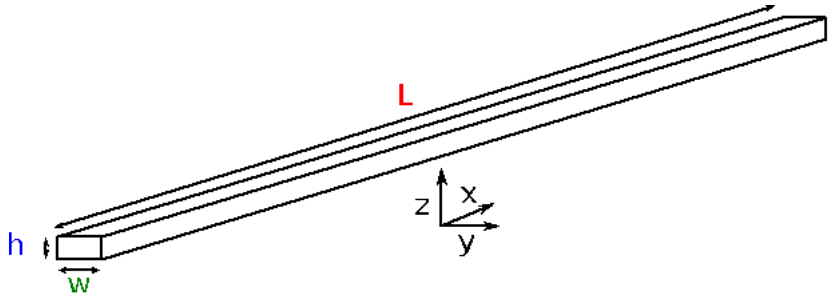


Figure 16. Diagram of a rectangular micro-channel.

Figure 16 represents a rectangular microfluidic channel characterized by three dimensions: L the length of the channel, w its width, and h its height. $L \gg h$, and $w \gg h$. Assuming that we are studying a fluid under permanent, incompressible, Newtonian flow, its density and viscosity are constants, and acceleration is zero.

Using Equation 4, $\frac{\partial^2 \boldsymbol{u}}{\partial y^2} = 0, \frac{\partial^2 \boldsymbol{u}}{\partial z^2} = 0$, we obtain:

$$\frac{\partial^2 \mathrm{u(z)}}{\partial \mathrm{z}^2} = \frac{-\Delta \mathrm{P}}{\eta \cdot \mathrm{L}} \qquad 13$$

Then, we obtain the following expression for the velocity field:

$$\mathrm{u(z)} = \frac{-\Delta \mathrm{P} \cdot \mathrm{z}^2}{2 \cdot \eta \cdot \mathrm{L}} + \mathrm{A} \cdot \mathrm{z} + \mathrm{B} \qquad 14$$

As in the circular cylinder, velocities are zero on the walls, *i.e.*, $u(-h/2) = u(h/2) = 0$, hence:

$$\mathrm{u(z)} = \frac{\Delta \mathrm{P}}{8 \cdot \eta \cdot \mathrm{L}} \cdot \mathrm{h}^2 \cdot \left(1 - \frac{4 \cdot \mathrm{z}^2}{\mathrm{h}^2}\right) \qquad 15$$

Then, we obtain the flow by integrating the section:

$$\mathrm{Q} = \int \int \mathrm{u(z)} \cdot \mathrm{dy} \cdot \mathrm{dz} = \frac{\Delta \mathrm{P}}{12 \cdot \eta \cdot \mathrm{L}} \cdot \mathrm{w} \cdot \mathrm{h}^3 \qquad 16$$

And hydraulic resistance is:

$$\mathrm{R_H} = \frac{12 \cdot \eta \cdot \mathrm{L}}{\mathrm{w} \cdot \mathrm{h}^3} \qquad 17$$

2. Analogy between analog electronics and microfluidics

2.1. Principle

Equations (10) - (11) and (15) - (16) demonstrate a possible analogy between laminar microfluidics and analog electronics. Ohm's law found in analog electronics is also found in fluid mechanics.

Electronic quantity	*Unit*	*Microfluidic quantity*	*Unit*
U	volt	ΔP	Pa
R	Ω	R_H	$Pa/(m^3/s)$
I	Ampere	Q	m^3/s

Usually found units are more in mbar for pressure, in µL / h for the flow rate and in mbar / (µL / h) for resistance. Conversions are done as follows:

- 1 m^3/s = 3.6 m^3/h = 3.6 x 10^6 L/h = 3.6 x 10^{12} µL/h
- 1 Pa = 10^{-2} mbar = 0.01 mbar
- 1 $Pa/(m^3/s)$ = 0.01/(3.6 x 10^{12}) mbar/(µL/h) = 2.777 x 10^{-15} mbar/(µL/h)

Thus, hydraulic resistors behave in a similar way to electric resistances: if N fluidic resistors $R_{H,i}$ are arranged in a series in a microfluidic circuit, the equivalent hydraulic resistance $R_{H,equ}$ of the circuit is equal to the sum of these same resistances:

$$R_{H,equ} = R_{H,1} + R_{H,2} + \cdots + R_{H,N} \quad (18)$$

Likewise, when these same resistors are placed in parallel, the equivalent hydraulic resistance is:

$$\frac{1}{R_{H,equ}} = \frac{1}{R_{H,1}} + \frac{1}{R_{H,2}} + \cdots + \frac{1}{R_{H,N}} \quad (19)$$

As in electronics, generalized Kirchhoff's laws apply to microfluidics. In electronics, the sum of the currents is zero at a node of the circuit (law of nodes). In microfluidics, the sum of the flows reaching an intersection is zero [47]:

$$\sum_{n=1}^{N} Q_n = 0 \quad (20)$$

The second fundamental law of electronics happens to be the law of energy conservation. Thus, the sum of the electrical voltages along a closed circuit is zero. In microfluidics, the sum of the different pressure differences on a closed ball must be equal to 0 [47]:

$$\sum_{n=1}^{N} \Delta P_n = 0 \quad (21)$$

2.2. Application example

The analogy between microfluidics and electronics makes it possible to use the same CAD tools, such as SPICE [67], to design microfluidic circuits and electronic circuits. A few practical applications are presented below.

2.2.1. Droplet handling

Xiaming Chen's team [5] worked on an integrated microfluidic chip performing several functions such as droplet generation, pairing, entrapment, fusion, mixing, and release. The circuit was fully controlled by a flow of liquid. It was applied to the screening of drug

compounds that inhibit tau-peptide aggregation [68], a phenomenon associated with neurodegenerative disorders.

Their approach was to design their microfluidic circuit based on the existing analogy with electronics. As showed in Figure 17, Chen modeled his microfluidic circuit from electrical diagrams to be able to handle drops containing the samples. Each trap well had an inlet on the left that connected to a short horizontal channel (referring to the main channel, *Lmain*, Figure 17A) through which a droplet could enter the well. It also had two spaces on the right that connected to the right branch of the bypass channel. This two-space design retained the trapped droplet while still allowing more fluid to pass through the trap.

While their application addressed drop movements, they modeled the movements of continuous fluids around the drops by an equivalent electrical diagram (Figure 17B).

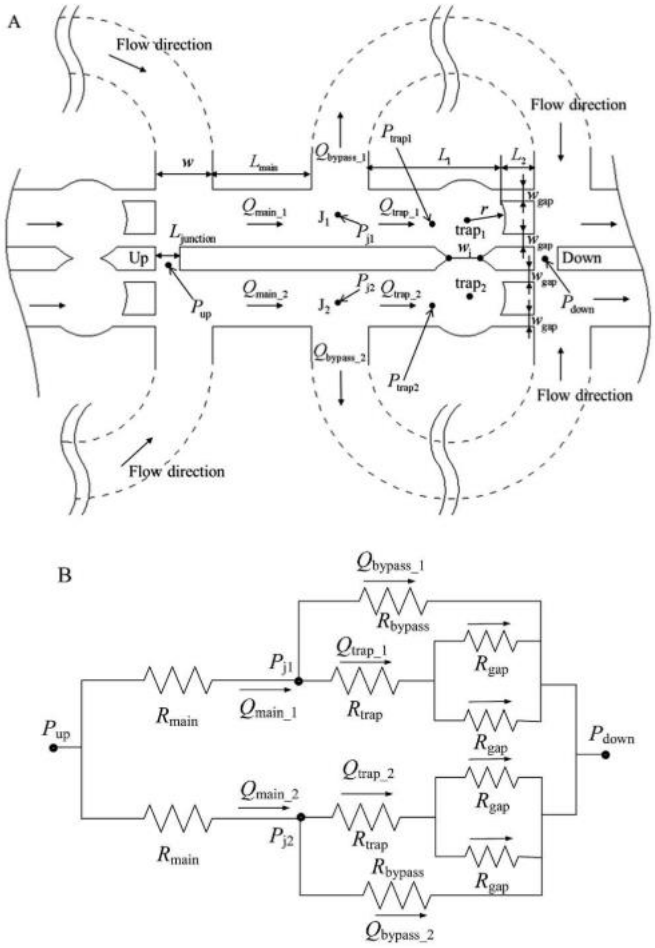


Figure 17. (A) Schematic representation of a microfluidic circuit with the dimensions required for capturing two droplets side by side for the products of each droplet to react with each other. (B) Electrical diagram of the microfluidic circuit.

2.2.2. Microfluidic dilution system

Kwang W. Oh's team developed the use of analogy for the design of many microfluidic tools, including a microfluidic dilution tool [47].

The issue of fluid mixtures in laminar microfluidics is no trivial one because mixing only takes place by diffusion of the molecules in the transverse direction (Figure 18). This problem is

further explained in Chapter 3. However, the notion of diffusion length is needed for a good understanding of this example. Diffusion length represents the length of channel necessary for a mixture considered uniform at the outlet of the channel [47] [69] [70]:

$$L_D \approx \frac{Q}{D_D} \qquad 22$$

Where L_D is the mixing length [m], Q the flow rate [m^3/s], and D_D the diffusion coefficient of the studied species [m^2/s].

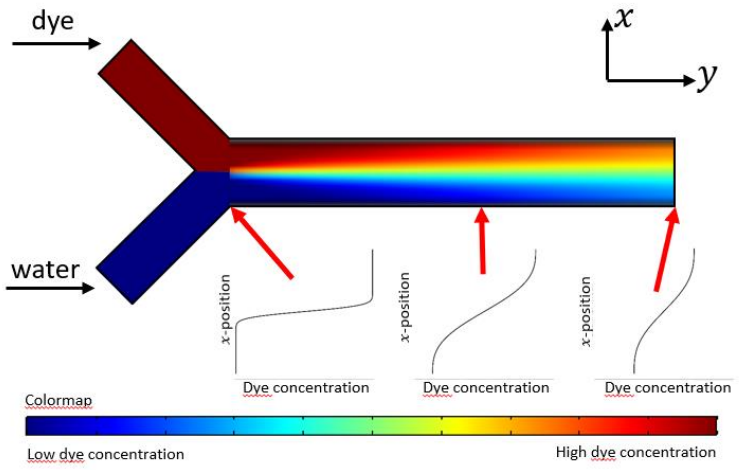


Figure 18. Diagram of a typical distribution of concentrations in a micro-channel in the absence of any specific mixing device. The bottom graph shows the distribution of concentrations as a function of the coordinate perpendicular to the channel wall (y) in several sections of the channel.

The concentration at the outlet of this type of mixer is directly related to the concentrations and flow rates of the related channels. Assuming that channel 1 has a concentration C_1 and a flow Q_1 and joins channel 2 with a concentration C_2 of the same molecule and a flow Q_2, the average concentration C at the outlet of the channel is:

$$C = \frac{Q_1}{Q_1 + Q_2} \cdot C_1 + \frac{Q_2}{Q_1 + Q_2} \cdot C_2 \qquad 23$$

Therefore, the concentration at the outlet of a mixer can be driven by the inlet flow rates, which can themselves be driven by the applied pressures or by the hydraulic resistances of the upstream circuits. Kwang *et al*. modulated this last parameter to achieve their dilution system. They established an equivalent electrical model of the microfluidic circuit, and then used CAD to calculate the hydraulic resistances required for each section of the microfluidic channel to obtain the desired flow rates.

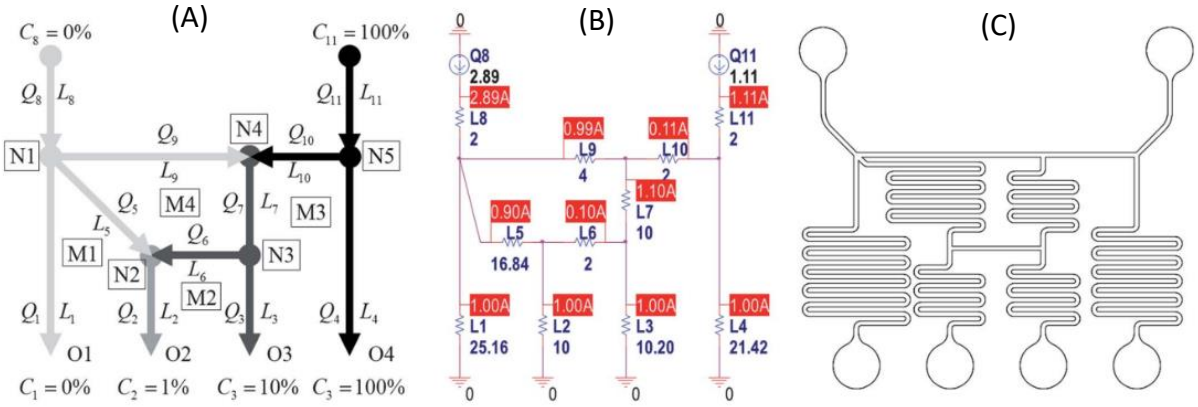


Figure 19. (A) Illustration of a typical series dilution microfuidic network generating a $\log_{10}$ profile of a sample concentration. (B) PSpice simulation results. (C) Arrangement of the mask resulting from the network.

Once the resistance values were established, the circuit was set up by adjusting the length of the channels. As seen in Equation 11, if the section of the channel is unchanged, R_H ∝ L. Here some desired lengths are directly related to the mixture length when dilution occurs. We sought to have four outlets with equal flow and outlet concentrations, from left to right 0%, 1%, 10% and 100%.

Let us take the path leading to outlet O3 in Figure 19 as an example. We wanted a concentration of 10% of the product and a flow rate of 1 µL / min at the outlet. Node N4 was the intersection of two channels, one containing no molecule and the other with a concentration of 100%. We wanted a concentration of 10% at the outlet. Based on Equation 22:

$$\frac{Q_{10}}{Q_9 + Q_{10}} = 0.1$$

and $Q_7 = Q_9 + Q_{10}$, hence $Q_7 = 10 \cdot Q_{10}$

For equivalent reasons, at node N2 we wanted:

$$\frac{Q_6}{Q_5 + Q_6} = 0.1$$

and $Q_2 = 10 \cdot Q_5$. These considerations made it possible to adjust the values of the hydraulic resistances of all the channels of the circuit. In addition, we also wanted to ensure that $Q_1 = Q_2 = Q_3 = Q_4 = 1$. According to Kirchhoff's laws, we wrote:

$$Q_{11} = Q_4 + Q_{10} = Q_4 + \frac{Q_7}{10}$$

But $Q_7 = Q_6 + Q_3$, so:

$$Q_{11} = Q_4 + \frac{Q_3}{10} + \frac{Q_6}{10} = Q_4 + \frac{Q_3}{10} + \frac{Q_2}{100} = 1.11$$

This made it possible to size flow Q_{11}. In the same way:

$$Q_8 = Q_1 + Q_5 + Q_9 = 1 + \frac{9 \cdot Q_2}{10} + \frac{9 \cdot Q_7}{10} = \frac{19}{10} + \frac{9}{10} \cdot (Q_6 + Q_3) = \frac{28}{10} + \frac{9}{100} \cdot Q_2 = 2.89$$

Thus, by modifying the resistances and with a stable pressure difference, the flow rate changed in each of the sections of the circuit. By properly adapting the resistances, we obtained the desired concentrations at the output.

2.2.3. CAD tools for microfluidic circuits

Although potentially very useful, the automation of microfluidic circuit design has received very little attention in the literature. While a few examples of CAD tools for digital microfluidics can be found [3] [71], examples are rarer for continuous microfluidics.

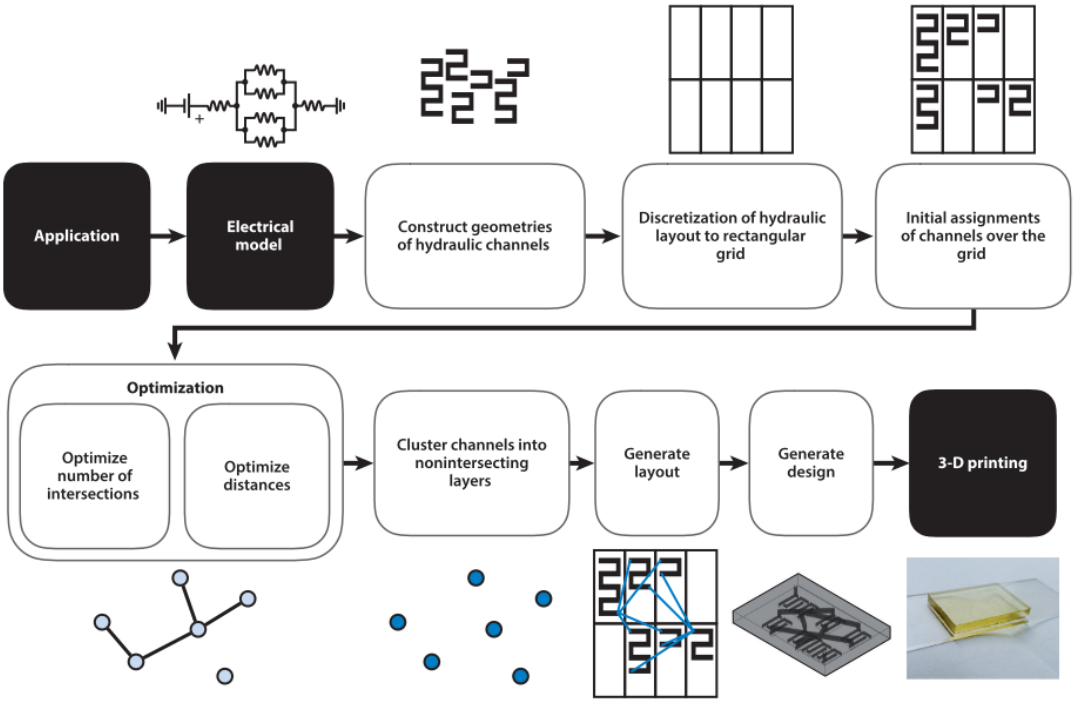


Figure 20. Diagram of the design flow used to create a 3D printing file for microfluidics.

Elishai Ezra Tsur [72] developed a complete CAD tool based around this microfluidic / electronic analogy. The entry point was an equivalent electrical model of the microfluidic circuit to be produced. This involved a first step of manual design to build a model in line with the intended application. The rest of the process was automated. The tool automatically created a layout of microfluidic channels whose equivalent electronic diagram corresponded to the one provided at the inlet and generated design files that could be directly used on a 3D printer (Figure 20). The principle of the tool was based on cutting the microfluidic circuit into sections of different lengths and assembling them to obtain the desired resistances between each node. Then, the different channels were linked together in accordance with the equivalent diagram, and placed on a surface by an optimization algorithm similar to the placement and routing algorithms used in electronics.

This end use clearly shows the enthusiasm behind the design of automated microfluidic circuits through the use of design tools already present for electronics.

3. Limitations of the use of high-abstraction simulation for labs on chips

As mentioned in section 1, the simulation of microfluidic circuits is one of the three fundamental parts of CAD of labs on chips, along with the simulation of biological reactions and the simulation of biosensors. However, the usual techniques described in this chapter are not fully satisfactory for several reasons.

The models initially described here are only valid for microfluidic systems made up of a single type of molecule: as seen in the example given in section 3.b, the calculation of the concentration of a molecule is only a consequence of the concentrations present at the inlet and of the flow rates. However, they do not play a direct role in electrical analogy.

In addition, calculations are made under the assumption of homogeneous mixtures, which is not necessarily the case in a microfluidic system [73].

4. Microfluidic circuit modeling using analogy

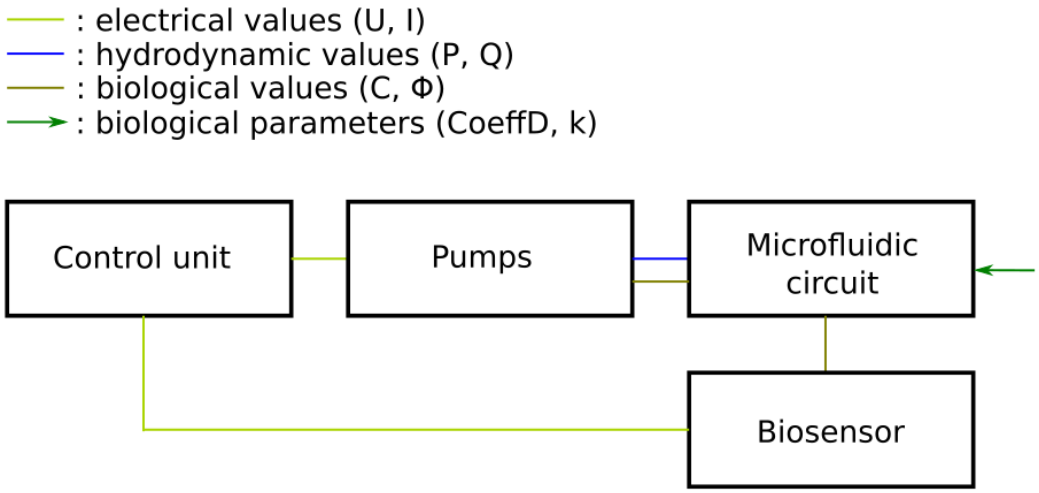


Figure 21. Detailed diagram of the different components. U = tension (V), I = intensity (A), P = pressure difference (Pa), Q = flow rate (m^3/s), C = concentration (mol/L), Φ = flow rate of molecules (mol/L/s), CoeffD = diffusion coefficient (m^2/s), and k = speed of the reaction (s^{-1}).

To overcome these limitations, we considered the development of a simulation environment in connection with the CAD tools used in electronics and capable of integrating these two issues. The basic idea of our simulator is schematically represented in Figure 21, based on a simplified example. An electronic model of a controller generates electrical signals to control a pump. The pump controls the flow rates of molecules (and therefore the concentrations of molecules) at the input of a microfluidic circuit. One or more biochemical reactions occur within the microfluidic circuit, which results in the generation of a biological signal. This signal is picked up by a biosensor (and more particularly by the transducer) which sends an electrical signal back to the controller.

From a simulation point of view, several models with different types of interfaces (electrical, biological, hydraulic) are available, all expressed in the same language, all based on equivalent electrical circuits and all interconnected by Kirchhoff's laws. This allows for a single simulation environment for the entire lab-on-a-chip. As for microfluidics, we obviously based ourselves

on the analogy presented above, but also went beyond by proposing i) a compact model of a microfluidic mixer to take incomplete mixtures of several molecules into account, and ii) microfluidic components whose concentrations are not homogeneous and / or in which the spatial localization of the molecules plays a determining role in the response of the system. These two new models are described in Chapters 3 and 4, respectively.

5. Conclusion

In this chapter, we presented the fluid mechanics equations that describe the behavior of the fluid in laminar microfluidics. Then, we detailed the principle of analogy between fluid mechanics in microscopic media and electronics.

We also saw that this principle was already widely studied for the design of microfluidic circuits. We will use this modeling for the simple parts of our lab-on-a-chip design tool.

The next step in the development of our lab-on-a-chip simulator will have to be the study of the mixture of two molecular species, as many labs on chips mix one or more reagents. We are going to study this mixture for laminar microfluidics, the limitations of electrical analogy modeling, and our response to this limitation.

Chapter 3

Microfluidic simulation of a mixer: a compact model

In the present chapter, we develop an important part of a lab-on-a-chip, namely the mixer. First, we show how the analogy between microfluidics and electronics allows us to model mixers.

Then, we will see that the modeling freedom is reduced, especially when we want a mixture of two solutions containing different molecules each creating a concentration gradient within a mixing channel.

To address this issue, we developed a compact model compatible with high-abstraction analogy modeling to improve specific parts of the lab-on-a-chip simulation.

We are going to explain the physical specificities of the microfluidic mixture and the resulting equations, and focus on the differential equation describing our problem.

The construction of the compact model, based on the solving of this same differential equation, will be detailed. A comparison of our model will also be made with our reference digital simulation software – COMSOL Multiphysics®.

1. Modeling a mixer by electrical analogy

1.1. A model of a perfect mixer: an example

To represent the problem of mixing, we are going to present the electrical diagram of a perfect mixer, whose principle is based on the work presented in the previous chapter. We added the flows of molecules to complete our model with current sources so as to simulate them.

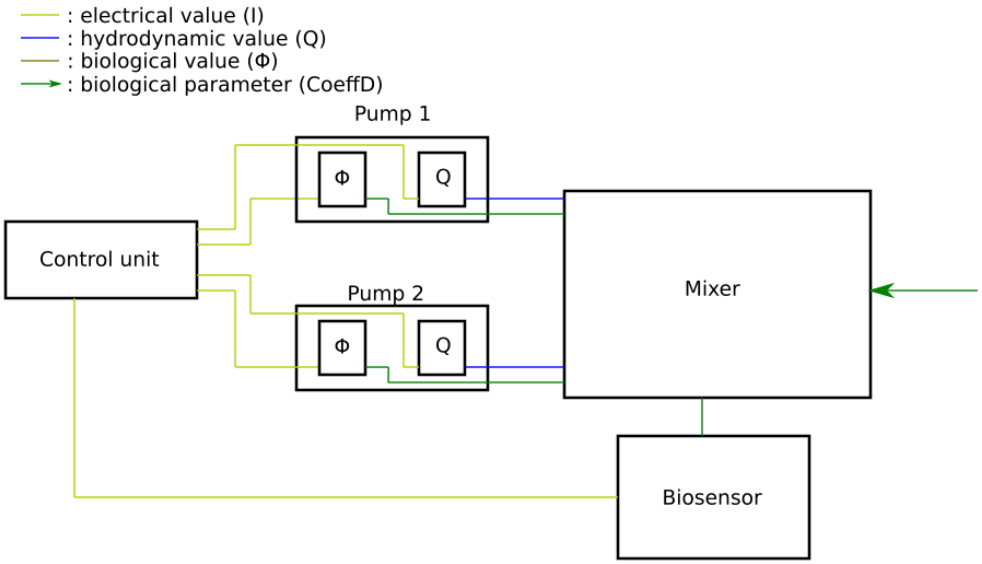


Figure 22. Block diagram of a perfect microfluidic mixer.

For this, the microfluidic circuit block was split up to provide the hydrodynamic and biological values, as showed in Figure 22 which represents a block diagram of a perfect microfluidic mixer. The controller generates electrical signals that control the pumps. In our example, the pumps are peristaltic ones that generate a flow. These pumps are also directly connected to a reservoir of molecules of variable molar quantity (experimentally, one can imagine a container connected to two inlets). One of the inlets lets in a concentrated solution of molecules, which increases the concentration in the container. The other inlet lets in a solvent that dilutes the concentration in the container. These pumps are connected to the mixer, is a more or less complex series of resistance networks depending on the application. For our simple case, we mixed a solution of molecule A from pump 1 and a solution of molecule B from pump 2.

As demonstrated in Chapter 2, we can model a perfect mixer with electronic components. To this end, we have to couple two sub-circuits: one of them calculates the hydrodynamic values, and the other calculates the biological values.

Figure 23 shows the conversion of Figure 22 into an electrical diagram. The links between the different nodes are made simply by connecting the two nodes with a wire. We are now going to further detail the mixer from the two hydrodynamic and biological points of view.

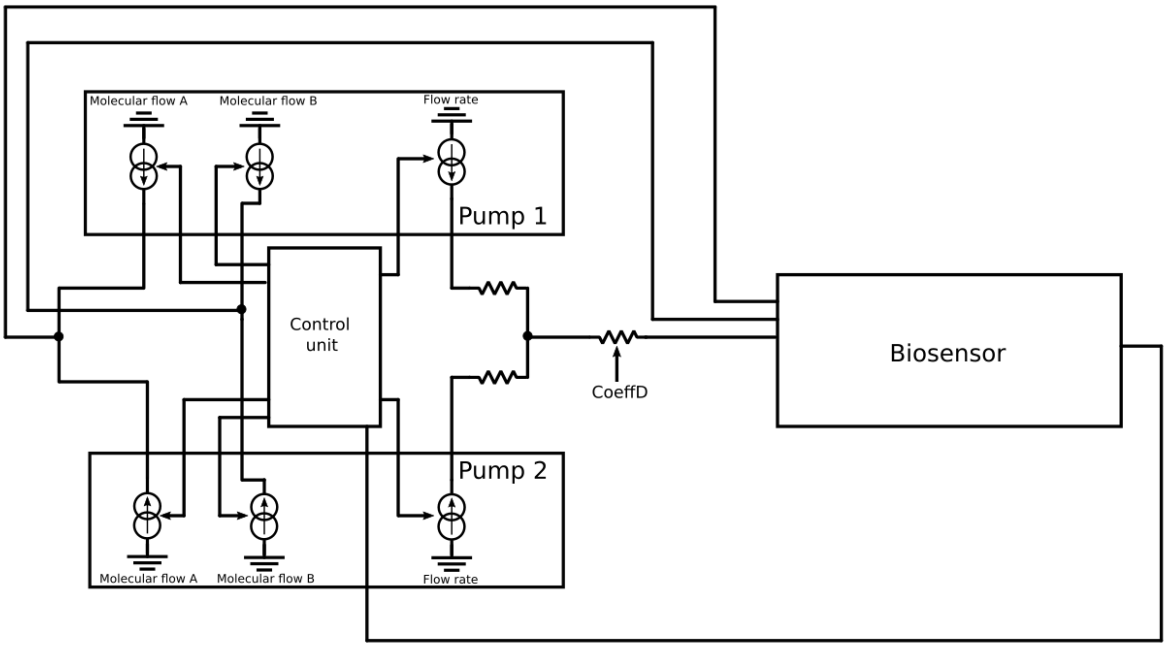


Figure 23. Equivalent electrical diagram of a block diagram of a simple microfluidic circuit with two microfluidic channels that meet and perfectly mix the reactants and then head toward a microsensor.

1.1.1. The mixer: fluid mechanic aspect

For this section, we need to know the size of the channel allowing the total mixing of the two species. As a reminder, the equation of the diffusion length is the following one:

$$L_D = \frac{Q}{D_D} \qquad 22$$

Where L_D is the diffusion length (m), Q the flow rate (m^3/s), and D_D the diffusion coefficient (m^2/s). Two species are mixed, with potentially different diffusion coefficients. For the two species to be mixed correctly, the one that takes the longest time to be mixed is chosen as the reference. The species with the lowest D_D determines the length needed for the mixer channel.

The lower scattering coefficient directly influences the resistance value of the mixer (Figure 23).

1.1.2. The mixer: biological aspect

The management of biological flows is simple in the present case. There are two streams of molecules, one for each pump. These flows are represented by separate current sources for molecules A and B. These different sources simply add up before reaching the biosensor (Figure 23).

1.2. Issue

In the analogy presented here, we only model fluids that are homogeneous or considered homogeneous. When we mix two products, a concentration gradient forms across the width of the channel when the diffusion length is not reached. The microfluidic / electronic analogy no longer holds since the electron flows are not differentiated when two branches of a channel

meet. To be able to model this phenomenon, we must complete the previous model to model this concentration gradient across the width of the channel.

1.3. Imperfect mixer case

The high-abstraction simulation based on the microfluidic / electronic analogy described in the previous chapter can pose problems because this technique is called into question when we want to know the concentration profiles across the width of a channel of a microfluidic mixer. As showed in Figure 24 extracted from an article by N. Scott Lynn Jr. [74], a microfluidic circuit allows the mixing of two products subsequently detected at the level of a narrowing in the center of the channel. A complete mixture of the two products would be useless because the analysis is performed only at a given point in the center of the channel, across its width.

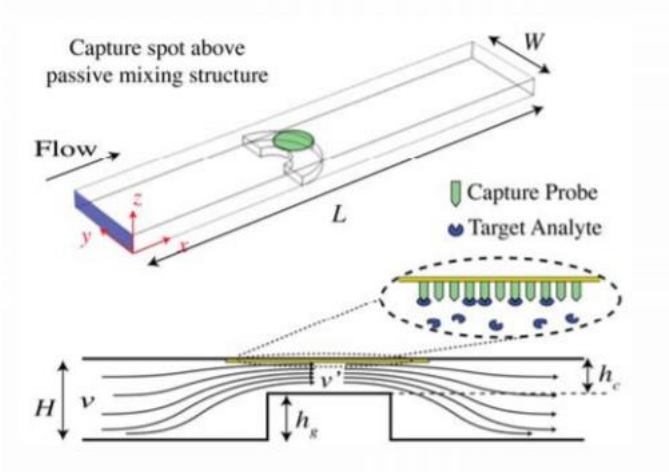


Figure 24. Schematic diagram of the microfluidic circuit of a passive mixer allowing better detection by a biosensor.

Pour répondre à cette problématique, nous avons développé un modèle compact de mélangeur imparfait qui aurait comme paramètres la géométrie du canal et la constante de diffusion des molécules mélangées, et en entrée-sorties les quantités hydrodynamiques (pressions et débit) complétées par des qualités traduisant les profils de concentration pour les différentes espèces à mélanger. To solve this issue, we developed a compact model of an imperfect mixer using the channel geometry and the diffusion constant of the mixed molecules as parameters. Moreover, at the inlet and outlet, the hydrodynamic quantities (pressures and flow rates) were supplemented by parameters of the concentrations profiles of the species to be mixed .

Ce modèle compact repose sur une solution analytique des équations de diffusion, sous certaines hypothèses, et en utilisant une décomposition en séries de Fourier. This compact model is based on analytical solving of the diffusion equations, under certain hypotheses, and using Fourier series decomposition.

2. Model construction

2.1. Advection-diffusion equation in the channel

The structure of a passive mixer is described in paragraph 2.2.2 of the previous chapter. The mixer is a cylindrical straight channel in which the two fluids flow in a laminar fashion. The

mixing between the two fluids is entirely governed by diffusion phenomena in the transverse direction to the flow.

The concentration of a molecule in the mixer is described by the advection-diffusion equation that follows the laws of mass conservation for an elementary volume [34]. We first considered a channel of rectangular section and the x and y axes representing the transverse and longitudinal directions of the channel, respectively (Figure 25). The junction introduced inhomogeneity of the concentrations in the x direction. The length of the channel was L and its width W. The origin of the coordinate system was located in the upper left corner of the channel. To simplify the problem, we considered that concentrations were homogeneous in the third direction, which is a reasonable assumption if the thicknesses of the channels before and after the junction were the same. In cylindrical geometry, the x axis is considered as the radius, so that the advection-diffusion equation in its full form is written:

$$\frac{\partial \mathrm{C}}{\partial \mathrm{t}} = \mathrm{D}\Delta \mathrm{C} - \boldsymbol{u}(\mathrm{x},\mathrm{y},\mathrm{t}) \cdot \nabla \mathrm{C} + \mathrm{s}(\mathrm{x},\mathrm{y},\mathrm{t}) \qquad 24$$

where $C(x,y,t)$ is the concentration of a given molecule (mol.L^{-1}), D is the diffusion coefficient of this molecule (m².s^{-1}), $\boldsymbol{u}(x,y,t)$ is the fluid velocity field (m.s^{-1}), $s(x,y,t)$ (mol.L^{-1}.s^{-1}) is a local source (or a loss, depending on its sign) of molecules (for example due to a chemical reaction occurring in the channel), Δ is the Laplacian operator, and ∇C is the gradient of C.

2.2. Assumption and simplification of the equation

In a straight channel, due to laminar flow, the transverse component (*i.e.*, along the x axis, Figure 25) of the velocity field can be considered zero. In addition, we considered the system to be in steady state. The velocity field was not time dependent and was the same regardless of the y position along the axis of the channel. In fact, the velocity field had only one component along the y axis, and this value only depended on x ($\boldsymbol{u}(x,y,t) = u_y(x)$). Finally, we considered that there was no source of molecules (for example, no chemical reaction occurring in the channel), so that $s(x,y,t) = 0$. As a result, the advection-diffusion equation was simplified as follows:

$$\frac{\partial \mathrm{C}}{\partial \mathrm{t}} = \mathrm{D} \cdot \left(\frac{\partial^2 \mathrm{C}}{\partial \mathrm{x}^2} + \frac{\partial^2 \mathrm{C}}{\partial \mathrm{y}^2} \right) - \mathrm{u_y}(\mathrm{x}) \cdot \frac{\partial \mathrm{C}}{\partial \mathrm{y}} \qquad 25$$

Finally, we also assumed that diffusion along the channel was negligible compared to transport due to advection. This hypothesis was validated *a posteriori* by digital simulations (see Section 3). According to these assumptions, equation 25 became:

$$\frac{\partial \mathrm{C}}{\partial \mathrm{t}} = \mathrm{D} \cdot \frac{\partial^2 \mathrm{C}}{\partial \mathrm{x}^2} - \mathrm{u_y}(\mathrm{x}) \cdot \frac{\partial \mathrm{C}}{\partial \mathrm{y}} \qquad 26$$

In steady state, this equation was written:

$$\mathrm{D} \cdot \frac{\partial^2 \mathrm{C}}{\partial \mathrm{x}^2} - \mathrm{u_y}(\mathrm{x}) \cdot \frac{\partial \mathrm{C}}{\partial \mathrm{y}} = 0 \qquad 27$$

2.3. Reference change

Let us now place ourselves within a frame of reference following a diffusion front, *i.e.,* a frame of reference animated by a constant speed v_y along the y axis. The coordinates in the new frame of reference (x', y') were linked to the coordinates in the terrestrial frame of reference (x, y) by the relations:

$$\begin{cases} \mathrm{x}' = \mathrm{x} \\ \mathrm{y}' = \mathrm{y} + \mathrm{v_y} \cdot \mathrm{t} \end{cases} \quad 28$$

In the new frame of reference, equation 27 read:

$$\mathrm{D}\frac{\partial^2 \mathrm{C}}{\partial \mathrm{x}'^2} - \mathrm{v_y} \cdot \frac{\partial \mathrm{y}'}{\partial \mathrm{y}} \cdot \frac{\partial \mathrm{t}}{\partial \mathrm{y}'} \cdot \frac{\partial \mathrm{C}}{\partial \mathrm{t}} = 0 \quad 29$$

From equation 28, $\frac{\partial y'}{\partial y} = 1$ and $\frac{\partial y\prime}{\partial t} = v_y$. Therefore, equation 29 was rewritten as follows:

$$\frac{\partial \mathrm{C}}{\partial \mathrm{t}} = \mathrm{D} \cdot \frac{\partial^2 \mathrm{C}}{\partial \mathrm{x}'^2} \quad 30$$

To simplify the writing, we worked within this new frame of reference and gave up apostrophes thereafter.

2.4. Solving the simplified advection-diffusion equation

As seen earlier, our frame of reference followed the molecule front and there was no loss or creation of material along the channel, so we considered that the concentration did not depend on y within this frame of reference. Using equation 30, we ended up solving equation 31 as follows:

$$\frac{\partial \mathrm{C(x,t)}}{\partial \mathrm{t}} - \mathrm{D}\frac{\partial^2 \mathrm{C(x,t)}}{\partial \mathrm{x}^2} = 0 \quad 31$$

We sought a solution to this equation by using the method of separation of variables [75], namely:

$$\mathrm{C(x,t)} = \mathrm{C_X(x)} \cdot \mathrm{C_T(t)} \quad 32$$

Equation 31 became:

$$\mathrm{C_X(x)}\frac{\mathrm{dC_T(t)}}{\mathrm{dt}} = \mathrm{DC_T(t)}\frac{\mathrm{d^2 C_X(x)}}{\mathrm{dx}^2} \quad 33$$

That we rewrote as:

$$\frac{1}{D \cdot C_T(t)}\frac{dC_T(t)}{dt} = \frac{1}{C_X(x)}\frac{d^2 C_X(x)}{dx^2}$$

This equation had to be valid for all values of x and t. However, the term on the left side of the equation did not depend on x and the term on the right side did not depend on t. Therefore, the only way to respect this equation whatever x and t was for the left and right terms to be constant. We called this constant α.

2.4.1. Calculation of the temporal term

$C_T(t)$ was the solution of the differential equation:

$$-\alpha \cdot D \cdot C_T(t) = 0 \quad (34)$$

This was a first-order equation whose general solution is an exponential:

$$C_T(t) = A \cdot e^{D \cdot \alpha \cdot t} \quad (35)$$

Where A is a constant of integration. We searched a solution for all $t > 0$. Then, it was essential that α be negative. If this had not been the case, the concentration would have tended towards infinity for a time that would have tended towards infinity. Therefore, constant α was rewritten $-\lambda^2$ with λ any real number. By introducing this new constant, the temporal term became:

$$C_T(t) = A \cdot e^{-\lambda^2 \cdot D \cdot t} \quad (36)$$

2.4.2. Development of the spatial aspect

The spatial term $C_X(x)$ was a solution of the differential equation:

$$\frac{d^2 C_X(x)}{dx^2} \mp \lambda^2 C_X(x) = 0 \quad (37)$$

For this term, we looked for a solution to this equation in the form of a Fourier series. However, expressing $C_X(x)$ as a Fourier series implies that $C_X(x)$ was periodic, which was not true in our case (Figure 18). To overcome this problem, we "periodized" the problem along the x axis by considering a series of parallel channels, each channel being a mirror of the previous one. Thus, we replaced the function $C_X(x)$ which was only defined in the interval $[0\ ;\ W]$ by a periodic function of period $2 \cdot W$ $C_X^*(x)$ defined in$[-\infty; +\infty]$. This new function $C_X^*(x)$ was decomposed into Fourier series:

$$C_X^*(x) = \sum_{n=0}^{\infty} C_n(x) = \sum_{n=0}^{\infty} A_n \cdot \sin(\lambda_n \cdot x) + B_n \cdot \cos(\lambda_n \cdot x)$$

$$C_{X,n}(x) = A_n \cdot sin(\lambda_n \cdot x) + B_n \cdot cos(\lambda_n \cdot x) \quad (38)$$

Where $C_n(x)$ is the nth term of the Fourier series, *i.e.,* the solution of differential equation (15) with $\lambda = \lambda_n$. A_n, and B_n are the nth Fourier coefficients and a constant associated with the nth term of the series. For the concentrations equal to zero on the walls of the channel, the boundary conditions for calculating A_n and B_n were $\frac{\partial C}{\partial x} = 0$ for $x = 0$ and $x = W$. This led to:

$$\begin{cases} \frac{\partial C}{\partial x}(0) = A_n = 0 \\ \frac{\partial C}{\partial x}(W) = B_n \cdot cos(\lambda_n \cdot W) = 0 \end{cases} \quad (39)$$

forcing the values of λ_n to:

$$\lambda_n = \frac{n \cdot \pi}{W} \quad (40)$$

Therefore, the complete equation of the spatial solution was:

$$C_X(x) = \sum_{n=0}^{\infty} B_n \cdot cos\left(\frac{n \cdot \pi \cdot x}{W}\right) \quad (41)$$

Where A_n are constants to be determined, defining the concentration profile.

2.4.3. General solution

The general solution of equation 29 was obtained by associating each term $C_{X,n}$ of the Fourier series with its time component $C_{T,n}$, as follows:

$$C(x,t) = \sum_{n=0}^{\infty} B_n \cdot cos\left(\frac{n \cdot \pi \cdot x}{W}\right) \cdot e^{-D*t*\left(\frac{n*\pi}{W}\right)^2} \quad (42)$$

2.5. Analytical expression

To calculate the Fourier coefficients, we considered equation 42 at $t = 0$. Based on the definition of the Fourier series, coefficient B_n was expressed as follows:

$$B_n = \frac{1}{W}\int_{-W}^{W} C(x,0)\, cos\,(\pi * n * \frac{x}{W})dx \quad (43)$$

Therefore, coefficient B_n depended directly on $C(x,0)$, which corresponded to the concentration profile at the channel inlet at $t = 0$. However, there was no analytical expression for $C(x,0)$, so we used a finite element simulator – COMSOL Multiphysics® [44] – to find an empirical equation for $C(x,0)$.

2.6. Empirical model of the input concentration profile

The geometry simulated under COMSOL is shown in Figure 25. To solve this simulation, we used the "laminar fluid mechanics" and "mixture of chemical species" modules. These modules allowed us to import the physics needed to simulate the fluid velocity fields and the transport of molecules (diffusion and convection), respectively.

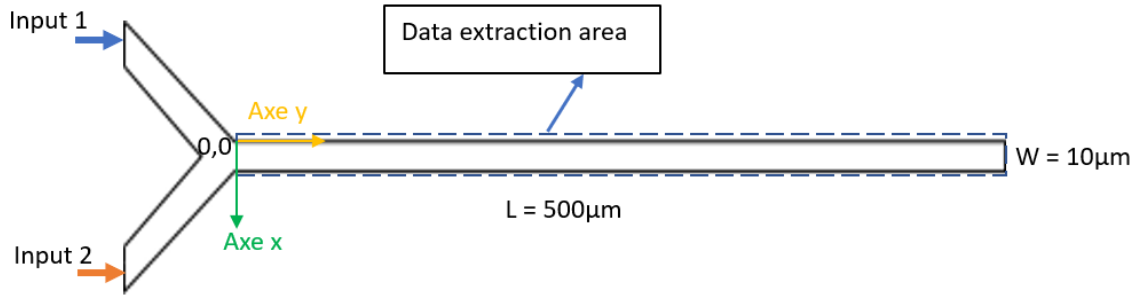


Figure 25. Diagram representing the COMSOL simulation of a passive microfluidic mixer.

All the simulations showed in this section were carried out based on this model, with input flow rates 1 and 2 varying between 0.5 nL/s and 2.5 nL/s. Entry 1 corresponded to pure solvent, while entry 2 contained the molecule whose concentration profile we wanted to know. The diffusion coefficient varied between 10^{-8} m²/s and 10^{-10} m²/s, and its concentration between 1 mM and 2 mM.

COMSOL simulations showed that whatever the parameters of the system, the concentration profile took the form showed in Figure 26. This profile is classically modeled by a sigmoid [47]which depended on four parameters:

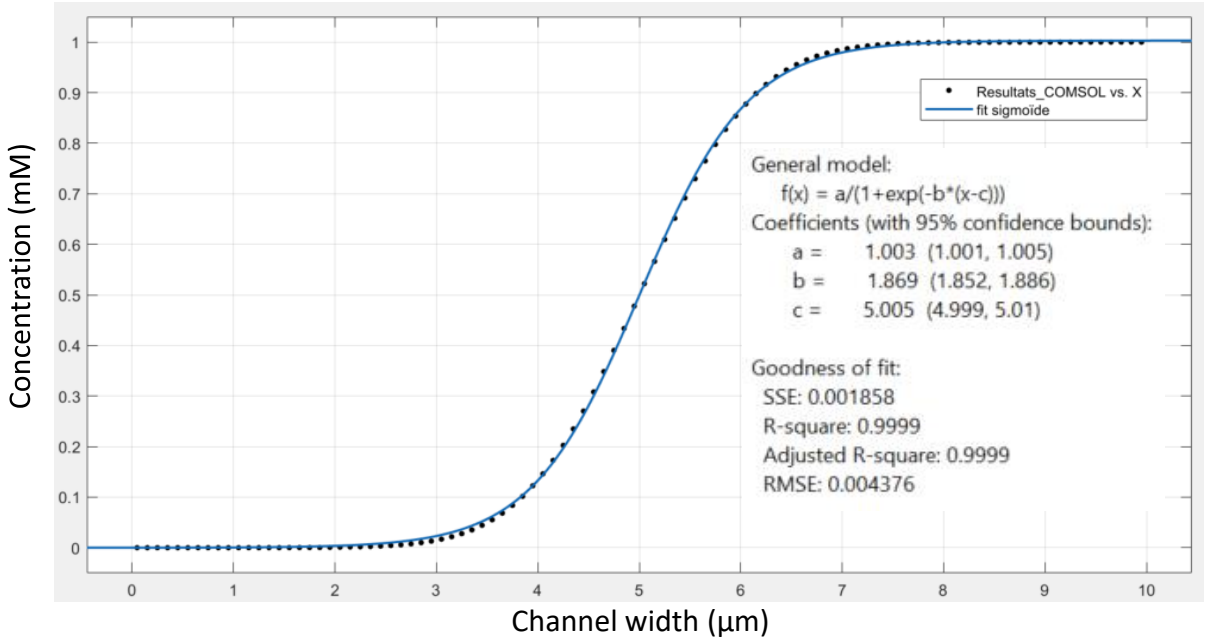


Figure 26.Concentration profile across the inlet of a microfluidic mixer channel. Parameter d is not mentioned because it is equal to 0 at the channel inlet. Data obtained using COMSOL, and data fit obtained using MATLAB.

$$C(x) = \frac{a}{1 + e^{-b(x-c)}} + d \qquad 44$$

where a, b, c and d are four parameters to be adjusted, a is the amplitude of the sigmoid, d its low value, c is the abscissa of the inflection point, and b is representative of the slope at the inflection point.

We used MATLAB to fit the sigmoid parameters to the curves provided by COMSOL. Under the conditions of Figure 27, we obtained a good fit with a root mean square deviation of 0.0044.

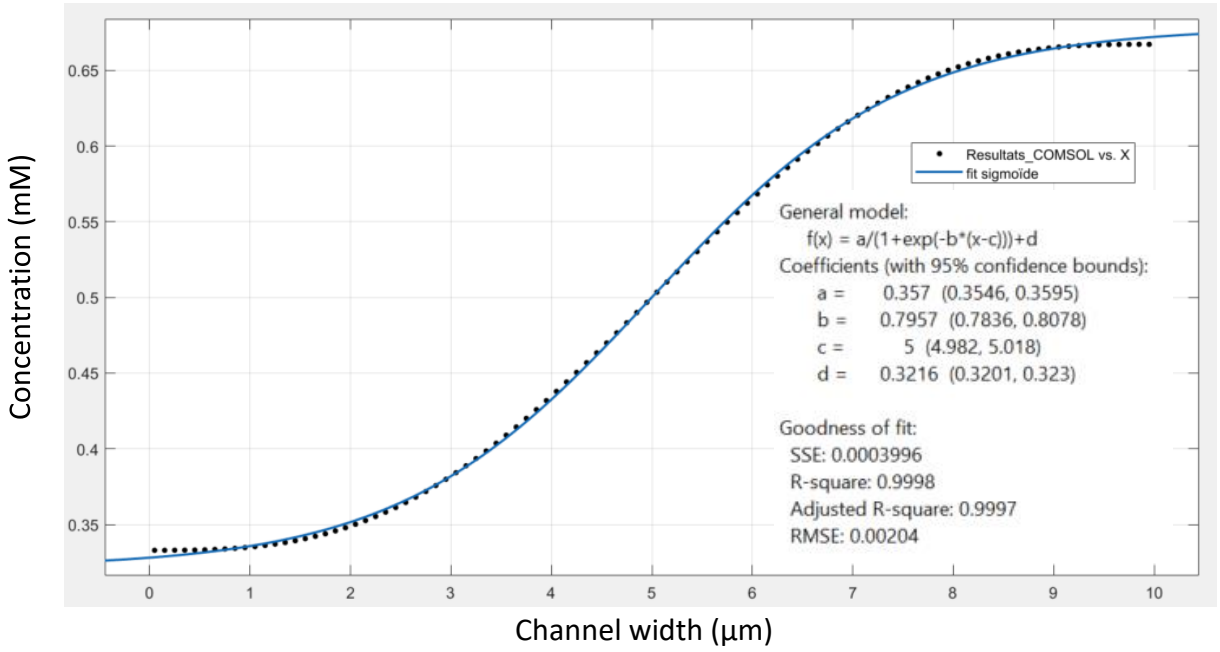


Figure 27. Concentration profile across the outlet of a microfluidic mixer channel. Data obtained using COMSOL, and data fit obtained using MATLAB.

We applied the same procedure for several flow rates and several initial concentrations. The extracted parameters showed that we could predict the values of a, b, c, and d as a function of the initial values of $Q_1$and Q_2 and of the concentrations of molecules C_1 and C_2at inlets 1 and 2 using the following equations:

$$\begin{cases} a = C_2 - C_1 \\ b = k_1 \cdot (Q_1 + Q_2) + k_2 \\ c = \dfrac{Q_1}{Q_1 + Q_2} \cdot W \\ d = C_1 \end{cases} \qquad 45$$

where k_1 and k_2 are two adjustment parameters that depend on the geometry of the Y junction and on the homogeneity of the fluid. They had to be extracted from the COMSOL simulation for each junction type. In our case, k_1 = 0.605 s.µm^{-4} and k_2 = 1.3 µm^{-1}.

The maximum difference between the sigmoid and the actual profile was 0.01 mM, which represents a relative error of about 2%.

We also took advantage of these COMSOL simulations to verify that the concentration profile maintained a sigmoid shape throughout the channel. The profile tended to flatten out due to diffusion throughout the channel, but the use of the sigmoid profile remained valid even at the outlet of the channel (Figure 27).

2.7. Validity of Fourier series decomposition.

The Fourier coefficients were calculated by digital integration in Python based on equation 43. The result of Figure 26 and its Fourier series decomposition up to order 20 is showed in Figure 28.

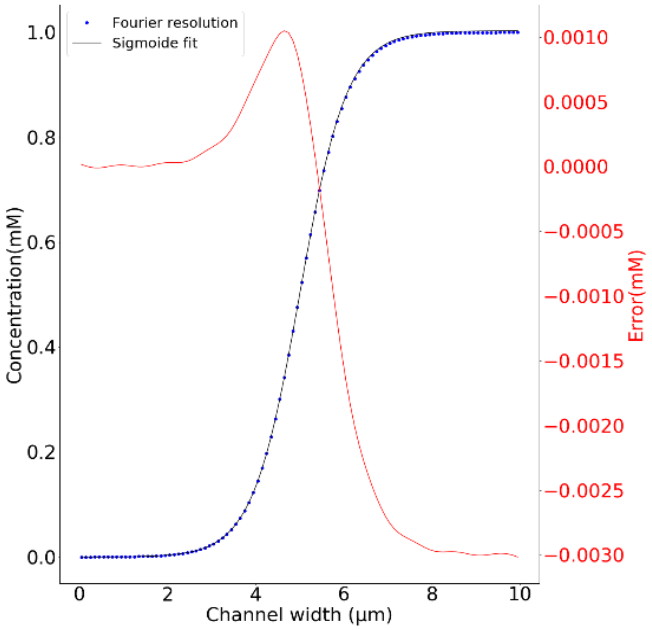


Figure 28. Fourier decomposition results obtained in Python vs. sigmoidal equation: distribution of concentrations at the inlet of the channel of the microfluidic mixer.

In practice, we only succeeded in calculating a finite number N of terms in the Fourier serie. It seemed important to us to estimate the minimum order N_{min} down to which to go to have a reasonable deviation. Table 1 summarizes the quadratic differences measured for i) the profile obtained by digital simulation under COMSOL and ii) the profile obtained with the sigmoid curve, each compared with the profile obtained after decomposition in Fourier series, as a function of N.

The approximation of the concentration profile simulated by COMSOL by a Fourier series was the sum of 2 errors, namely the profile approximation by a sigmoid and the sigmoid approximation by a Fourier series. We defined N_{min} as the value beyond which the second error became negligible compared with first one. Based on the data in Table 1, N_{min} was set to 5.

Table 1. Quadratic error in the channel for a Fourier series of order N.

N-order	1	3	5	7	9+
MSE COMSOL/Fourier (%)	2.70	0.418	0.330	0.317	>0.311
MSE Sigmoid/Fourier (%)	2.76	0.835	0.303	0.116	>0.05

3. Validation of the working hypotheses

The established model is based on the fact that diffusion is negligible compared with advection in the direction of the channel (see section 2.2). In order to validate this hypothesis, we compared the results obtained in COMSOL simulations with i) a case of isotropic diffusion (standard case), and ii) a case of anisotropic diffusion obtained by canceling diffusion in the axis of the channel. The other characteristics of the simulation were the same as before. The

results are showed in Figure 29 for different pairs of inlet flow rates. We observed a maximum error of 1.2% across all the tests, which confirms the minor role played by diffusion along the axis of the channel in the simulated phenomenon.

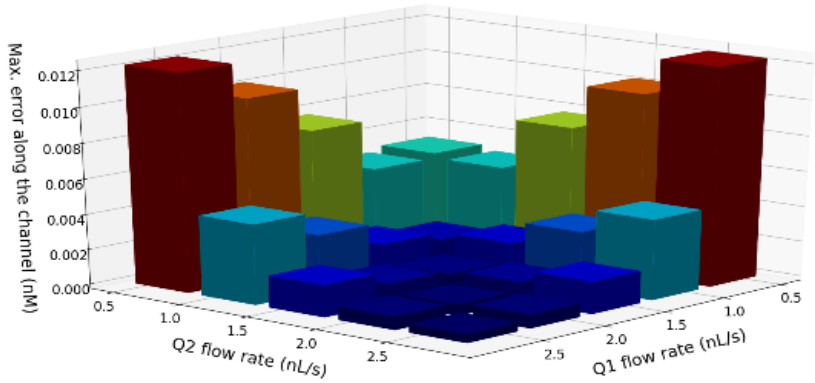


Figure 29. Validation of a hypothesis: maximum errors for different flow rates. In this 3D chart, the flow rates Q1 and Q2 are along axes x and y, while the maximum mean quadratic errors possibly found throughout the whole channel are along axis z.

4. Validation of the compact model

The compact mixer model was written in Python. The concentration profiles in the transverse axis of the channel obtained with COMSOL and the compact model were compared at different positions in the channel and for different configuration parameters (Table 2).

An example of the distribution of concentrations for each of the 6 configurations is showed in Figure 30. The results were qualitatively equivalent to those found in the literature [69] [70].

Table 2. Recapitulative table of the simulation experiments.

Experiment #	Flow rate Q1 (nL/s)	Flow rate Q2 (nL/s)	Ratio R = Q1/Q2	Channel flow rate Q = Q1+Q2 (nL/s)
1	0.5	0.5	R = 1	Q = 1
2	2.5	2.5	R = 1	Q = 5
3	1	0.5	R = 0.5	Q = 1.5
4	0.5	1	R = 2	Q = 1.5
5	2.5	0.5	R = 0.2	Q = 3
6	0.5	2.5	R = 5	Q = 3

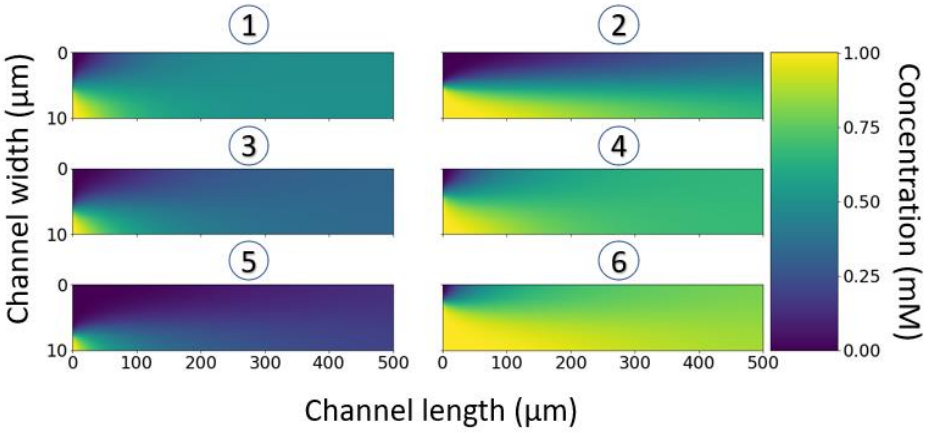


Figure 30. Results of experiments 1 to 6 aimed at solving Fourier series in Python. The channel width on the Y-axis is at the 1:10 scale for better readability. The ladder on the right indicates the concentrations in the channel.

Figure 31 shows the measured absolute difference between the simulation results of our model and the COMSOL simulation results for the 6 configurations described in Table 2. The absolute maximum error was approximately 0.07 mM, which represents 7 % of the concentration range inside the channel. It should also be noted that the maximum error occurred at the inlet of the channel. This can be explained by the disturbance of the laminar flow due to the Y junction, which was calculated by COMSOL but not integrated in our model. We also noticed a decreasing error along the channel.

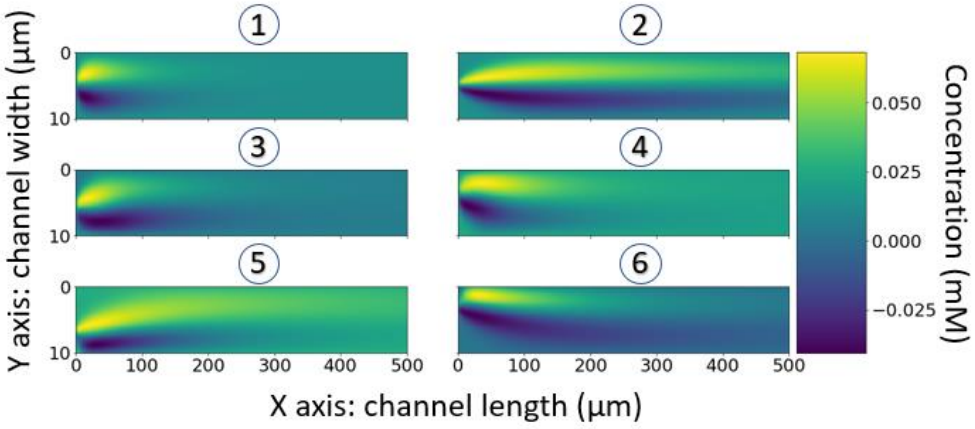


Figure 31. Results of experiments 1 to 6: results of the solving of Fourier decomposition series in Python, minus COMSOL results. The channel width on the Y-axis is at the 1:10 scale for better readability. The ladder on the right indicates the concentrations in the channel.

Figure 32 shows a comparison between the output concentration profile calculated by COMSOL and the one estimated by our compact model for configuration 1. The maximum error was 0.02 mM, which is 4% of the mean concentration at the outlet of the channel. We also calculated the quadratic error for 25 couples of flows along the channel. The square error was acceptable in the channel (the maximum square error was less than 2% throughout the entire channel) and very low at the outlet (<0.3%) (Figure 33).

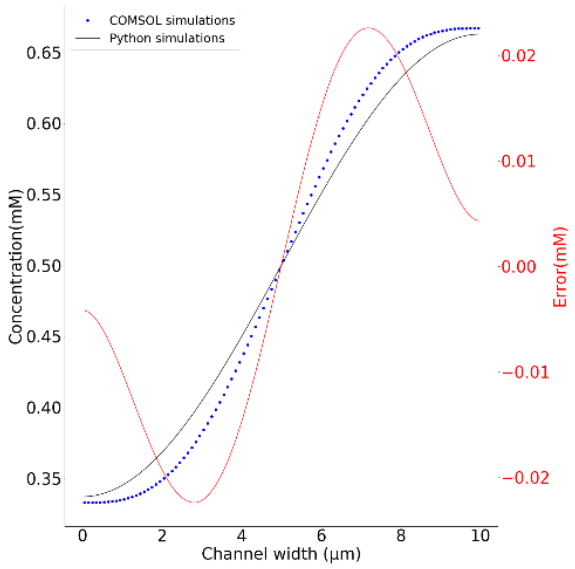


Figure 32. Cross section of the channel outlet at the end of experiment 2. Differences between Python and COMSOL simulations.

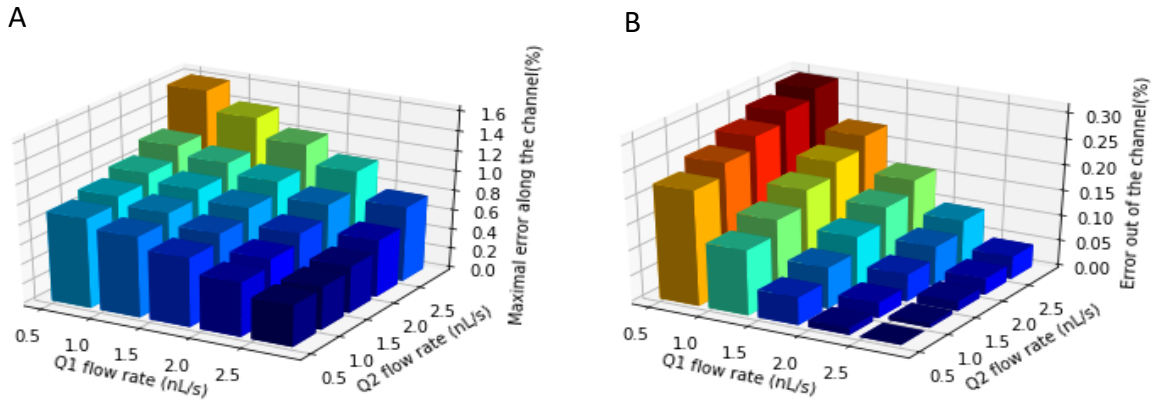


Figure 33. Maximum quadratic error throughout the channel (A) and at the channel outlet (B) for the different flow rate combinations.

5. Writing the compact model

The compact model in Python used curve fitting and digital integration functions of the SciPy Python module that do not naturally exist in Verilog-A or SPICE. Digital integration can be done relatively easily by coding a Verilog-A function based on the trapezoid method. The adjustment function has to be replaced by a manual adjustment process to be developed.

5.1. Calculation of the inflection point

To simplify the extraction of the sigmoid parameters , we rewrote it as follows:

$$C(x) = \frac{2 * (a' - d')}{1 + e^{b(x-c)}} + d' \qquad 46$$

With this shape, c and a' were the coordinates of the inflection point $C(x)$, and d' was the asymptote value of $C(x)$ at the inflection point. By definition, the inflection point corresponded to the maximum of the derivative of $C(x)$ (Figure 34A).

To find the inflection point, we digitally computed $C'(x)$ (the derivative of $C(x)$), and then we performed a Lagrange interpolation with the three highest points of $C'(x)$ (Figure 34B). The Lagrange interpolation is a polynomial of order 2:

$$\mathrm{L(x)} = \mathrm{a_2 \cdot x^2 + a_1 \cdot x + a_0} \quad 47$$

Let $(x_1; y_1)$, $(x_2; y_2)$ and $(x_3; y_3)$ be the three points through which L(x) must pass. By definition of the Lagrange polynomial, the coefficients a_0, a_1 and a_2 can be calculated from these three points, thus:

$$\mathrm{a_2} = \frac{\mathrm{y_1}}{(\mathrm{x_1 - x_2}) \cdot (\mathrm{x_1 - x_3})} + \frac{\mathrm{y_2}}{(\mathrm{x_2 - x_1}) \cdot (\mathrm{x_2 - x_3})} + \frac{\mathrm{y_3}}{(\mathrm{x_3 - x_1}) \cdot (\mathrm{x_3 - x_2})} \quad 48$$

$$\mathrm{a_1} = \frac{-\mathrm{y_1} \cdot (\mathrm{x_2 + x_3})}{(\mathrm{x_1 - x_2}) \cdot (\mathrm{x_1 - x_3})} - \frac{\mathrm{y_2} \cdot (\mathrm{x_1 + x_3})}{(\mathrm{x_2 - x_1}) \cdot (\mathrm{x_2 - x_3})} - \frac{\mathrm{y_3} \cdot (\mathrm{x_1 + x_2})}{(\mathrm{x_3 - x_1}) \cdot (\mathrm{x_3 - x_2})} \quad 49$$

$$\mathrm{a_0} = \frac{\mathrm{y_1 \cdot x_2 \cdot x_3}}{(\mathrm{x_1 - x_2}) \cdot (\mathrm{x_1 - x_3})} + \frac{\mathrm{y_2 \cdot x_1 \cdot x_3}}{(\mathrm{x_2 - x_1}) \cdot (\mathrm{x_2 - x_3})} + \frac{\mathrm{y_3 \cdot x_1 \cdot x_2}}{(\mathrm{x_3 - x_1}) \cdot (\mathrm{x_3 - x_2})} \quad 50$$

The Lagrange interpolation of $C'(x)$ is showed in Figure 34C. As it is a second-order polynomial, it was easy to determine the position of its maximum, and therefore the abscissa of the inflection point:

$$\mathrm{c} = -\frac{\mathrm{a_1}}{2 \cdot \mathrm{a_2}} \quad 51$$

From the value of c, we calculated a' by a simple linear interpolation of the sigmoid around the inflection point:

$$\mathrm{a'} = \mathrm{y'_1} \cdot \frac{(\mathrm{c - x'_2})}{(\mathrm{x'_1 - x'_2})} + \mathrm{y'2} \cdot \frac{(\mathrm{c - x'_1})}{(\mathrm{x'_2 - x'_1})} \quad 52$$

where $(x'_1; y'_1)$ and $(x'_2; y'_2)$ are the coordinates of the two points of the sigmoid surrounding the inflection point.

5.2. Calculation of parameters b and d

In addition to giving the point of inflection, Lagrange polynomial also calculates the slope at the point of inflection. This slope s is simply the value of $L(x)$ in c.

$$\mathrm{s} = \mathrm{a_2 \cdot c^2 + a_1 \cdot c + a_0} \quad 53$$

s was also equal to the derivative of $C(x)$ for $x = c$. Based on equation 46, the equation became:

$$s = \frac{d' - a}{2} \cdot b$$

Therefore, b was calculated from s only if d' was estimated beforehand. However, d' corresponded to the value of $C(x)$ as x approached infinity. We assumed that the value $C(W)$

was a good approximation provided that the sigmoid had reached its asymptote for x = W, which was not sure (Figure 34A).

Consequently, the search for the values of b and d' could only be done using a very simple iterative method. In the first iteration, we fixed $d' = C(W)$ and $b = \frac{2 \cdot s}{d' - a}$ and we calculated the squared error ε_0 between the sigmoid and its model. Then, we increased d' by Δd and recalculated the squared error ε_1. We did the same until $\varepsilon_{k+1} > \varepsilon_k$ over two successive iterations to obtain the best approximation of d' (up to Δd) and b.

This method, although trivial, is effective for two reasons: i) the initial choice of b and d' provided clues about the direction in which d' should evolve to try to reduce the error, and ii) the reference data were not noisy. The results were perhaps less precise than those possibly obtained with more sophisticated optimization methods (gradient descent for example), but the main advantage of this method was that it could be easily implemented in Verilog- AT.

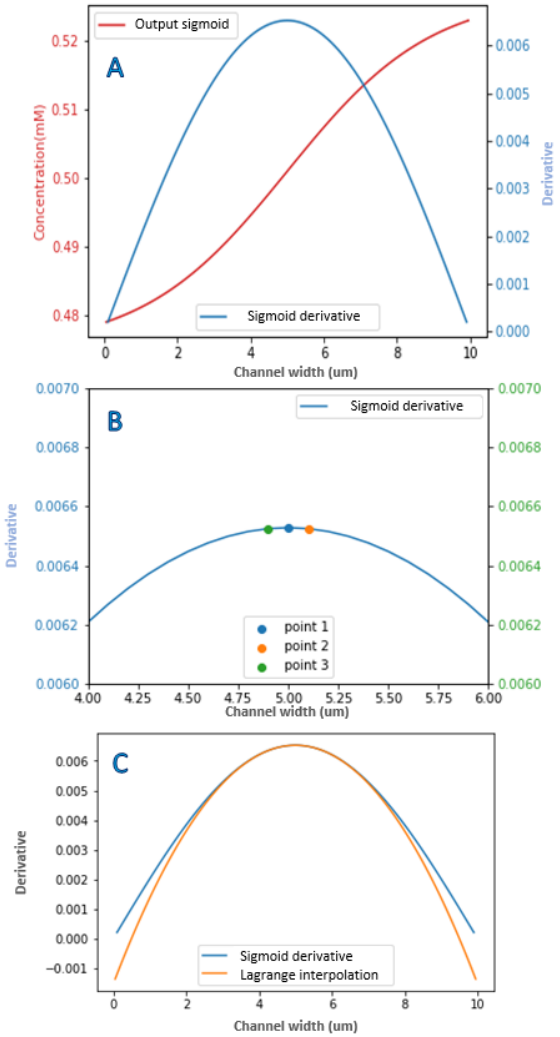


Figure 34. (A) Sigmoid curve following a Fourier series simulation and representation of its derivative. (B) Zooming on the top of the derivative curve, showing the three points used to carry out Lagrange interpolation. (C) Sigmoid derivative curve and superimposed curve of the second-order equation calculated from Lagrange interpolation.

6. Model breakdown

In this last section, we demonstrate that our model can serve as a building block to make up a model of a more complex microfluidic channel network. We wanted our compact model to be broken down for us to model successive mixers with the same compact model used repeatedly. For this, the inputs and outputs had to be compatible. Regarding the hydraulic parameters (pressure and flow rate), this compatibility was directly ensured by the fact that we used quantities bound by Kirchhoff's laws. Regarding the concentration profile, we had to produce a model corresponding to the one described in Figure 35.

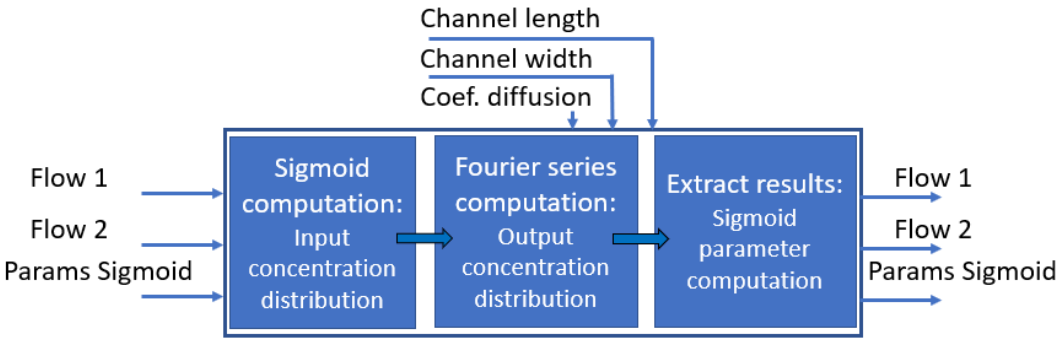


Figure 35. Representation of the compact model simulating a simple microfluidic mixer.

This module is made up of three successive blocks. The first block calculates the distribution of concentrations at the inlet of the channel. If the model corresponds to a section of the channel just after the junction, the sigmoid parameters are not yet known and must be calculated from the concentrations and flow rates according to Equation 45. If the channel section does not contain a junction, the sigmoid parameters correspond to those at the outlet of the previous channel section and are used by the model to calculate the input profile. The second block calculates the Fourier coefficients from the concentration profile according to Equation 42. Finally, the third block uses the procedure for extracting the sigmoid parameters described in section 4. These parameters are reintroduced at the input of the next section.

To validate the model setup, we compared the simulation results of a mixer consisting of a junction and a single 500-µm section with those of a mixer consisting of a junction and two successive 250-µm sections. Fig. 36A & B shows the raw results of the 500-µm simulation and the result of the fusion of the successive 250-µm simulations; the color bar represents the concentration in the channel. Figure 36C shows a 2D graph of the absolute difference between the two simulations; the colored bar represents the absolute error between Figure 36A and Figure 36B. The error was <1%, so we considered it negligible.

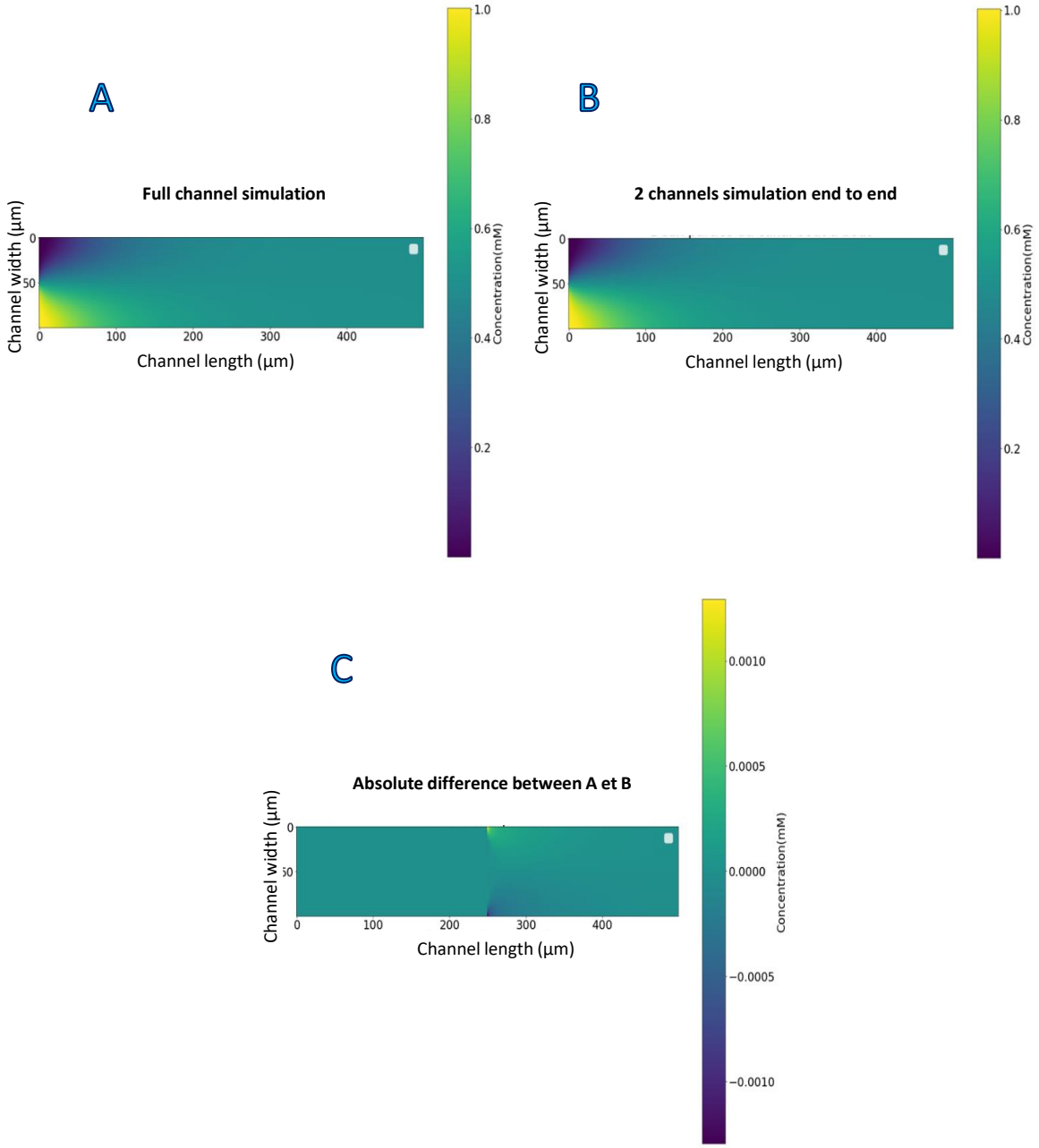


Figure 36. For simulation conditions such as Q_1 = Q_2 = 0.5n L/s, (A) Result of the simulation of a 500-µm microfluidic channel. (B) Result of the fusion of successive simulations of a 250-nm channel flowing into a second 250-nm channel. (C) Difference between (A) and (B).

7. Implementation of the compact model in the global simulation

Having a compact model directly translatable into SPICE language greatly helped us for general implementation.

Using the concept of a basic lab-on-a-chip developed in the previous chapter, we directly replaced our simple mixer by our compact model in a SPICE environment. The input data of the compact model remained the same (Figure 37), with:

- the hydrodynamic values of the two associated channels
- biological values such as concentrations

- biological parameters.

These values provided the sigmoid parameters, as seen above. Then, at the output of the compact model, we collected:

- hydrodynamic values
- the sigmoid parameters of each of the species, reflecting the distribution of the concentrations across the width of the channel.

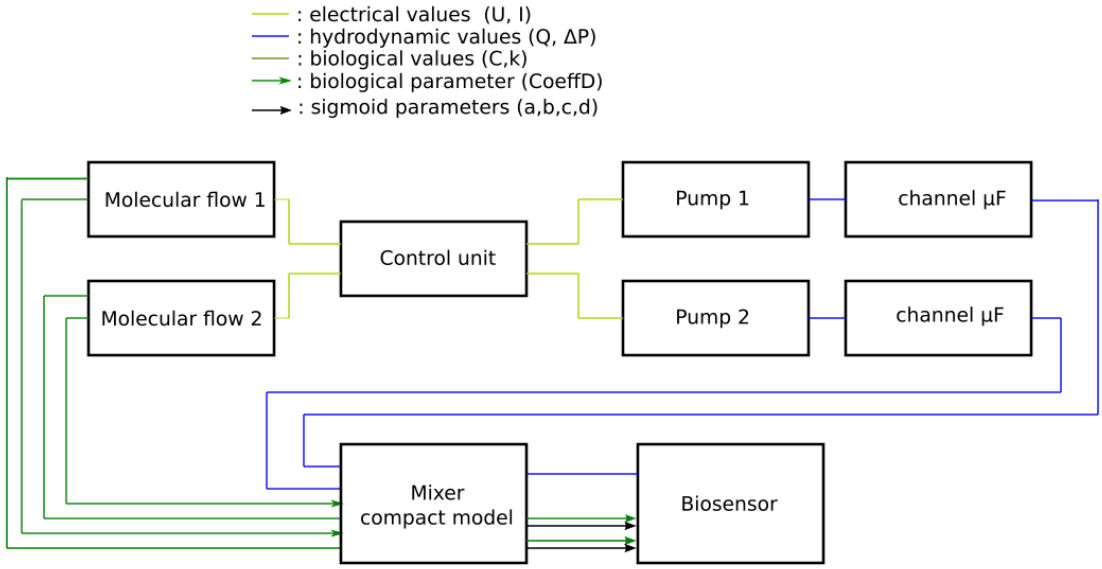


Figure 37. Schematic representation of the simulation of a lab-on-a-chip taking the compact mixer model into account.

8. Conclusions

In this chapter, we initially showed that the modeling of a microfluidic mixer by electrical analogy required improvement. In some cases, modeling the concentration gradient within a channel is essential for modeling labs on chips.

Then, we explained the phenomena associated with passive mixing in a microfluidic mixer. Using the associated equations, we developed a compact model by solving this equation using Fourier series decomposition. Then, we developed a compact model of a microfluidic mixer with convincing results compared with a more complex finite element simulator.

However, improvements still need to be made to obtain a complete compact model describing a microfluidic mixer because part of the sigmoid parameters were calculated with adjustment parameters obtained from the COMSOL simulation. These adjustment parameters are linked to the geometry of the junction between the two afferent channels. A relation could be found so as to do without ancillary simulation tools.

We failed to effectively describe the reaction within the mixer. Direct development *via* a Fourier series seemed particularly difficult.

To overcome this hurdle, we developed a 2D simulation tool using the finite differences compatible with a SPICE environment. This new tool enabled us to simulate the movements

of concentrations more precisely. It is compatible with BB-SPICE, so that it can be used to model biological reactions.

Chapter 4

2D finite difference simulator for low-level-abstraction simulation

In this chapter we develop an important part of our complex modeling tool – low-abstraction modeling – because we needed to model certain parts of the labs on chips more precisely. The challenge was to do low-abstraction modeling while remaining in a microelectronics simulation environment compatible with the tools developed in the previous chapters.

First, we are going to present the tool we used as a working base, *i.e*., a tool for modeling finite difference scattering in a SPICE environment. This tool was developed within our team, and we validated its usefulness for specific labs on chips.

In a second step, we are going to develop the improvements we brought to the tool in order to add advection – the impact of fluid movements on local concentrations.

1. Introduction

As seen above, we needed to have access to several levels of abstraction to create our simulator. First, we looked at the behavior of microfluidic systems as a whole, assuming that the concentrations were homogeneous in the channels. Then, in Chapter 3, we considered the case of a non-ideal mixer in which concentration gradients could exist. We wrote a behavioral model of this type of device based on the advection-diffusion equation. In the present chapter, we are going to consider the case of microfluidic devices in which the concentration is inhomogeneous and must be calculated at the lowest level of abstraction, *i.e.*, by directly solving the advection-diffusion equation described in the previous chapter.

The sample application used throughout this chapter is a microfluidic cavity. Many labs on chips use cavities where the analytes pass to be detected and analyzed using biosensors. To best simulate this, the concentration of the molecule of interest has to be determined precisely at the level of the biosensor, even if it is not necessarily homogeneous in the cavity. To do this, we developed a simulator capable of solving the advection-diffusion equation in a SPICE environment to make it compatible with other previously established models.

The development of this simulator was based on an existing tool developed within our research team that simulates the diffusion of molecules for applications in synthetic biology. In the present chapter, we are first going to demonstrate how this tool can be adapted for lab-on-a-chip simulation, and then how it can be enriched to take advection phenomena into account when the sample containing the analyte is carried along in a flux.

2. Presentation of the existing tool

The tool on which we based our work was developed as part of the thesis of Elise Rosati. Its objective was to simulate the behavior of biological systems in which the concentration of molecules was not uniform across space (*e.g.,* diffusion of molecules secreted by a cell, inter-cellular communication, etc.). The tool was published in 2018 [76]. Its operation is outlined below. A more detailed description and application examples can be found in Elise Rosati's thesis manuscript [64].

2.1. General presentation

The simulator was made up of several interconnected tools (Figure 38). The user first specified the input parameters of the system, namely the values and position of the molecular flows, the physicochemical characteristics of the studied species, and the geometry of the simulation space (the volume of interest). These parameters were used by a mesh generator coded in C ++ that discretized the volume of interest in parallelepipeds. The user chose the size of the elementary mesh but could also specify areas in which a finer mesh was necessary. The mesh generator provided two CSV files; one contained the list of network nodes with their coordinates, the other the list of meshes with the numbers of the nodes of each of the corners of each mesh.

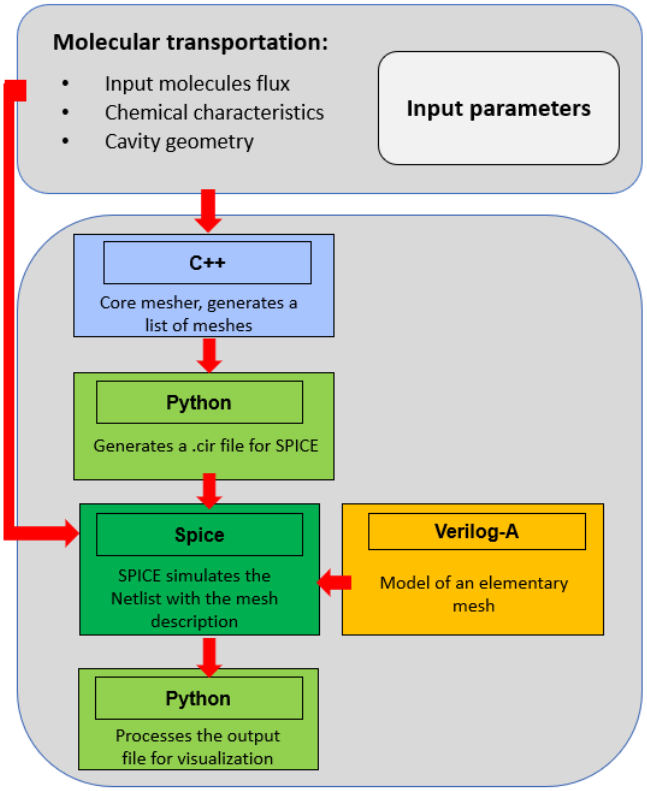


Figure 38. Schematic representation of the data transfer for the simulation of diffusion.

A Python script was used to read the CSV files and generate a SPICE netlist made up of a series of instances of an elementary mesh model written in Verilog-A or SPICE. More details on this elementary mesh model are given in section 2.b. The netlist was supplemented by molecule flows, models of chemical reactions likely to take place in the area of interest, other instances of components to simulate boundary conditions, and simulation guidelines.

The simulation was carried out by a SPICE simulator. Two simulators were validated: a free open-source one (NgSpice [77]), and an industrial one (Specter [78]). A new Python script was used to collect and process the simulation results.

2.2. Operation of the mesh generator

The mesh generator written in C ++ took a file describing the space to be discretized (size, minimum number of divisions for each direction) and a file of the areas where the mesh had to be refined and the degree of refinement associated with each area as input data. If a mesh overlapped a zone of refinement, it was divided into 8 sub-meshes, then each sub-mesh was again divided into 8 sub-sub-meshes, etc. This process was repeated as many times as the degree of refinement of the area required. Then, the algorithm checked that there was one degree of refinement difference at most between a given mesh and its direct neighbors. If not, the neighbors also underwent a refinement step to follow the rule of the degree of refinement difference.

Once the process was completed, the mesh generator generated a list of meshes and their associated nodes, and a list of all the coordinates of the nodes. The mesh was *a priori* in 3D, but could easily be reduced to a 2D case by considering only the mesh layer for $z = 0$.

The mesh generator provided two output files. The netlist was a CSV file made up of one row *per* mesh within the mesh and 27 columns *per* row. The first column was the mesh number. Columns 2-9 corresponded to the node number of each corner, columns 10 to 15 to the node number of the center of each face and columns 16 to 27 to the number of the middle of each edge. The value -1 was attributed if a node did not exist, *e.g.*, when two adjacent meshes had the same degree of refinement. The nodelist file was a CSV file made up of one line *per* node within the mesh and 4 columns. The first column was the node number; the next three were the coordinates of the nodes in x, y and z.

2.3. Model of an elementary mesh

The unit cell described the contribution of a cell to the flow of molecules arriving at each of its nodes. For the sake of simplicity, we described the model of the elementary mesh in 2D, but this model was also established in 3D [79]. The concentration $C(x, y, t)$ of a given molecule in a closed space was described by the 2D diffusion equation:

$$\frac{\partial \mathrm{C}}{\partial \mathrm{t}} = \mathrm{D} \cdot \left(\frac{\partial^2 \mathrm{C}}{\partial \mathrm{x}^2} + \frac{\partial^2 \mathrm{C}}{\partial \mathrm{y}^2} \right) - \mathrm{d_X} \cdot \mathrm{C} \qquad 54$$

where D is the diffusion coefficient (m²/s) and d_X the degradation constant. We considered the discretization of space in Figure 39.

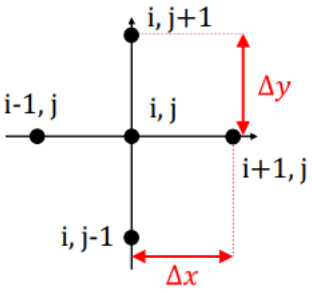


Figure 39. Notation of the discretized space.

Let us now consider a discretized space (Figure 39) and the concentration $C_{i,j}$ at a point of coordinates (i, j). Let Δx and Δy be the steps between the nodes in the two directions. If we apply a centered Euler discretization scheme, the second derivative of equation 54 can be expressed as a function of the concentrations of the nearby nodes:

$$\frac{\partial^2 \mathrm{C_{i,j}}}{\partial \mathrm{x}^2} = \frac{\frac{\partial \mathrm{C}}{\partial \mathrm{x}}(\mathrm{x} + \Delta \mathrm{x}) - \frac{\partial \mathrm{C}}{\partial \mathrm{x}}(\mathrm{x} - \Delta \mathrm{x})}{\Delta \mathrm{x}} = \frac{\frac{\mathrm{C}(\mathrm{x} + \Delta \mathrm{x}) - \mathrm{C}(\mathrm{x})}{\Delta \mathrm{x}} - \frac{\mathrm{C}(\mathrm{x}) - \mathrm{C}(\mathrm{x} - \Delta \mathrm{x})}{\Delta \mathrm{x}}}{\Delta \mathrm{x}} \qquad 55$$

Or also, using discretized space notation, like:

$$\frac{\partial^2 \mathrm{C_{i,j}}}{\partial \mathrm{x}^2} = \frac{\mathrm{C_{i+1,j}} - \mathrm{C_{i,j}} - (\mathrm{C_{i,j}} - \mathrm{C_{i-1,j}})}{\Delta \mathrm{x}^2} \qquad 56$$

Equation 56 can also be applied to the y coordinate. If we consider a square mesh where $\Delta x = \Delta y = \Delta l$, equation 54 becomes:

$$\frac{\partial \mathrm{C_{i,j}}}{\partial \mathrm{t}} = \frac{\mathrm{D}}{\Delta \mathrm{l}^2}\left(\mathrm{C_{i+1,j}} + \mathrm{C_{i-1,j}} + \mathrm{C_{i,j+1}} + \mathrm{C_{i,j-1}} - 4 \cdot \mathrm{C_{i,j}}\right) - \mathrm{d_X} \cdot \mathrm{C_{i,j}} \qquad 57$$

This differential equation can be found by considering an electrical node whose voltage would be $C_{i,j}$, connected to the four neighboring nodes by resistors of value $\frac{\Delta l^2}{D}$ and to ground by a capacitor of value 1 and a resistor of value $\frac{1}{d_X}$. Based on this analogy, an equivalent electrical model of an elementary cell was built by distributing these 8 components (Figure 40).

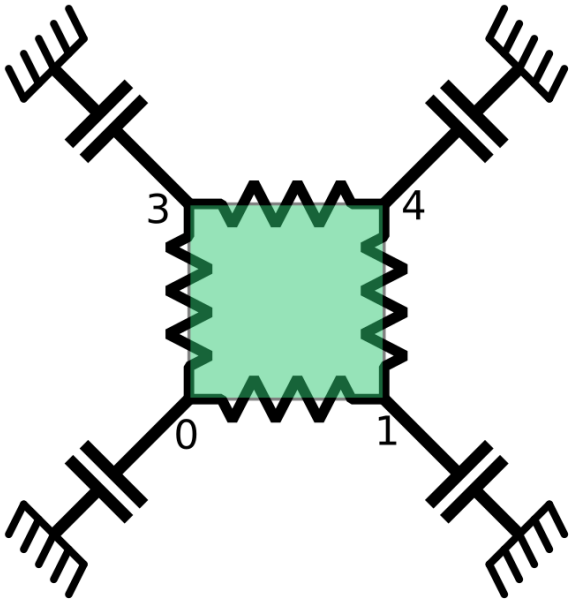


Figure 40. Diagram of an elementary mesh.

In the 4 neighboring meshes, $C_{i,j}$ is the common node (Figure 41). When the neighboring meshes are of different sizes, the number of nodes of the elementary mesh changes and the distribution of the electrical components between each of the nodes also changes. The calculations are detailed in [64].

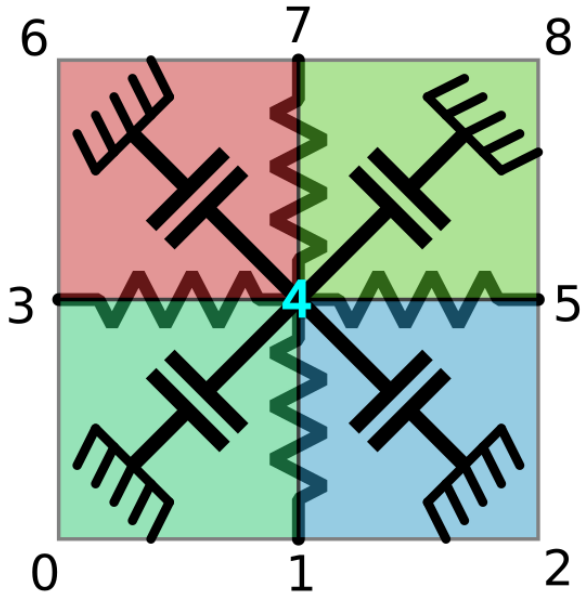


Figure 41. Diagram of a node in a simple square mesh.

The elementary mesh model was implemented in Verilog-A and in SPICE. It is composed of 9 terminals representing the molecular concentrations at each of the corners of the mesh (C0, C1, C2 and C3) and in the middle of each side (C01, C12, C23 and C30) as well as a reference node. It takes 10 parameters described in Table 3 as input data. The corner nodes are always connected, while the nodes in the middle of each side are only connected in case of refinement.

Table 3. Parameters of the mesh mode.

Parameters	Definition
ID	Mesh number
N	Refinement degree
X01, X12, X23 and X30	Boolean that indicates if a neighboring smaller mesh is connected in the middle of the corresponding edge
R_0	Local degradation constant
K_0	Scaling parameter (default: 1)
D_0	Local diffusion constant
Mesh size (Ms)	Maximum size of the edges of a mesh

2.4. Organization of the Netlist for SPICE

The SPICE simulator required a description of the system in the form of a netlist. This file contained the following data:

1) the definition of the global parameters of the model

2) the instantiations of the elementary meshes according to the list of meshes provided by the mesh generator

3) a map of the initial concentrations (optional)

4) instantiation of specific components to reproduce boundary conditions

5) instantiations of all the biological models of local mechanisms consuming or producing molecules

6) simulation guidelines.

Different types of boundary conditions can be described. By default, the space represented by the mesh is considered closed. From a modeling point of view, this corresponds to open circuits after each edge node. The space can also be thought of as large enough to have a fixed (often zero) concentration at its edges. In this case, the edge nodes are connected to fixed voltage sources. Finally, additional scattering beyond the mesh boundaries can also be modeled by adding a grounded resistor to all edge nodes. Obviously enough, the boundary conditions can be differentiated within a same simulation.

Biological models can also be represented by equivalent electrical networks [57].

3. Use of the existing tool for simulating a biosensor

3.1. Presentation of the case study

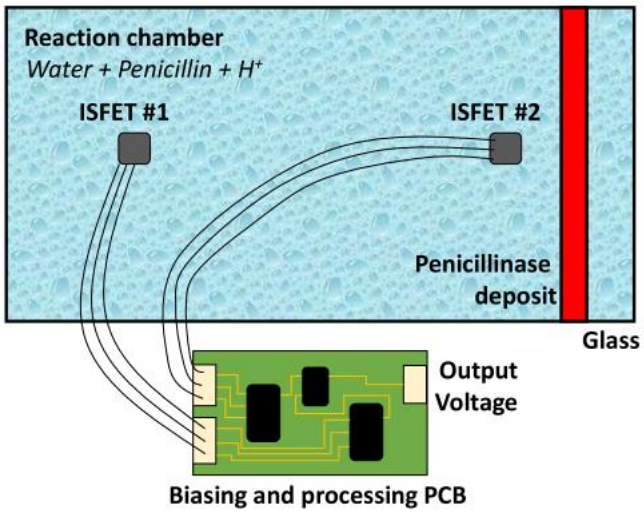


Figure 42. Diagram of a lab-on-a-chip for detecting penicillin.

To illustrate the possible simulation of a lab-on-a-chip with the tool described above, we chose the simple case study of a lab-on-a-chip for detecting penicillin (Figure 42). It was composed of a 40 x 20 x 5 mm^3 reaction chamber containing the sample to be analyzed. The dimensions were not microfluidic, but we were only studying diffusion, so there was no movement of fluids. Thus, the study was transcribed regardless of the dimensions. The rear face of the reaction chamber was a glass plate functionalized by local deposition of penicillinase. The solution inside the chamber was in contact with two ISFETs *via* holes drilled in the glass plate. The penicillin contained in the sample was degraded by the penicillinase on the glass. This

reaction produced H^+ ions, leading to a local pH decrease measured by a pair of ISFETs: one was close to the functionalized zone (ISFET # 2: measurement ISFET), and the other was distant (ISFET # 1: reference ISFET). The quantity of H^+ ions released and the reaction speed depended on the concentration of penicillin in the sample (the more penicillin in the solution, the faster the pH decrease).

3.2. Biosensor model

3.2.1. Model of the diffusion of molecules in the cavity

The tool described in section 2 was used to model the diffusion of molecules in the cavity. We first considered the problem as a 2D problem. The volume of the cavity was discretized in a surface of homogeneous 2x2-mm^2 meshes. In practice, two chemical species diffused into the cavity – H^+ ions and penicillin –, so that we had to implement two diffusion models (one for each molecule).

3.2.2. Biological model

Our lab-on-a-chip bio-receptor was an enzymatic reaction that transformed a substrate (S; β-lactam antibiotic or penicillin) into a product (P). The reaction was catalyzed by an enzyme (E; β-lactamase or penicillinase). B-lactamases acylate β-lactam and then use strategically positioned water molecules to hydrolyze the acylated substrate. In this way, β-lactamase is regenerated and can inactivate additional β-lactam molecules [80]. This reaction releases H^+ ions. The reaction can be represented by the following balance equation:

$$E + S \underset{k_{-1}}{\overset{k_1}{\rightleftarrows}} E{:}S \xrightarrow{k_2} E - S \xrightarrow{k_3, H_2O} E + P + H^+ \qquad 58$$

where *E:S* is the enzyme-substrate complex, *E-S* is the acyl-enzyme, k_1 and k_{-1} are the association and dissociation rate constants of the pre-acylation complex, respectively, k_2 is the constant of the acylation rate, and k_3 is the deacylation rate constant. Assuming that the first reaction was much slower than the other two, this mechanism was modeled by a simple Michaelis-Menten model giving the effective reaction rate as:

$$\mathrm{v}([\mathrm{E}], [\mathrm{S}]) = \mathrm{k_{cat}} \cdot [\mathrm{E}] \cdot \frac{[\mathrm{S}]}{\mathrm{K_m} + [\mathrm{S}]} \qquad 59$$

where

$$\mathrm{K_m} = \frac{\mathrm{k_3}}{\mathrm{k_2} + \mathrm{k_3}} \cdot \frac{\mathrm{k_{-1}} + \mathrm{k_2}}{\mathrm{k_1}}$$
$$\mathrm{k_{cat}} = \frac{\mathrm{k_2} \cdot \mathrm{k_3}}{\mathrm{k_2} + \mathrm{k_3}} \qquad 60$$

This reaction has been described in BB-SPICE [57] and was converted into an equivalent SPICE model.

3.2.3. ISFET model

An ISFET can be modeled basically as a MOSFET whose threshold voltage V_T is changed by a term that depends on the pH of the solution according to the following relation:

$$V_T = V_{T0} + S \cdot (pH - pH_0) \quad (61)$$

where V_{T0} is the threshold voltage of a standard MOSFET (fixed by geometry and technology), S is the sensitivity of the ISFET, which depends on the electrochemical properties of the electrodes and solutions according to Nernst's law, and pH_0 is a virtual pH value at which the threshold voltages of the ISFET and equivalent MOSFET are equal.

3.2.4. Complete model

The full model is depicted in Figure 43. It consists of two diffusion models, one for penicillin and the other for H^+ ions. The nodes, where the immobilized enzyme from the penicillin diffusion simulation and the H^+ ions are located, are linked to the biological reaction models (one *per* node). The ISFET models are connected to the H^+ ion diffusion model at the nodes corresponding to ISFET positionings.

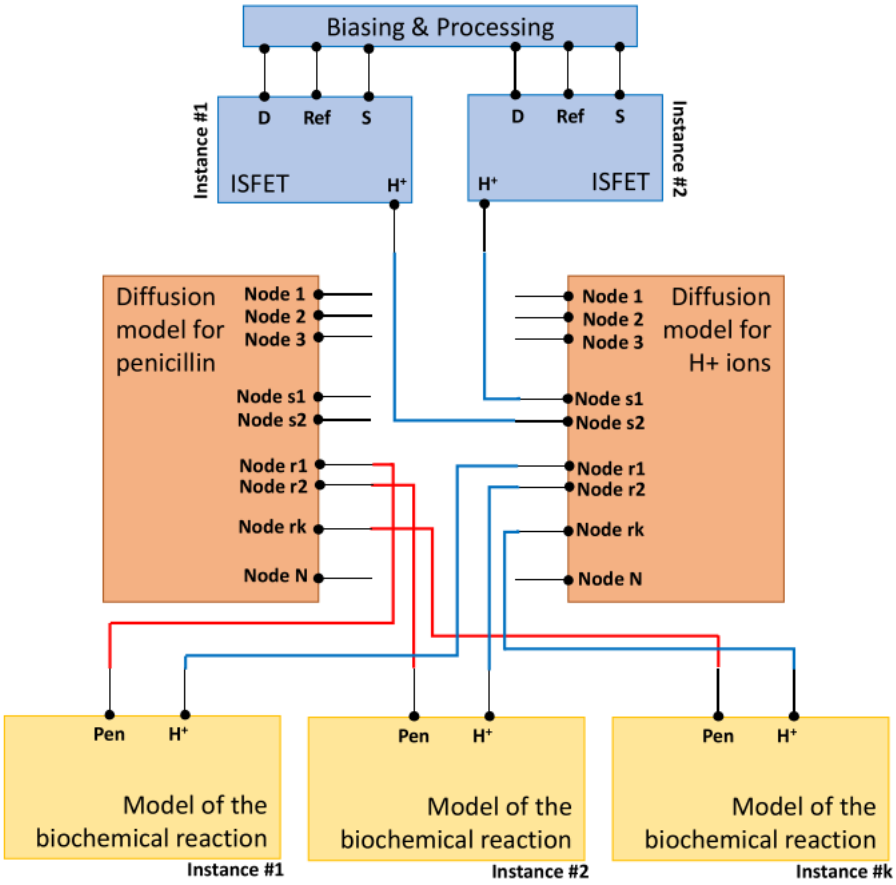


Figure 43. Overall diagram of the interactions between the different models.

3.3. Results

3.3.1. 2D Model

First, each part of the sensor was simulated individually. For the ISFET, the static characteristics $I_D - V_{GS}$ were shifted as a function of the pH (Figure 44). The shift was about 350 mV between the curve at pH 4 and the curve at pH 10, in accordance with the theory of Nernst's law with 60 mV / pH.

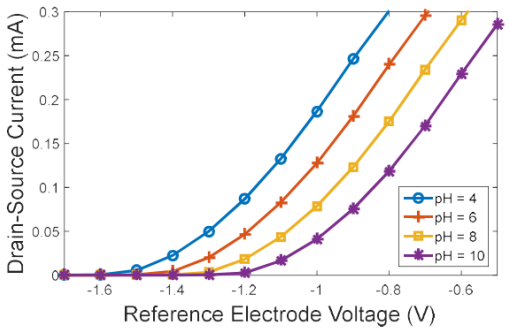


Figure 44. Results of the IFSET model simulation: drain-source current as a function of the reference voltage at different pH values.

Figure 45 shows the transient change in pH during the penicillin-penicillinase reaction for 2 different initial concentrations of penicillin and two different concentrations of penicillinase. Again, the results are consistent with the theory: the steady state (when all penicillin molecules are consumed) corresponds to a concentration of H^+ equal to the initial concentration of penicillin. In addition, the more penicillinase on the plate, the faster the reaction.

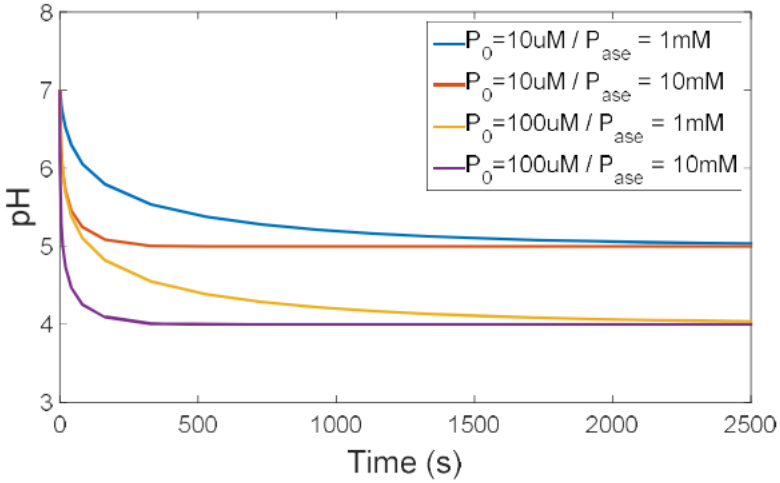


Figure 45. Transient simulation of the biochemical reaction: evolution of the pH of the solution at different initial penicillin concentrations and different penicillinase concentrations.

The diffusion model was simulated with constant H^+ sources at the nodes, corresponding to the functionalized zone (Figure 46). Six different configurations were tested. The diffusion constant of H^+ ions was fixed at 0.01 mm/s.

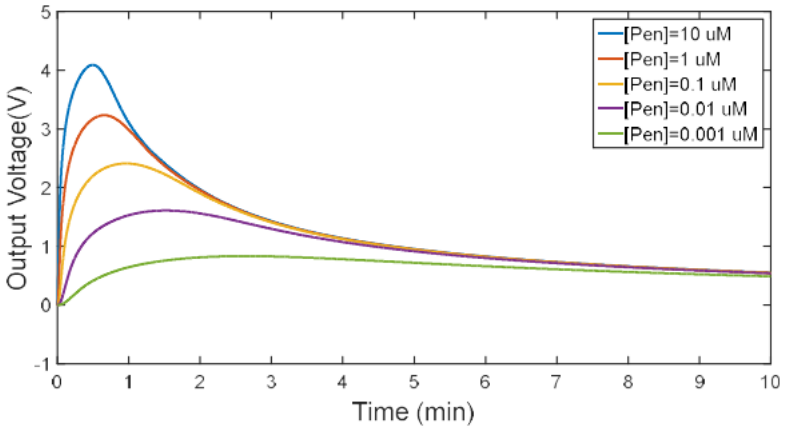


Figure 46. Simulated transient response of the biosensor as a function of the penicillin concentration.

Finally, the full model was simulated for different concentrations of penicillin in the sample (Figure 47). Bell-shaped responses were observed, composed of two phases. First, H^+ ions were produced and diffused to the measuring ISFET, so that the output voltage increased. Then, H^+ production decreased (penicillin was completely consumed), and more and more ions reached the reference ISFET, so that the output voltage decreased. In steady state (still not reached after 10 minutes), the output voltage tended towards 0.

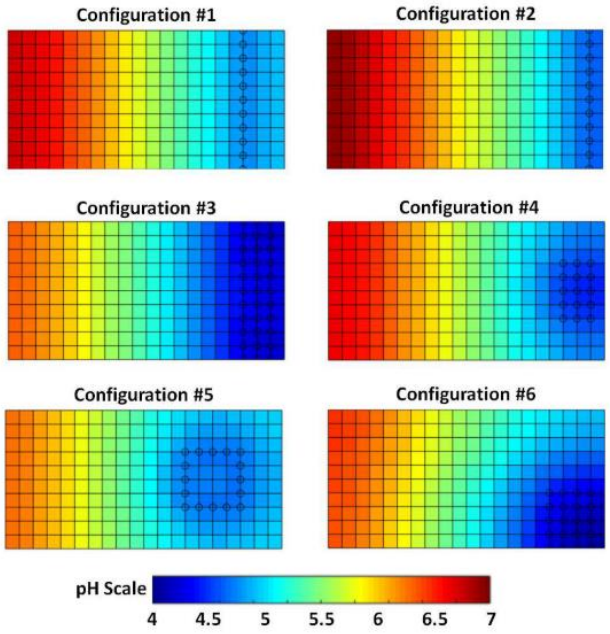


Figure 47. Simulation results of the diffusion model: pH maps simulated for six different configurations. For each simulation, H^+ production was modeled by local sources (dots), and a reaction consuming H^+ was added everywhere.

Figure 48 shows the different results obtained under the different configurations. This study was published [76].

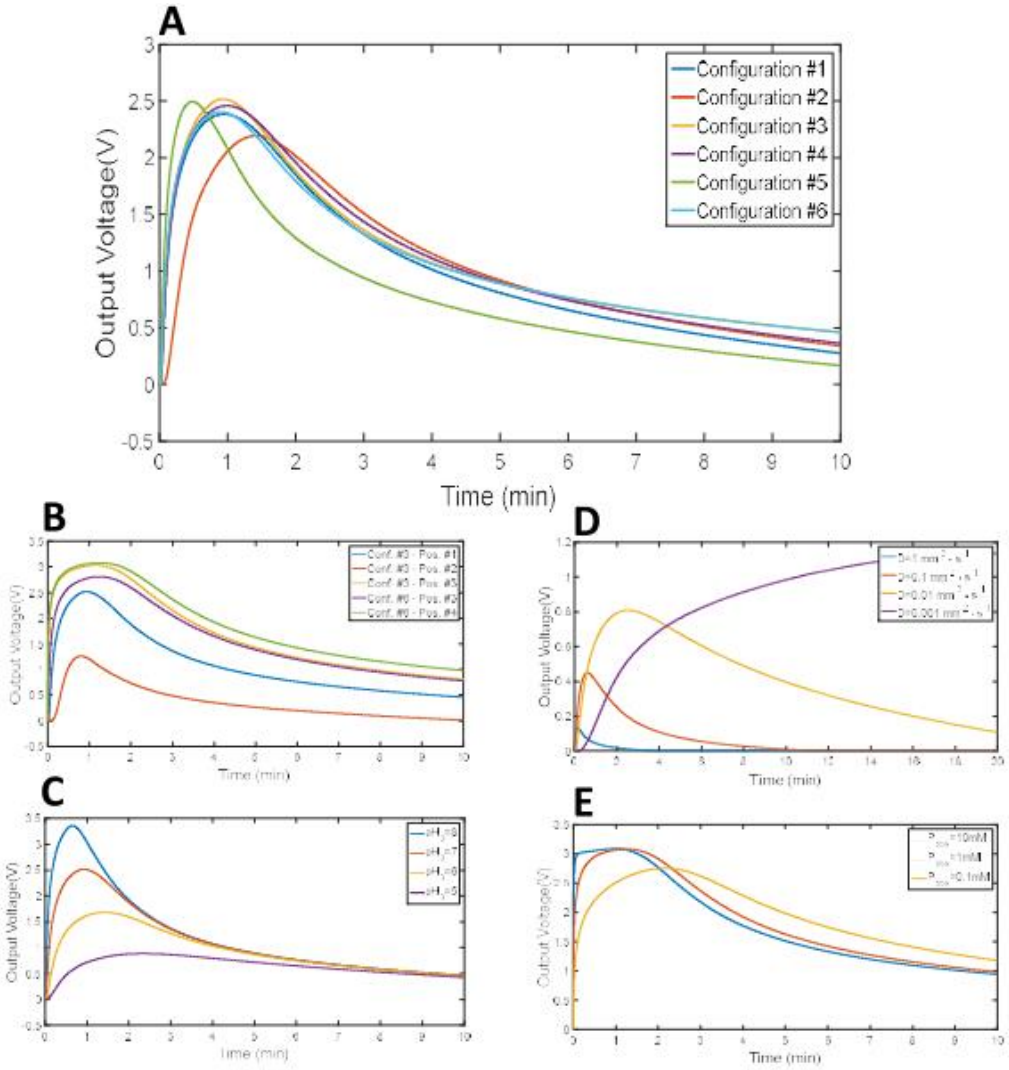


Figure 48. Results simulated with the virtual prototype of the sensor.

3.3.2. 3D Model

As described in section 2, the tool for simulating molecular diffusion also works in 3D. Therefore, we used it to test alternative configurations of electrode positioning and penicillin deposition while exploiting the 3 dimensions of the cavity.

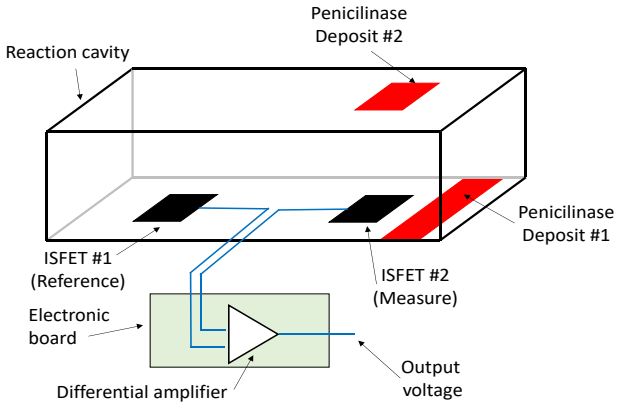


Figure 49. Diagram of a lab-on-a-chip for detecting penicillin in 3D.

The new version of the biosensor is described in Figure 49. It was composed of a 40 x 20 x 10 mm^3 cavity filled with the sample to be analyzed. At the bottom of the cavity, the solution was in contact with two ISFETs measuring local pH variation. The solution was also in contact with a strip of immobilized penicillinase that could be deposited on the bottom of the cavity, to the right of the second ISFET, or on the top of the cavity, facing the second ISFET.

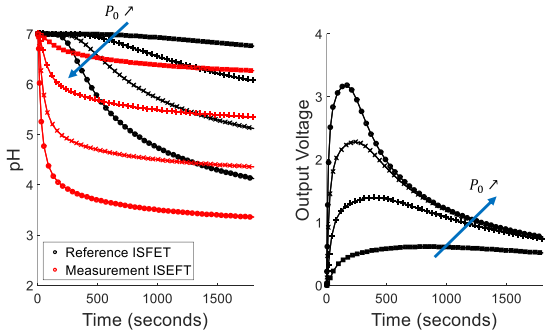


Figure 50. Simulation results of the full model of the sensor under standard configuration. Dots, initial penicillin concentration of 1 mM; crosses, 100 µM; + symbols, 10 µM; squares, 1 µM. Left, comparison of the pH values calculated at the nodes corresponding to the reference IFSET (black) and the measurement IFSET (red). Right, output voltages.

Transient simulations were performed with an initial pH of 7, the 1 mM penicillinase deposit in position #1, and four initial penicillin concentrations in the sample: 1 mM, 100 µM, 10 µM and 1 µM. The results are given in Figure 50. The left sub-graph shows the simulated pH values at each ISFET. Two different behaviors were observed. At the start of the measurements, the reference ISFET played its role and the pH measured at the output of the circuit corresponded to the pH at the measurement ISFET. After a few minutes, the pH of the reference ISFET also dropped due to the diffusion of H^+ ions into the cavity. At this point, the biological measurement was no longer relevant. This was confirmed by the bell-shaped output voltage in the graph on the right. For a concentration of 1 mM, the expected output was 4 V, but the voltage actually peaked at 3.2 V after 170 seconds before decreasing. Such results are very important for the designers when optimizing their sensors, as showed below.

The same simulations were performed under configuration #2 (Figure 51). The sensor was less sensitive under this configuration. Moreover, the problem of the pH drift at the reference ISFET was not solved.

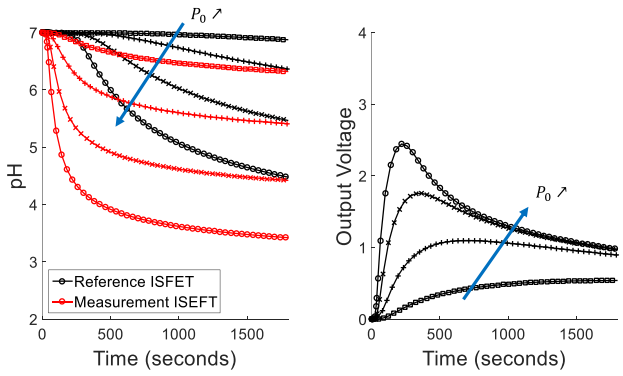


Figure 51. Simulation results obtained for the full model of the sensor under configuration #3. Dots, initial penicillin concentration of 1 mM; crosses, 100 µM; + symbols, 10 µM; squares, 1 µM. Left, comparison of the pH values calculated at the nodes corresponding to the reference IFSET (black) and the measurement IFSET (red). Right, output voltages.

The 3D reaction-diffusion model also allowed studying more complex geometries. For example, to overcome the drift problem, we could imagine creating a wall inside the cavity between the two ISFETs (Figure 52). Such a geometry can be easily modeled with our tool by imposing a zero diffusion coefficient at the level of the holes.

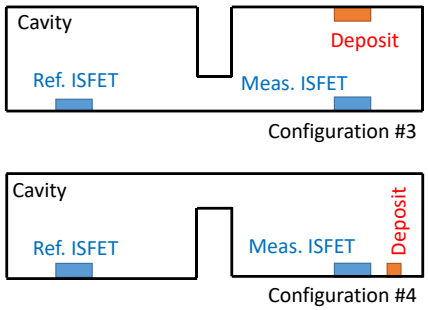


Figure 52. Description des deux nouvelles configurations comprenant une paroi à l'intérieur de la cavité pour gêner la diffusion de H + de l'ISFET de mesure à celui de référence. Description of two new configurations including a wall inside the cavity to slow down H^+ diffusion from the measurement IFSET to the reference IFSET.

Two new geometries described in Figure 52 were simulated under the same conditions as previously, depending on the location of the penicillinase deposit. In both cases, the wall was located on the deposit side (Figure 52) to hinder the diffusion of H^+ ions towards the reference ISFET. The results are given in Figure 51 for configuration #3 and in Figure 53 for configuration #4. The efficiency of the wall was demonstrated. It doubled the time during which measurements were relevant (the pH at the reference ISFET decreased by less than 1%). Thanks to this modified sensor geometry, the output of the measurement ISFET reached a higher value, *i.e.*, the sensitivity of the sensor was improved. The peak value of the sensor response increased by 10% under both configurations.

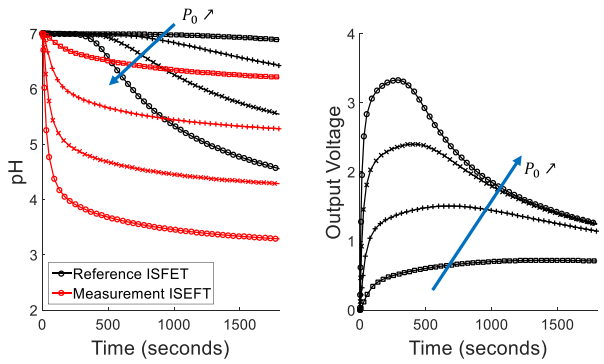


Figure 53. Simulation results obtained for the full model of the sensor under configuration #4. Dots, initial penicillin concentration of 1 mM; crosses, 100 µM; + symbols, 10 µM; squares, 1 µM. Left, comparison of the pH values calculated at the nodes corresponding to the reference IFSET (black) and the measurement IFSET (red). Right, output voltages.

4. Integration of advection

The initial version of the simulator perfectly simulated diffusion and could be used directly in our lab-on-a-chip modeling environment. However, many labs on chips take moving fluids into account [73] [74], so we upgraded the existing simulator to integrate advection phenomena.

4.1. General presentation

Figure 54 is a description of the data flows and working environments used to generate the model, using the structure of the existing tool (Figure 38) to which two new main modules were added.

The new equation to be solved to take advection and diffusion into account was:

$$\frac{\partial C}{\partial t} = D \cdot \Delta C - \nabla(\mathbf{v} \cdot C) + \sum_k r_k \qquad 62$$

where D (m² / s) is the diffusion coefficient of the molecule, $\mathbf{v}$ (m / s) the speed of the fluid, and r_k the speed of the k^{th} chemical reaction. These three quantities can depend on time and space. r_k also depends on the concentrations of the other molecules involved. On the other hand, the velocity $\mathbf{v}$ of the fluid is obtained by solving Stokes' equation (see Chapter 3).

The complete simulation of advection and diffusion phenomena in a microfluidic circuit involved the coupled resolution of two partial differential equations, which is possible with multiphysics simulators (*e.g.*, COMSOL), but ambitious using equivalent SPICE models. On the other hand, considering that the microfluidic system was in a steady state, $\mathbf{v}$ only depended on space and became a constant of the equation with respect to time. This function – or velocity map $\mathbf{v}(x, y, z)$ – was calculated by solving Stokes' equation statically and integrated at the level of the unit cell *via* constant parameters. This was the role of the two additional modules.

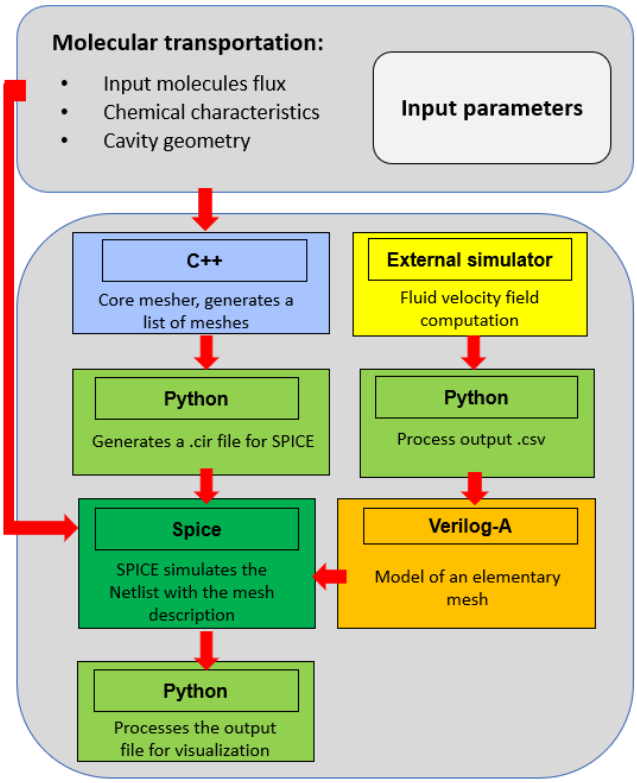


Figure 54. Basic modules of the tool. Labels, programming languages.

4.2. Velocity field simulation

The first step in the model generation process was to calculate the speed map. Stokes' equation, which describes the flow of a viscous, incompressible fluid at small Reynolds numbers [81], was used. It was written as follows:

$$\mu\Delta\mathbf{v} = \nabla\, \mathrm{p} - \rho \cdot \mathbf{f} \qquad 63$$

Where p is pressure (Pa), ρ is the density of the fluid (kg.m^{-3}), μ is dynamic viscosity (kg / m / s), $\boldsymbol{v}$ is the speed map (m / s) and $\boldsymbol{f}$ is a mass force (N / kg) exerted on the fluid (*e.g.*, gravity). This force is often negligible under the type of microfluidic conditions that we are studying.

The equation was solved by a partial differential equation solver. These solvers use a finite element approach to solve Stokes' equation by discretizing the space with a mesh specific to the tool. Two tools were tested: COMSOL Multiphysics®, and FeelPP [82]. FeelPP is an open-source finite-element differential equation solver developed by a research team of the Institute for Applied Mathematics Research (Pr. Prud'homme / IRMA) in Strasbourg.

The results presented below were obtained with COMSOL Multiphysics® for easier handling. Though commercial, COMSOL has a graphical interface that greatly facilitates testing and validation. Nevertheless, we checked that COMSOL and FeelPP provided equivalent speed maps and could be freely swapped for the different case studies presented here.

The simulator yielded the speed map as a text file (COMSOL) or a CSV file (FeelPP). These speeds were calculated at the nodes of the mesh used to solve Stokes' equation. The mesh did not necessarily correspond to the mesh generated by our own tool, so that an interpolation step was necessary; it was carried out by simple linear interpolation under Python.

4.3. New model of the unit cell

The new model of the unit cell was obtained by discretizing the advection-diffusion equation. The model was first written and validated in 2D. $C(x, y, t)$ was the distribution of concentrations across time and space. The reaction-advection-diffusion equation, which fixes the concentration profile of each molecule involved in the area of interest, was written as follows:

$$\frac{\partial \mathrm{C}}{\partial \mathrm{t}} = \mathrm{D} \cdot \Delta \mathrm{C} - \nabla(\boldsymbol{v} \cdot \mathrm{C}) + \sum_{\mathrm{k}} \mathrm{r_k} \qquad 64$$

Where D (m² / s) is the diffusion coefficient of the molecule, $\mathbf{v}$ (m / s) is the velocity field of the fluid calculated beforehand by a digital solver, r_k (mol / L / s) is the velocity of the k^{th} chemical reaction, which depends on the localization and the concentration of the different species involved in the reaction.

As before, the model of the unit cell was written to obtain, once assembled in a mesh, a set of ordinary differential equations corresponding to the discretization of Equation (64) by finite differences in the absence of the reaction term. For example, if we consider a node (i, j), its

associated concentration $C_{i,j}$, the local diffusion constant $D_{i,j}$ and the two local components of the speed vector $v_{x,(i,j)}$ and $v_{y,(i,j)}$, Equation (64) at this node is written:

$$\frac{\partial \mathrm{C_{i,j}}}{\partial \mathrm{t}} = \mathrm{D_{(i,j)}} \cdot \left(\frac{\partial^2 \mathrm{C_{i,j}}}{\partial \mathrm{x}^2} + \frac{\partial^2 \mathrm{C_{i,j}}}{\partial \mathrm{y}^2} \right) + \mathrm{v_{x,(i,j)}} \cdot \frac{\partial \mathrm{C_{i,j}}}{\partial \mathrm{x}} + \mathrm{v_{y,(i,j)}} \cdot \frac{\partial \mathrm{C_{i,j}}}{\partial \mathrm{y}} \quad (65)$$

Finite difference discretization of the second-order terms of Equation (65) with a central scheme read as follows:

$$\frac{\partial^2 \mathrm{C_{i,j}}}{\partial \mathrm{x}^2} = \frac{\mathrm{C_{i-1,j}} + \mathrm{C_{i+1,j}} - 2 \cdot \mathrm{C_{i,j}}}{\Delta \mathrm{x}^2} \quad (66)$$

where Δx^2 is the distance between two consecutive nodes in the direction of x. Similarly, discretization of the first-order derivative with a central scheme read:

$$\frac{\partial \mathrm{C_{i,j}}}{\partial \mathrm{x}} = \frac{\mathrm{C_{i+1,j}} - \mathrm{C_{i-1,j}}}{2 \cdot \Delta \mathrm{x}} \quad (67)$$

Discretization on the y axis was obtained in the same way. Finally, the discrete writing of equation (65) for a square mesh ($\Delta x = \Delta y = \Delta l$) was:

$$\frac{\partial \mathrm{C_{i,j}}}{\partial \mathrm{t}} = \frac{\mathrm{D_{i,j}}}{\Delta \mathrm{l}^2} (\mathrm{C_{i+1,j}} + \mathrm{C_{i-1,j}} + \mathrm{C_{i,j+1}} + \mathrm{C_{i,j-1}} - 4 \cdot \mathrm{C_{i,j}}) + \frac{\mathrm{v_{x,(i,j)}}}{2 \cdot \Delta \mathrm{l}} (\mathrm{C_{i+1,j}} - \mathrm{C_{i-1,j}}) + \frac{\mathrm{v_{y,(i,j)}}}{2 \cdot \Delta \mathrm{l}} (\mathrm{C_{i,j+1}} - \mathrm{C_{i,j-1}}) \quad (68)$$

The first term of this equation was modeled by the same electrical components as in the case of pure diffusion. The second term was implemented directly because $C_{i+1,j}$ and $C_{i-1,j}$ were not known in the same cell. As a result, this term was split into two: $(C_{i+1,j} - C_{i,j}) + (C_{i,j} - C_{i-1,j})$. Then, each of these two terms was modeled by current sources applied to the nodes depending on the concentration in these nodes. It is important to note that unlike the diffusion terms, these terms were oriented: the flow coming from the left tended to decrease with $C_{i,j}$, while the flow coming from the right tended to increase. This particular property of the advection terms led us to use controlled current sources rather than simple resistors, as in the case of diffusion. The model of the unit cell is given in Figure 55.

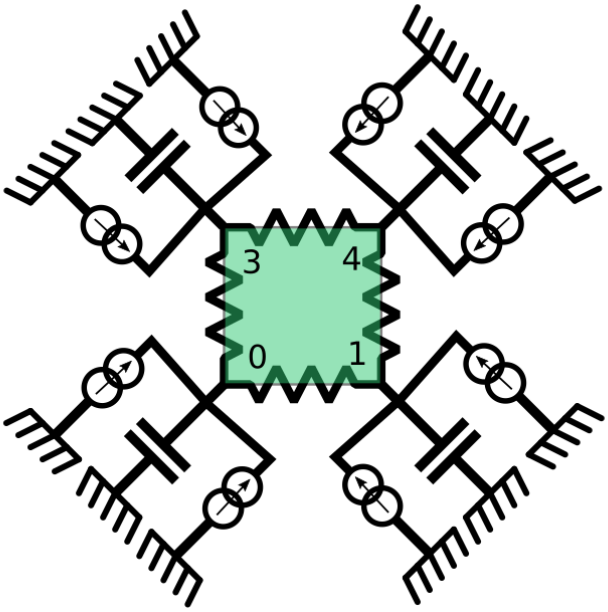


Figure 55. Verilog-A representation of a mesh. Resistances between nodes are proportional to the diffusion coefficient. The current sources are proportional to the velocity field.

All the electronic components that formed the mesh network were divided into 4 identical subunits and duplicated 4 times (one for each node) to obtain the model of the elementary mesh. Finally, the model had four biological terminals, one for each corner of the rectangle.

5. Validation of the advection-diffusion model

5.1. Validation of the unit cell model based on a simple case

The modeling of the diffusion terms was validated by [64], so we focused on the advection terms. First, we validated our model in particular cases in which the diffusion-advection equation had an analytical solution, *e.g.*, when the velocity field was written as follows:

$$\boldsymbol{v}(\mathrm{x},\mathrm{y}) = \begin{pmatrix} \mathrm{v}\cdot\dfrac{1}{1+\mathrm{x}} \\ 0 \end{pmatrix} \tag{69}$$

Then, the static equation (64) with a degradation term was written:

$$\mathrm{D}\cdot\left(\frac{\partial^2\mathrm{C}}{\partial\mathrm{x}^2}+\frac{\partial^2\mathrm{C}}{\partial\mathrm{y}^2}\right)+\mathrm{v}\cdot\frac{1}{1+\mathrm{x}}\cdot\frac{\partial\mathrm{C}}{\partial\mathrm{x}}-\mathrm{d}\cdot\mathrm{C}=0 \tag{70}$$

With initial conditions with homogeneous limits in y, we showed that C only depended on x. Equation (70) had an analytical solution that could be expressed with Bessel functions in the following form:

$$\mathrm{C}(\mathrm{x}) = (1+\mathrm{x})^{-\mathrm{n}}\cdot\left(\mathrm{A}\cdot\mathrm{J}_{\mathrm{n}}(\mathrm{i}\cdot\alpha\cdot\mathrm{x})+\mathrm{B}\cdot\mathrm{Y}_{\mathrm{n}}(\mathrm{i}\cdot\alpha\cdot\mathrm{x})\right) \tag{71}$$

where J_n and K_n are the Bessel functions of the first and second kind, and $\alpha = \sqrt{\frac{d}{D}}$ and $n = \frac{D^2-v}{2}$.

Using our environment and our elementary cell model, we calculated the concentration map in the case of a 100x100-µm square discretized into 1x1-µm squares with a constant concentration on the right edge of the 10-mM square, and zero concentration on the left edge of the square as a limiting condition. The diffusion coefficient D was fixed at 1 µm²/s, the speed v at 0.5 µm/s and the degradation constant d at 1.25 s^{-1}. The simulation results are presented in Figure 56-A1 for a negative velocity flow v oriented in the same direction as diffusion, and in Figure 56-B1 for a positive velocity v oriented in the opposite direction to diffusion.

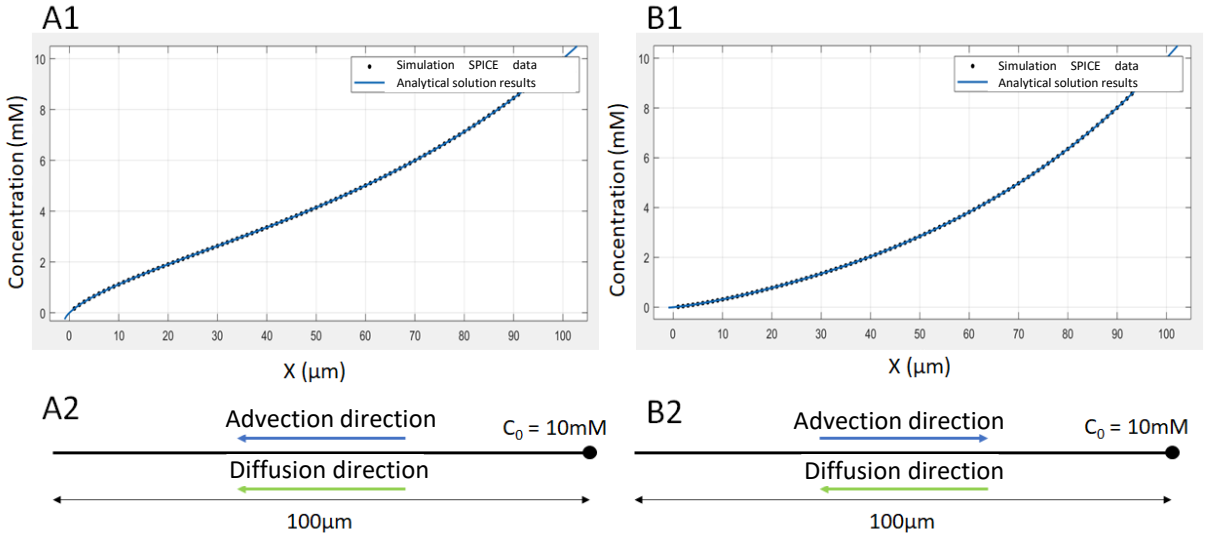


Figure 56. Validation of velocity as represented by a current source. A2 & B2, schematic representation of the 1D simulation using SPICE (the difference between the two models is the direction of velocity); A1 & B1, results obtained using SPICE and results obtained using the analytical tool.

The simulation results were imported into MATLAB to be compared with the analytical solution found earlier. We used the *curvefitting* function in MATLAB to check to what extent we could model our simulated result with the analytical function of Equation 64 by using functions of n and α for A and B respecting the boundary conditions, leaving parameters n and α free. The results of this adjustment can be found in Figure 56. The adjustment was obtained for $n = 0.75$ and $\alpha = 0.025$ (the expected values for parameters D, v and d).

Based on these results, we validated our new unit cell model for a simple case.

5.2. Validation by comparison with COMSOL

The second validation step consisted in comparing the results obtained under our simulation environment with those obtained under COMSOL in more complex cases. The simulated cavities were generally rectangular to have a mesh and a delimitation of conditions at the edges (wall, inlets, outlets, etc.) that were simple and directly compatible with the square meshes of our simulator.

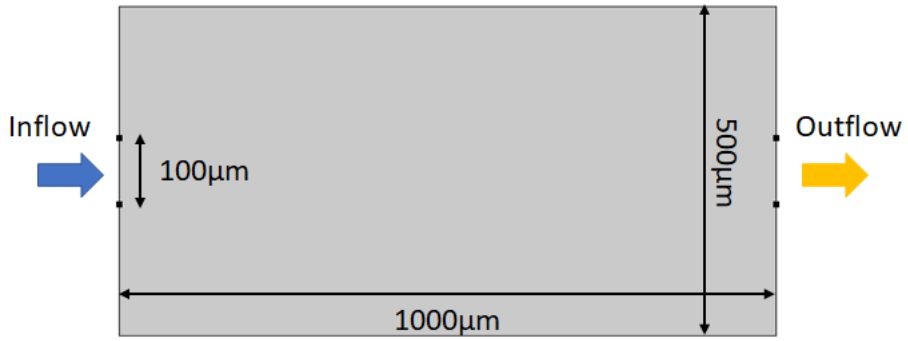


Figure 57. First example of a cavity geometry for simulating molecular transport.

The first simulation tests were carried out with a rectangular cavity measuring 500 x 1,000 µm. The fluid entered through a 100-µm opening centered to the left of the cavity and went out through an equivalent opening to the right of the cavity. We worked with a diffusion coefficient of $1e^{-9}$ m² / s, *i.e.*, a standard diffusion coefficient for molecules like small proteins [83]. The starting concentration at the inlet of the cavity was 1 mM. During the tests, the speed of entry of the fluid into the cavity was made to vary across three orders of magnitude, between 10^{-6} and 10^{-3} m / s.

We used two extensions of COMSOL Multiphysics to calculate the reference concentration maps. Using the first extension, we calculated the velocity field associated with fluid mechanics; using the second one, we calculated the concentration movements. For the sake of simplicity, we reused the velocity field maps calculated by COMSOL during the reference simulation to integrate them into our simulation environment.

Two types of conditions at the concentration boundaries were tested. The first condition was to impose fixed concentrations at the inlet and outlet. This made it easier to learn and compare with COMSOL but was not very realistic from a biological point of view. The second condition consisted in imposing a flow of molecules at the inlet and leaving a free flow at the outlet proportional to the concentration at the outlet.

The results of the SPICE simulation are given in Figure 58. The absolute difference between the COMSOL and SPICE simulations is given in Figure 59. The maximum absolute error between the COMSOL and SPICE simulations was 5.3%, while the average root mean square error (RMSE) for the entire cavity was less than 3%.

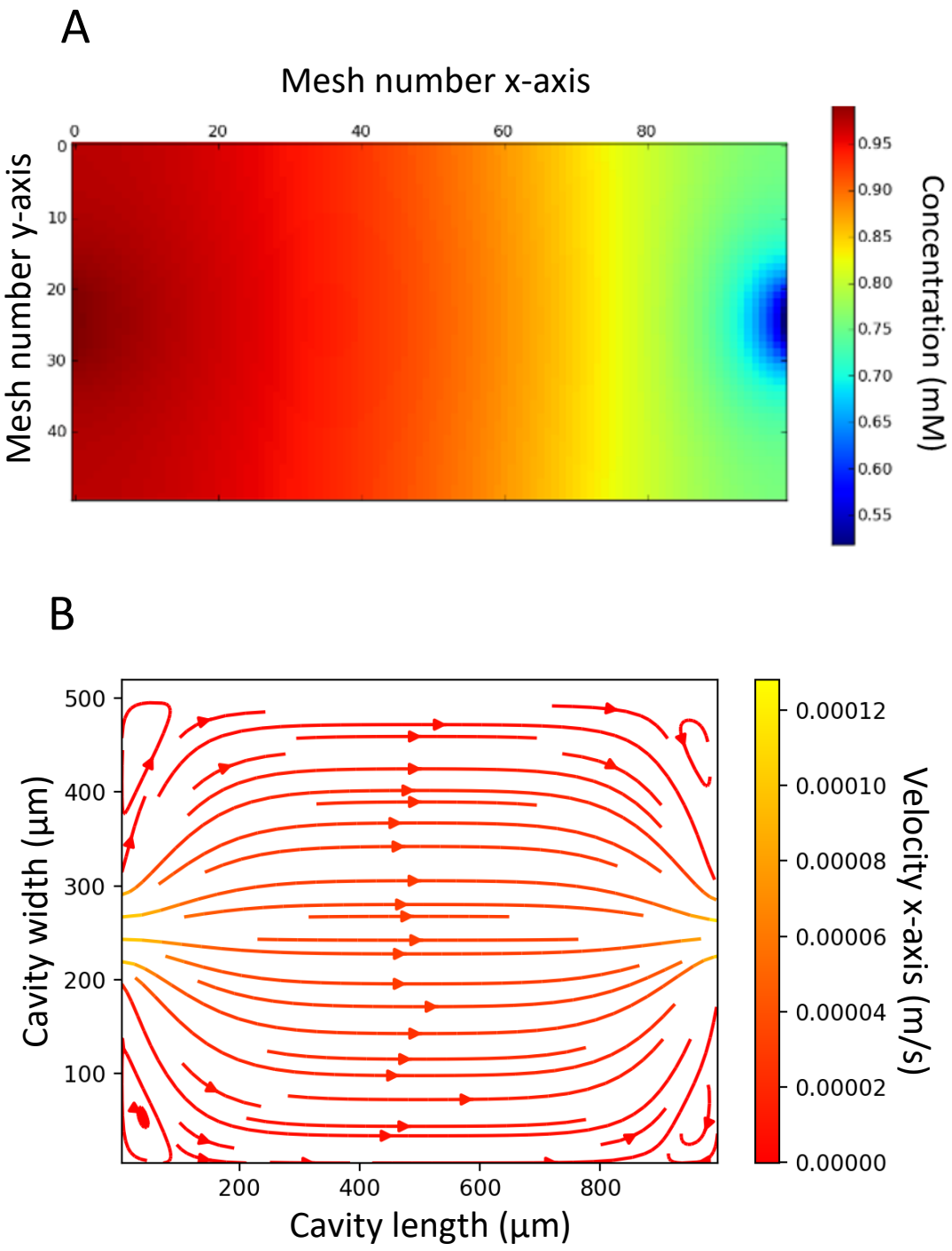


Figure 58. Results of the simulation of the first cavity geometry. (A) Results for the concentrations. (B) Associated velocity field.

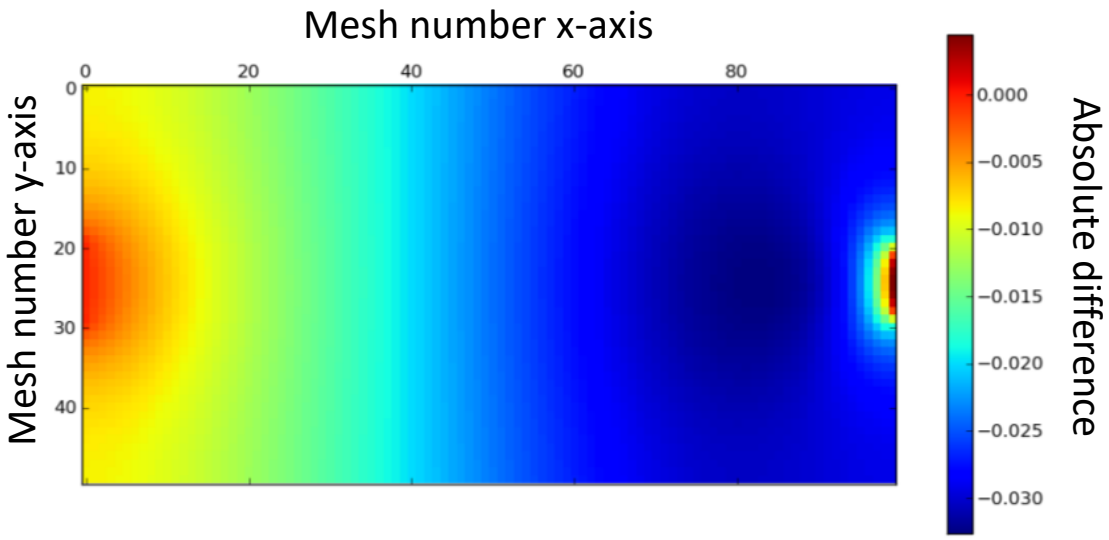


Figure 59. Schematic representation of the absolute simulation differences between our simulator and COMSOL.

The second geometry was a cavity similar to the previous one, but with a narrowing 50 µm long and 50 µm wide in its middle (Figure 60). To simulate the internal walls of the cavity and the space in which molecules could not diffuse, we set the diffusion coefficient of the meshes corresponding to these holes to 0.

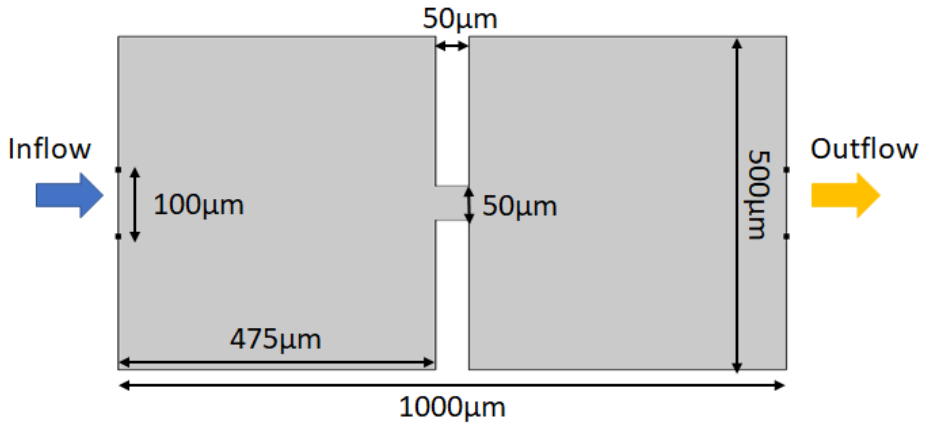


Figure 60. More atypical cavity geometries commonly found in labs on chips, with two successive cavities.

Similar errors were measured for different input streams and different boundary conditions. We also simulated the second geometry (Figure 60). Here again, the COMSOL and SPICE simulations were similar, and the RMSE was less than 5%. One of the simulation results of this geometry is showed in Figure 61.

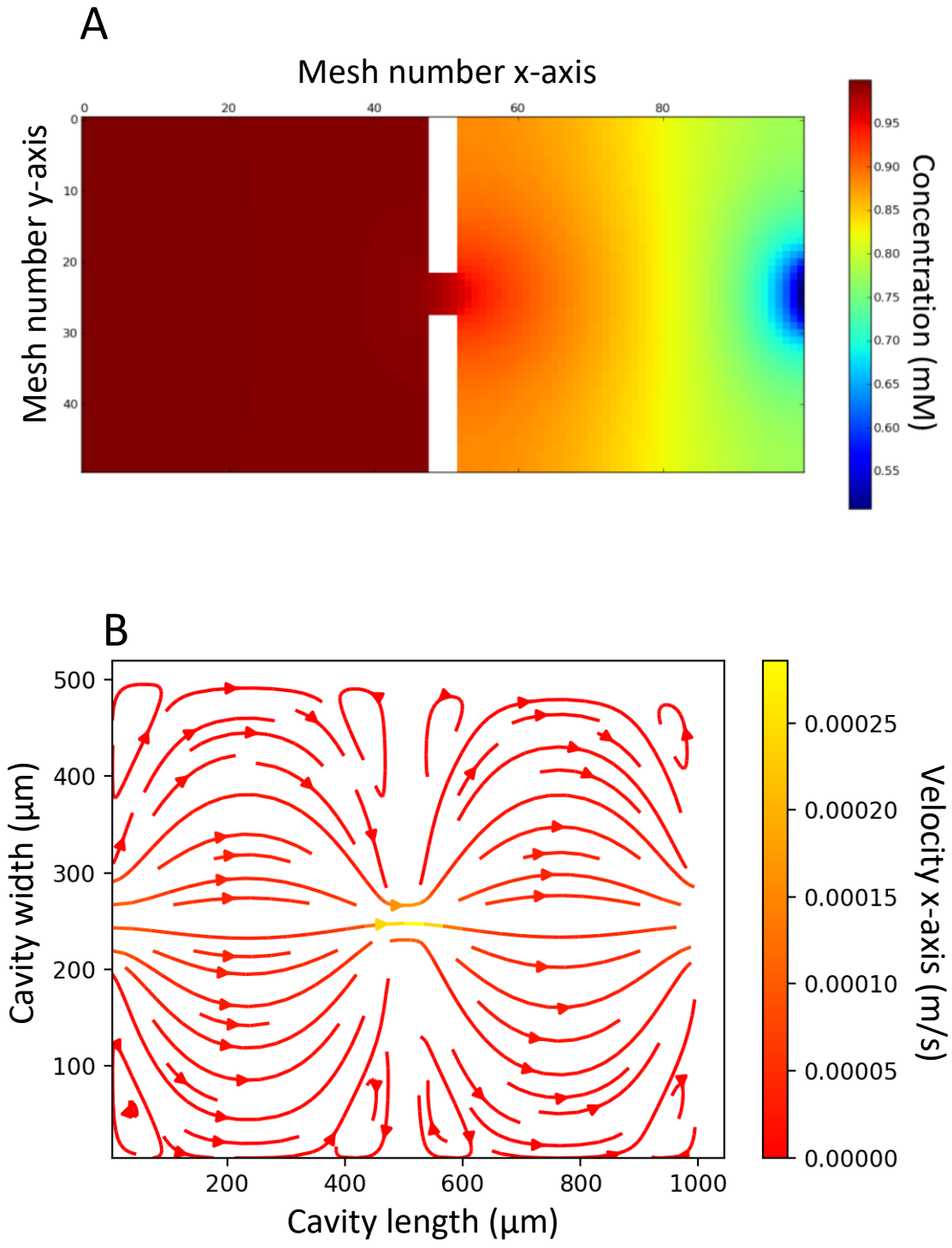


Figure 61. Results of the simulation of a geometry with two successive cavities. (A) results for the concentrations. (B) Associated velocity field.

5.3. Integration of a biological reaction

This last test consisted in demonstrating the interest of the RADE model in the context of the design of a lab-on-a-chip. The case study was quite close to that described in paragraph 3 of this chapter. We used the same biological reaction model, with penicillin contained in the inlet flow reacting with penicillinase immobilized at the bottom of the cavity. As before, this reaction released H^+ ions that modified the pH locally. Two ISFETs were used to monitor local

pH at two different points in the cavity, one located very close to the inlet and the other just behind the penicillinase deposit. The overall model was the same as in Figure 43, except that we replaced the diffusion models with advection-diffusion models. An example of simulation results, *i.e.*, a pH map of the cavity, is given in Figure 62. It can be used to predict the optimal position of the sensors, the size and position of the enzyme deposit, optimize the geometry of the sensor, or predict its evolution over time.

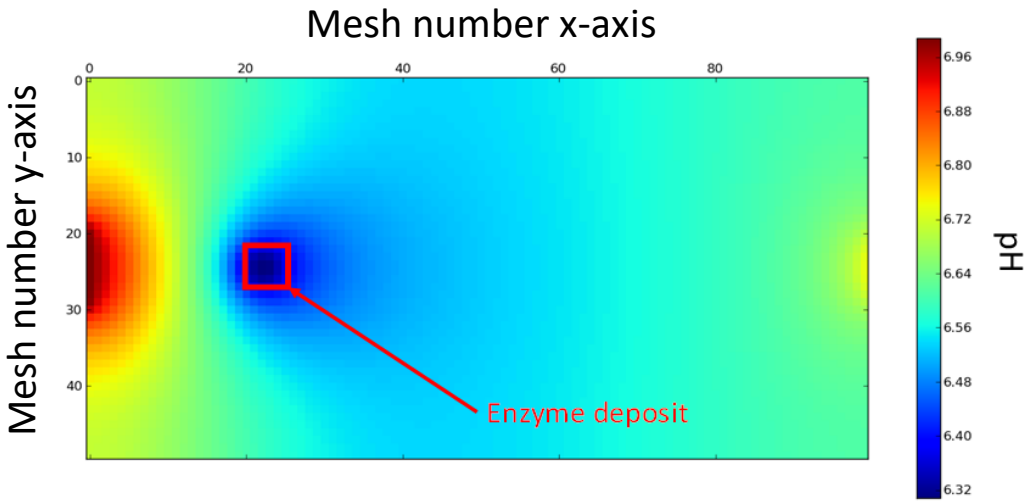


Figure 62. Result of the simulation based on a new mesh model, including a biological reaction.

6. Implementation in the final module

Resuming the example presented at the end of the previous chapter, we added the finite difference simulator described in this chapter to the mixer model to allow for low-abstraction modeling in our complete lab-on-a-chip model (Figure 63). .

The sigmoid parameters at the outlet of the mixer provided the distribution of the concentrations across the width of the channel for us to define the input parameters of the simulator described throughout this chapter.

The nodes representing our biosensor were linked to a biosensor model, as in our previous simulations (see paragraph 3). Then, the biosensor model sent the result back to the controller.

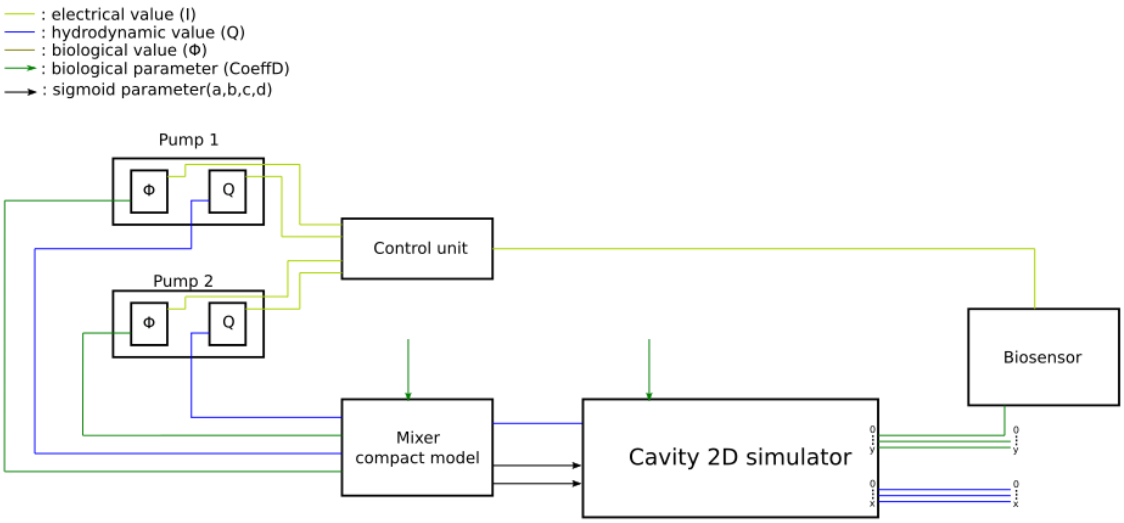


Figure 63. Schematic representation of the simulation of a lab-on-a-chip including the compact model for the mixture, followed by a simulation of finite differences in a cavity containing the biosensor.

7. Conclusion

In this chapter, we developed a low-abstraction simulator using finite differences in a microelectronics simulation environment.

First, we addressed the simulation of diffusion and the various results obtained by coupling our simulation model of diffusion with biological models and biosensors.

Then, we improved this simulator by adding advection to our diffusion model.

Remaining in the same simulation environment allowed our model to communicate perfectly with the different models developed in the previous chapters.

Our new model incorporating advection will need to be validated in a 3D space using a refinement procedure, as we did with the diffusion model.

Chapter 5

Case study: development of a microfluidic PCR system

In order to validate our SPICE modeling / simulation approach based on an experimental case, we developed a highly multiphysics lab-on-a-chip. We chose the polymerase chain reaction (PCR), which couples microfluidics, electronics, biology and heat science.

We are first going to present the principle of this lab-on-a-chip which aims to optimize PCR, which is a classical method of multiplying DNA strands.

Then, we will show the different parts of the development that led to a functional tool, as well as the thermal model that we developed to be added to our complete model.

In a third part, we will show the compact model associated with the biological PCR reaction.

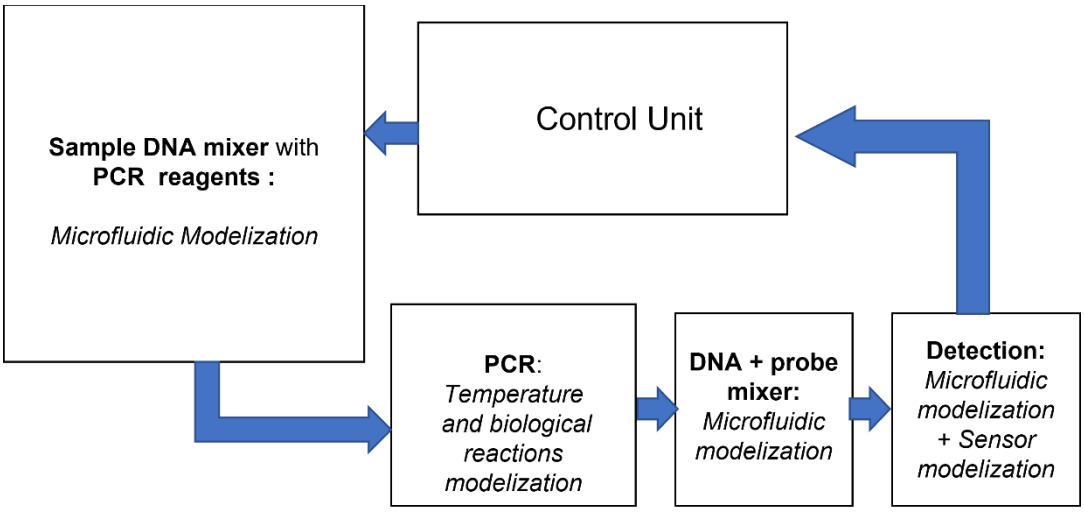
Finally, we will explain the improvements required for our lab-on-a-chip to be fully modeled using our simulation tools.

1. Objective

The objective of this last section was to produce a biosensor in which the tools presented above would be applied. We planned to build a first version to experimentally validate our models, and then to use the CAD environment to propose and test improvements.

The choice of the type of lab-on-a-chip was made based on the tools and skills available within the team. We chose a classical biological reaction relatively well known in biology – PCR. Based on existing work [7] [15] [84], we turned to microfluidic PCR to have continuous control of the hydrodynamic and biological variables. Microfluidic PCR also has the advantage of combining four areas: biology, microfluidics, electronics, and heat science. The first prototype was developed through educational projects with first-year students of the IT-Health sector of "Télécom Physique Strasbourg". They are presented in this chapter.

This work enabled us to represent the virtual prototyping of a theoretical lab-on-a-chip. The lab-on-a-chip would be a chain of mixtures including the PCR mix, aimed at detecting a specific gene in a DNA sample. It can be summed up in the following diagram:



Sample mixing as well as marker mixing could be modeled with the mixer presented in Chapter 3. The PCR was modeled as presented in this chapter. Finally, the cavity containing the biosensor was modeled with the simulation tools presented in Chapter 4.

2. Theoretical principles

2.1. General presentation of the PCR

PCR is a technique leading to the *in-vitro* amplification of a specific DNA sequence. From a sample containing a lot of different DNA sequences in very small quantities – *e.g.*, a blood drop –, the PCR quickly duplicates certain targeted DNA sequences in a great enough number for them to be detected, and above all to distinguish them from other non-targeted sequences also present in small quantities.

The term PCR has become very popular since the outbreak of the 2020 health crisis because the technique is used for detecting SARS-COVID-2. After extracting viral RNA from a pharyngeal sample, the test transcribes RNA into DNA. This DNA is amplified by PCR using a

sequence specific to the DNA sequence transcribed from SARS-COVID-2 RNA, in order to have enough DNA for detection [85].

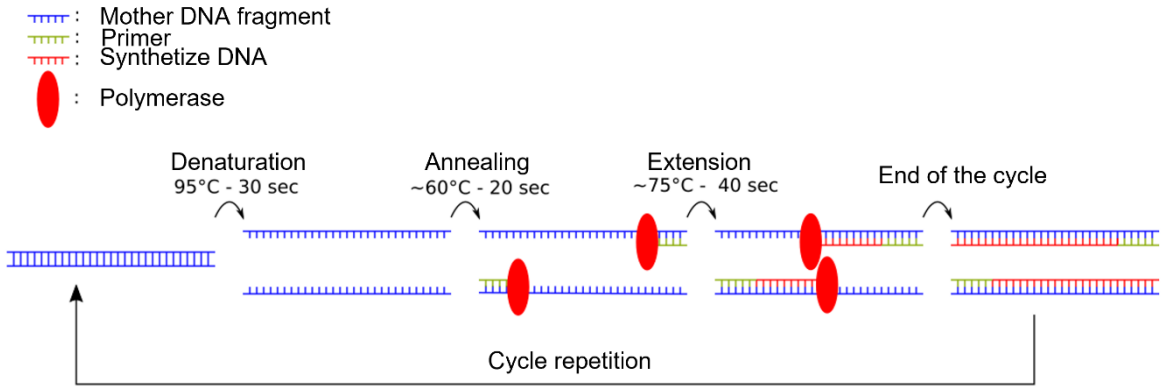


Figure 64. Schematic representation of the different PCR steps.

The principle of the PCR is based on DNA replication *via* temperature cycles. Each run includes three steps: denaturation, primer pairing (or hybridization), and extension (Figure 64).

The first step is denaturation. At room temperature, DNA is double-stranded. However, the hydrogen bonds that hold these two strands together weaken as temperature rises. Denaturation occurs between 90° and 95° [86]. It takes 10 to 15 minutes during the first cycle but is much faster during the following ones (around 30 seconds).

The second step is the pairing (or hybridization) between the single DNA strands and the primers. Primers are short DNA sequences complementary to the start and end of the DNA sequence to be amplified. They serve as a starting point for the synthesis of the complementary strand. When temperature is lowered, the primers bind to single-stranded DNA. The temperature allowing the binding of the primers to the single DNA strands ranges between 55°C and 65°C. This step lasts 2 to 60 seconds [87].

After hybridization, DNA is double stranded at the level of the primers and single stranded elsewhere. The last step of the PCR – extension – completes the molecules to produce double-stranded DNA again. An enzyme called polymerase binds to the end of the primer and fixes the complementary nucleotides to the initial sequence one after the other until a complete double-stranded DNA strand is obtained. This step is carried out at a temperature of approximately 72°C – the activation temperature of the enzyme – and lasts between 4 and 120 seconds, depending on the size of the strand to be synthesized.

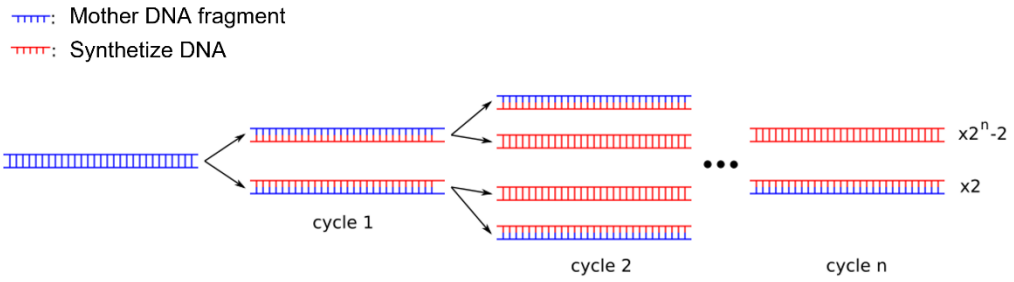


Figure 65. Schematic representation of DNA multiplication by PCR after n cycles.

Figure 65 shows the theoretical result achieved after n temperature cycles. If all the steps go perfectly well and the reagents are not in limiting quantities, the amount of DNA is doubled at each step, bringing the final amount to 2^n times the initial amount.

PCR cycles can include only two temperatures [88]: as pairing and polymerization take place at relatively similar temperatures, these two phases can be merged at an average temperature. This relatively common technique greatly facilitates the performance of the temperature cycles, but efficiency is lower because the reactions no longer take place at optimal temperature for the reaction rates. For example, if pairing takes place at a higher temperature than expected, the primers will not hybridize on 100% of the DNA strands, and final amplification will be less than 2^n.

2.2. Microfluidic PCR

Figure 66. Picture showing an open thermocycler. The samples inside the PCR microtubes are heated at the level of their caps when the upper part of the thermocycler (in white on the picture) is shut onto the tubes.

PCRs are conventionally done using a laboratory instrument called a thermocycler (Figure 66), but can also be carried out with a microfluidic system.

Several types of microfluidic PCR systems are already available, with two different types of temperature regulation strategies.

A first technique consists in immobilizing the fluid in an area where the temperature is modulated to obtain temperature cycles. As showed in Figure 67 taken from [89], a pump controls the speed of the sample movements submitted to the temperature cycles under the control of a Peltier module, at the level of the aluminum part. Then, the sample is directed to an area with a nanoprism – the biosensor; it works using surface plasmon resonance (SPR) (see Chapter 1, paragraph 2.4), and detects DNA.

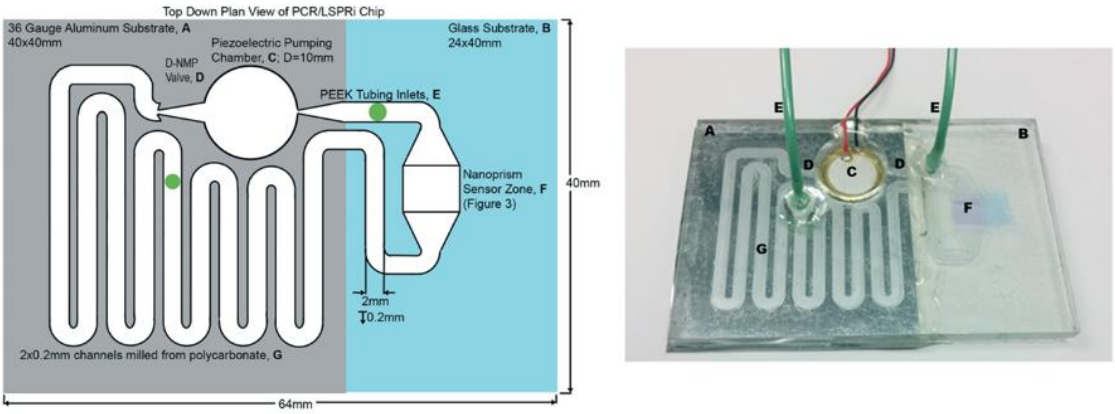


Figure 67. The aluminum substrate A is used for the contact with the Peltier thermocycler; the glass substrate B is used for the optical transmission measurements across the area of the nanoparticle sensor. The channels made from milled polycarbonate are glued to the aluminum and glass substrates by an engraved dry film adhesive[89].

A second technique aims to move the DNA sample from one temperature zone to another during the temperature cycles. Thus, as showed in Figure 68 taken from [90], the sample, in the form of drops, is directed to a micro-reactor with two zones of fixed temperature. The drops rotate in a circle from one area to another for the sample to follow the temperature cycles of the PCR.

We drew our inspiration from this last technique to carry out our microfluidic PCR. We created a system made up of two temperature-regulated zones and a microfluidic channel allowing the PCR mixture to circulate between these two zones. This mixture was called a premix and contained the DNA sample, the polymerase, nucleotides, and a pH buffer to maintain the acidity of the medium. In practice, we built this microfluidic channel using a capillary.

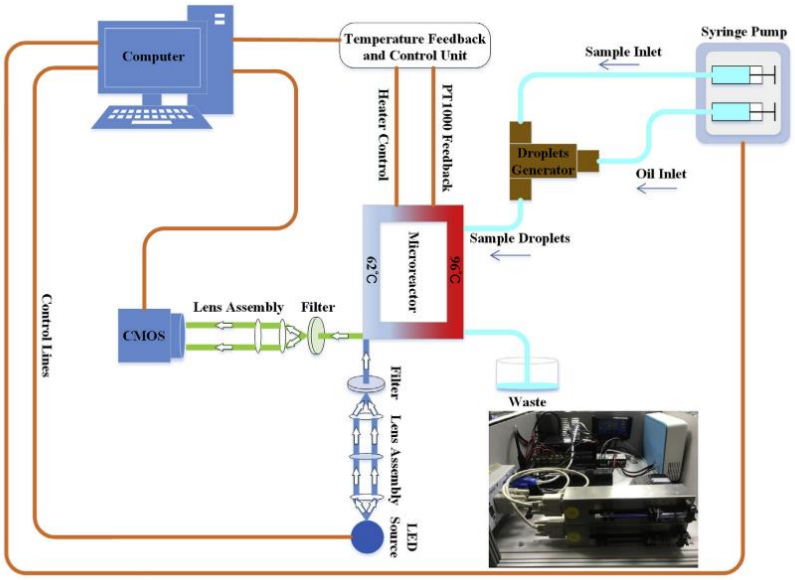


Figure 68. Schematic illustration of the architecture of a continuous-flow digital PCR system. The samples are injected through the droplet generator, and then fed by the microreactor while fluorescence intensity is measured by the optical module. [90]

The device is described in Figure 69. It consisted of two zones at fixed temperatures (95°C and 72°C) heated by a heating mat. These two areas were separated by thermal insulation (Figure 69B). Heating mats were used to bring an aluminum plate to the correct temperature (Figure 69C) equipped with small stops to guide the Teflon capillary tube. Then, the tube was wrapped around the device (Figure 69E). The capillary tubes were surrounded by a thermal interface pad (Figure 69D & F) allowing better heat transfer between the aluminum plate and the capillary. Finally, the entire device was sealed with a final aluminum plate to avoid potential temperature differences in the capillary and hold the capillaries in place (Figure 69G & H).

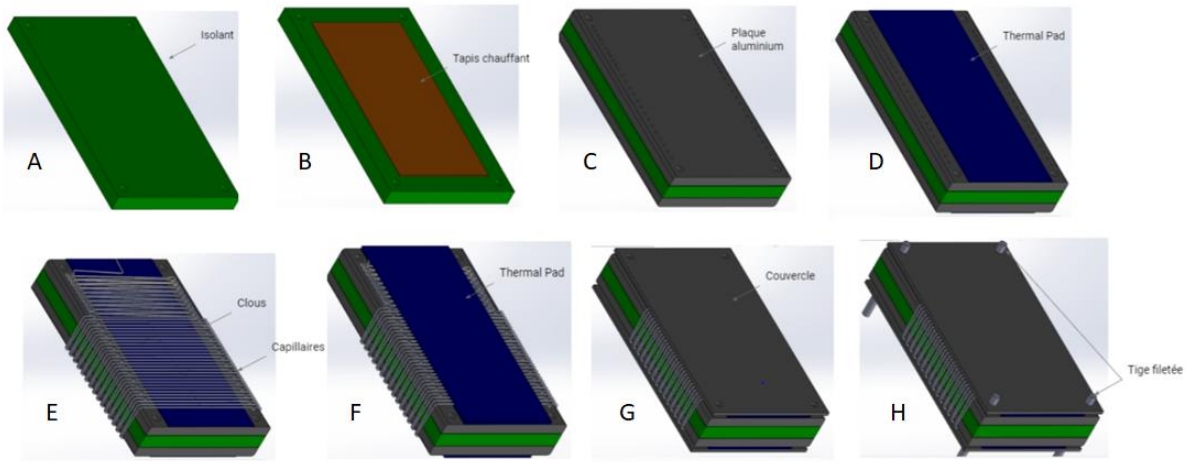


Figure 69. SolidWorks® representation of our microfluidic PCR setup.

The entire system was temperature controlled *via* temperature sensors and a controller integrated in an Arduino®.

First, we designed a thermal model to size the device, for it to be an integral part of the complete lab-on-a-chip model.

3. Thermal sizing of the device

3.1. Thermal model of the overall device

The thermal model of each face of the device is described in Figure 70. The heat transfers between the different components of the device (heating mats, capillary, aluminum plate, cover) are represented by red arrows, while the thermal losses are represented by blue arrows. The objective was to determine the powers to be supplied to the system for the capillary to be brought to the desired temperature, taking time constraints to reach thermal equilibrium into account.

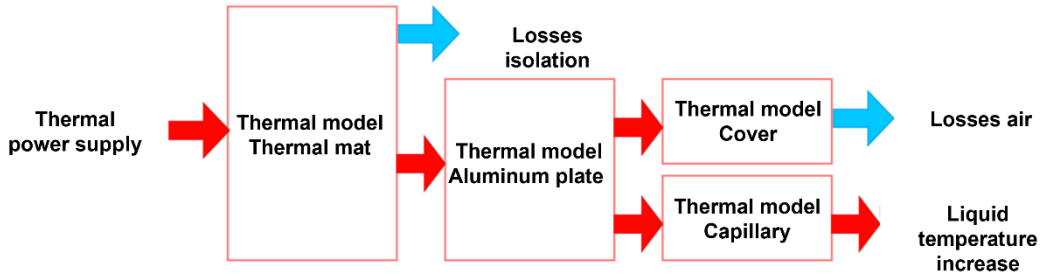


Figure 70. Thermal modeling of the system.

The equivalent electrical model of this system is showed in Figure 71, and the correspondence between the names of the components and their role is given in Table 4. The plate was considered much longer and wider than thick, and thermal conduction within the aluminum block was neglected.

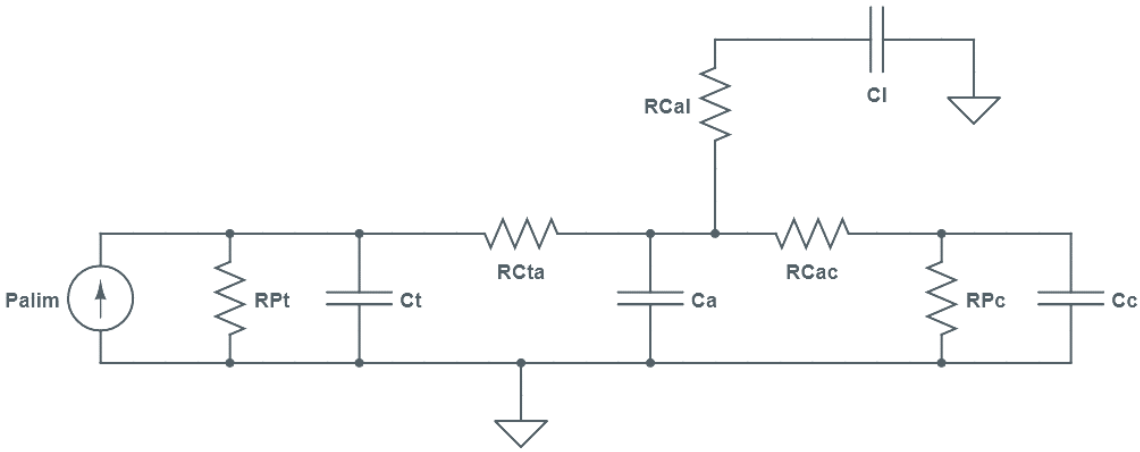


Figure 71. Electrical model equivalent to our thermal model.

Table 4. Legends of Figure 71.

P_{alim}	Thermal power supplied by the power supply
RP_t	Heat resistance loss by convection between the mat and air
C_t	Thermal capacity of the mat
RC_{ta}	Thermal conduction resistance between the mat and the aluminum plate
C_a	Thermal capacity of the aluminum block
RC_{ac}	Thermal conduction resistance between the aluminum plate and the cover
RP_c	Heat resistance loss by convection between the cover and air
C_c	Heat capacity of the cover
RC_{al}	Thermal conduction resistance between the aluminum plate and the liquid in the capillary
C_l	Thermal capacity of the liquid in the capillary

3.2. High-level thermal model of the aluminum plates

3.2.1. Static model

We calculated the power needed to bring the aluminum plate to the desired temperature. The model presented in Figure 70 was simplified as follows:

- The thermal conductions were all ideal; the conduction resistances were zero.
- The thermal capacity of the mat, the liquid and the cover were negligible compared to that of the aluminum block.

Under these conditions, at equilibrium, the thermal power supplied to the aluminum block by the heat source was offset by the thermal power dissipated by the block. Considering an aluminum block measuring 12 x 7 x 0.6 cm, the dissipated power was the sum of three contributions:

- Conduction losses, which were zero according to the assumptions.
- Losses by convection, whose power was modeled by the following law:

$$W_c = h \cdot S \cdot (T_A - T_{ambient}) \quad 72$$

Where h is the thermal convection coefficient (W/m²/K) – an empirical parameter whose order of magnitude is 10 W/m²/K for air –, S the contact surface (m²), and T_A temperature (K).

- Radiation losses, whose power was modeled by Stefan's law (these losses are not shown on the equivalent electrical diagram):

$$W_r = \varepsilon \cdot \sigma \cdot S \cdot T_A^4 \quad 73$$

Where ε is the emissivity of the material (unitless and equal to 1 for a black body), and σ is the Stefan-Boltzmann constant (= 5.67x10^{-8} W/m²/K⁴).

First, we calculated the amount of heat Q needed to heat the aluminum plate to a temperature $T_{setpoint}$. This amount of heat was:

$$Q = M * C_p * \left(T_{request} - T_{ambient}\right) \quad 74$$

where M (kg) is the mass of aluminum, and C_p the specific heat capacity of aluminum (897 J/Kg/K).

This value provided an approximation of the thermal power W_{in_0} to be supplied to the aluminum block during a heating time $t_{heating}$ to raise the temperature of the aluminum block ($W_{in_t} = \frac{Q}{t_{heating}}$). In practice, the temperature reached after a time t_{heat} was lower than $T_{setpoint}$ because of the losses.

We also calculated the thermal power to be supplied to the system to maintain it at $T_{setpoint}$.

Based on Equations (1) and (2), the ratio between radiation losses and convection losses was written:

$$\gamma = \frac{\varepsilon \cdot \sigma \cdot T_{setpoint}^4}{h \cdot \left(T_{setpoint} - T_{ambient}\right)} \quad (75)$$

This ratio was less than 0.1 from 35°C for h = 10 W/m²/K and $\varepsilon = 0.03$ for aluminum [91]. Therefore, the losses by convection were negligible.

3.2.2. Dynamic model

Based on the equivalent electric model and working hypotheses, we established the differential equation governing the temporal evolution of the temperature of the aluminum plates:

$$M \cdot C_p \cdot \frac{dT_A}{dt} = -h \cdot S \cdot (T_A - T_{ambient}) + W_{in} \quad (76)$$

This differential equation is that of a first-order system with a second member [92]. Therefore, the change in temperature T over time was expressed as follows:

$$T(t) = T_{ambient} + \left(T_{final} - T_{ambient}\right) \cdot \left(1 - \exp\left(\frac{-t}{\tau}\right)\right) \quad (77)$$

with

$$\tau = \frac{M \cdot C_p}{h \cdot S}$$

and

$$T_{final} = \frac{W_{in}}{h \cdot S}$$

3.2.3. First dimensioning

Thanks to the high-level model explained above, we sized the thickness of the aluminum plate and the power levels to be supplied by the heating mats for implementing the system.

A 6-mm thick plate was chosen. It seemed a good compromise between rapid heating and a certain thermal inertia giving the system less sensitivity to external disturbances. In addition, by fixing h = 5 W/m²/K, with a 12 x 7 x 0.6 cm aluminum plate with a density of 2,700 kg/m^3, we estimated losses by convection to be 6.66 W at 95°C. Therefore, the heating mats had to provide at least this power to maintain the temperature of the plate. A power of at least 10.17 W was needed to obtain 95°C in 15 min, neglecting losses. A heating mat capable of supplying 20 W was considered sufficient to heat the system within the required time.

3.2.4. Experimental validations

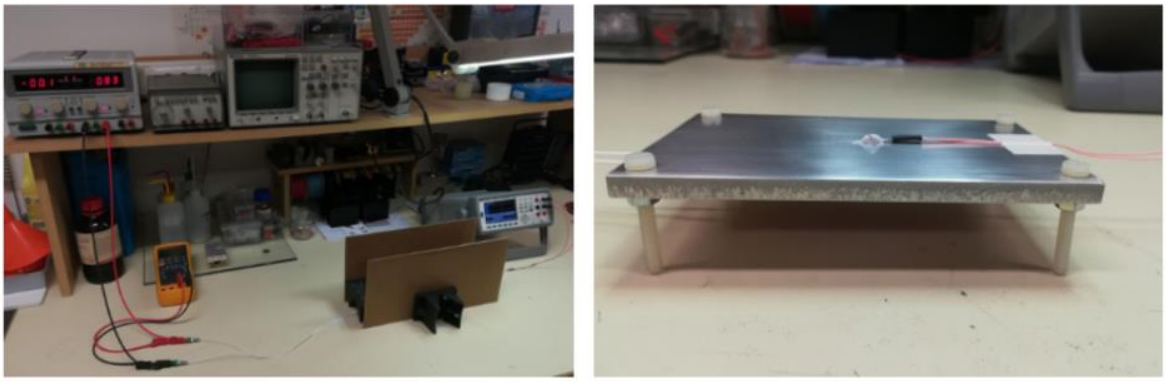


Figure 72. Photographs of the first experimental setup.

A 120 × 70 × 6 mm aluminum plate was heated by means of a 100 × 50 mm heating mat located on the underside. The mat was supplied by a laboratory power supply. A pt100 temperature probe was placed on the surface of the plate (Figure 72). Twenty W were injected into the system to reach the desired temperature faster. The response of this system was first order, with a final temperature of 126°C (Figure 73). The settling time was 400 seconds for 72°C and 878 seconds for 95°C. We also observed a slight delay (of a few seconds) of the system (Figure 73B). These 5 sec delays were negligible compared with the 15-min heating time. The delay was mainly explained by the conduction effects (between the mat and the aluminum plate, then within the aluminum plate), which were neglected.

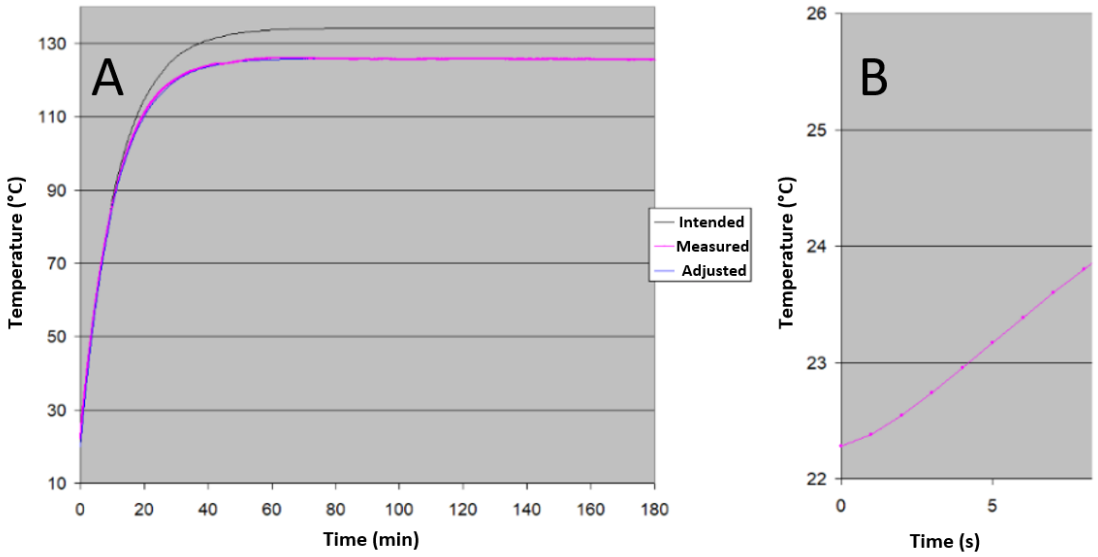


Figure 73. Comparison of experimental results with expected results. (A) Temperature curves as a function of time. (B) Focus on the first seconds of heating.

We fitted the experimental curve (Figure 73A) with the theoretical model (equation 77), leaving the value of the convection coefficient as degree of freedom. We obtained 5.7 W/m²/K, which was consistent with theory.

3.3. Advanced thermal model of the complete system

In the high-level model, we were interested in an insulated aluminum plate, considering it as an ideal conductor and without considering the other elements of the system (capillary, cover, possible interaction between the two aluminum plates). In the present section, we are going to use a lower-level model based on partial derivative equations to address these two questions.

3.3.1. Model of the aluminum plate with conduction

Thermal conduction in aluminum is supposed to be perfect, hence instantaneous propagation of heat inside the trays and a uniform temperature. Experience has showed that the propagation of heat from the underside in contact with the heating mat to the top was rapid but not instantaneous [93]. However, non-infinite conductivity can cause temperature inconsistency in the horizontal plane: the losses on the side faces could lead to a lower temperature on the edges of the upper face than in its center.

RDM software [94] was used to study diffusion within the aluminum plate.

The simulated device is depicted in Figure 74. It consisted of a 120 x 6 mm aluminum plate and a 100-mm heat source on its underside representing the heating mat. RDM simulated the conduction phenomena in solids but also dissipation by convection and radiation. The simulation (Figure 74) showed that the lack of uniformity on the upper face remained low: the difference between the center and the edges was 0.4°C.

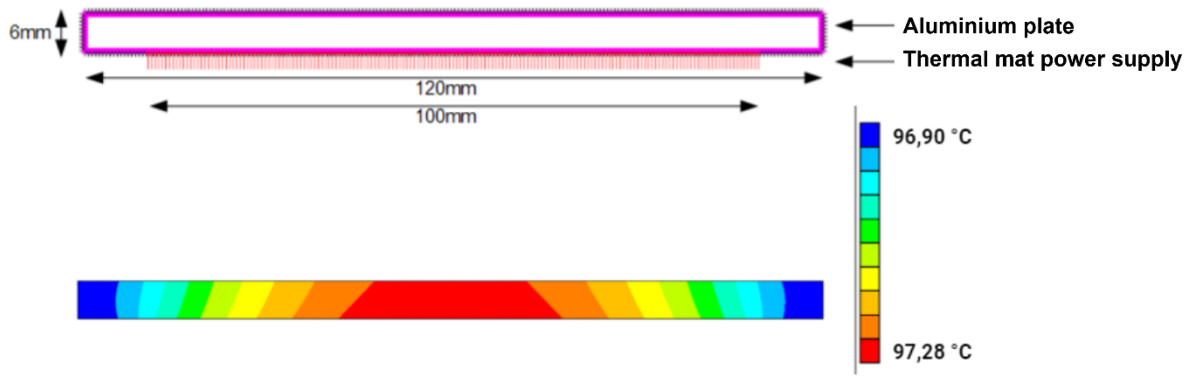


Figure 74. Schematic diagram of the simulation system and its results.

3.3.2. Model of the insulator between the aluminum plates

We studied the interaction between the 95°C plateau and the 72°C plateau. We took an insulating plate with two trays around it. The simulated system is described in Figure 75, which presents a diagram of the simulation of the insulation device of the two temperature zones and the corresponding results . The system consisted of an aluminum plate heated to 95°C, an insulator, and another aluminum plate. Everything was thermally insulated. This simulation made it possible to measure the impact of the heat of the top plate on the heat of the bottom plate and to size the insulation material. Several insulating materials were tested. Our final choice was glass wool (Figure 75). Glass wool provided a temperature difference of about 50°C, which was more than double the difference required for our system.

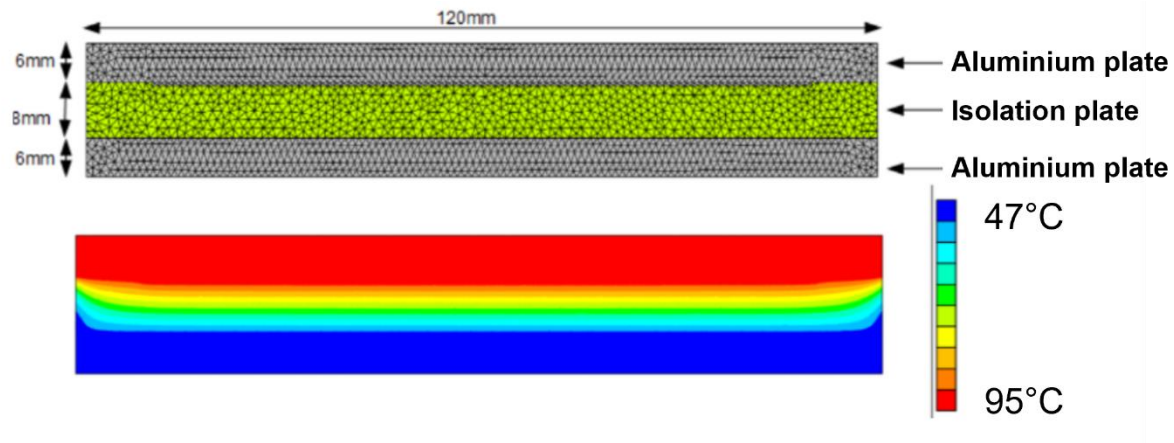


Figure 75. System diagram of the simulation of the insulation of the two temperature zones, and simulation results.

3.3.3. Study of the complete system

The last simulation concerned the complete system (Figure 76). As the system was almost symmetrical by horizontal translation, we only simulated a slice of the board to limit the necessary computing power.

Based on this simulation, 3.48 W were needed to bring the system to the desired temperatures, and 1.46 W in the bottom plate. This simulation was also used to identify the dynamic parameters of the system so as to calculate the regulation to be implemented. First, we sent 20 W to each plate, assuming that the initial temperature was 20°C. We reached 95°C in 650 s.

On the other hand, starting from the regulated situations at 95°C and 72°C, we sent a 0.2-W step in the vicinity of the necessary powers in order to measure:

• the time constant of the system in the vicinity of the working temperature

• dynamic thermal resistance in °C/W near the operating point

• the system delay.

For the plate at 95 °C (72 °C), the time constant was 1,279 s (1,279 s), and thermal resistance was 16.22 °C/W (16.44 °C/W). Thermal resistance only depended on the material and its geometry, so that finding 2 identical values was coherent. Delay was 4.75 s (6.20 s), which was considered negligible.

The thermal simulations also showed that only a few tenths of a second were necessary for the capillaries to reach the temperature of the plate. On the other hand, the temperature decrease in the vicinity of a capillary entering the system was negligible. Based on these observations, we considered that the heating of the capillary, the liquid and the plate was altogether instantaneous.

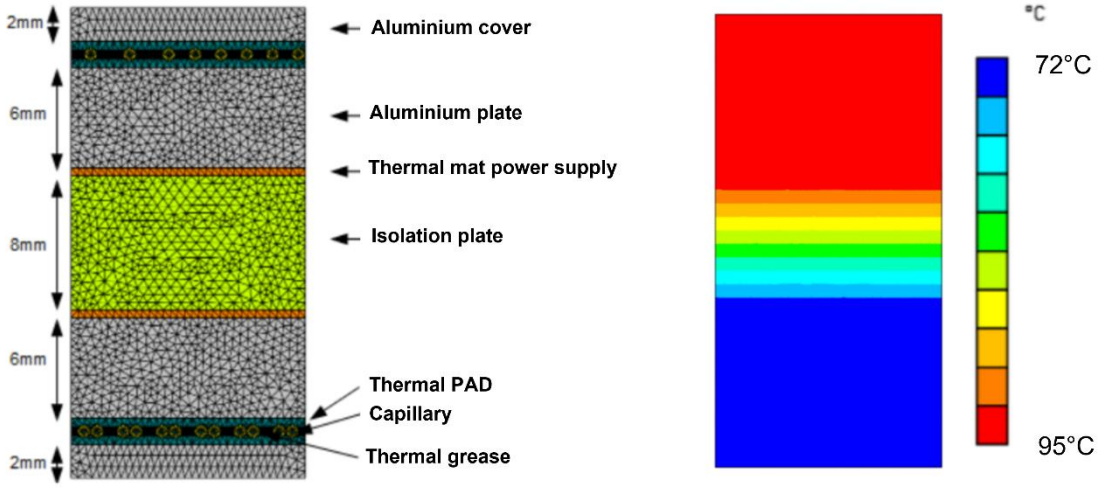


Figure 76. Simulation of the distribution of temperature across the complete system.

3.3.4. *Assessment of thermal modeling*

Thermal modeling allowed us to clarify many points before carrying out actual assembly:

- The thermal resistance of aluminum was negligible, so we considered the temperature of the plate to be uniform.
- The capillaries in the thermal pad did not prevent the propagation of heat to the cover.
- An 8-mm thick glass wool type insulator was necessary to limit interactions between the 2 plates.
- The holding powers were approximately 3.48 W for the 95 °C plate and 1.46 W for the 72 °C plate.
- Fluids flowing through the capillaries warmed up to plate temperature in less than one second.
- The complete system was modeled as a first-order system with pure delay. We determined the time constant and resistance at the operating point. We also established that the delay could be neglected.

3.4. Design of the temperature control system

3.4.1. *Regulatory strategy*

The control system is showed in Figures 77 and 78. It was made up of two heating mats and two LM35-type temperature sensors. The sensors provided an analog voltage of 10 mV/°C, which was acquired by an Arduino board *via* its analog-to-digital converters. The heating mats were controlled by a digital signal generated by the converter *via* a MOSFET (IRF520).

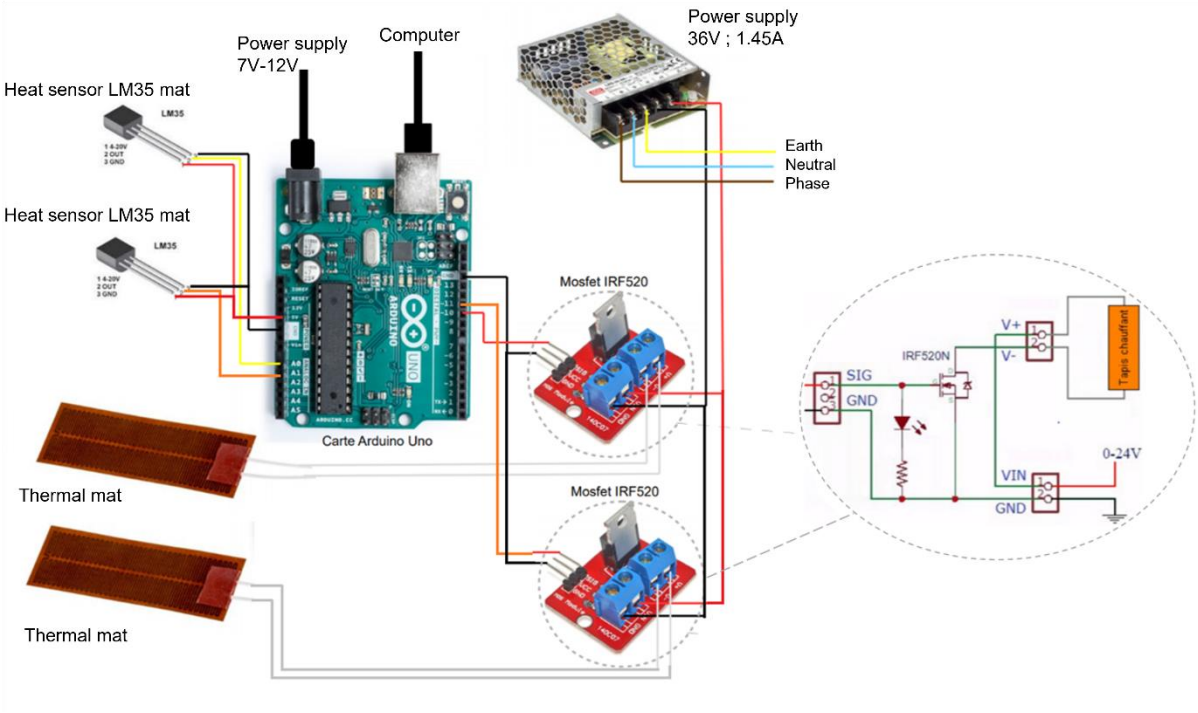


Figure 77. Visual electronic diagram of the full setup for regulating the temperature of the system.

The regulation strategy consisted in initially sending a high power to the heating mats so that they quickly reached the pre-set temperature, and then to maintain them at their pre-set temperature using the measurements of the temperature sensors.

Two types of correctors were studied: an all-or-none one, and a PI corrector in which control is done *via* pulse width modulation (PWM). The PI corrector with PWM output was not satisfactory during the experimental tests, while the on/off one showed satisfactory first results for a first application.

Therefore, we temporarily left the PI corrector aside, and kept the on/off one. However, the PI corrector remains a possible solution to improve the results obtained under on/off temperature regulation.

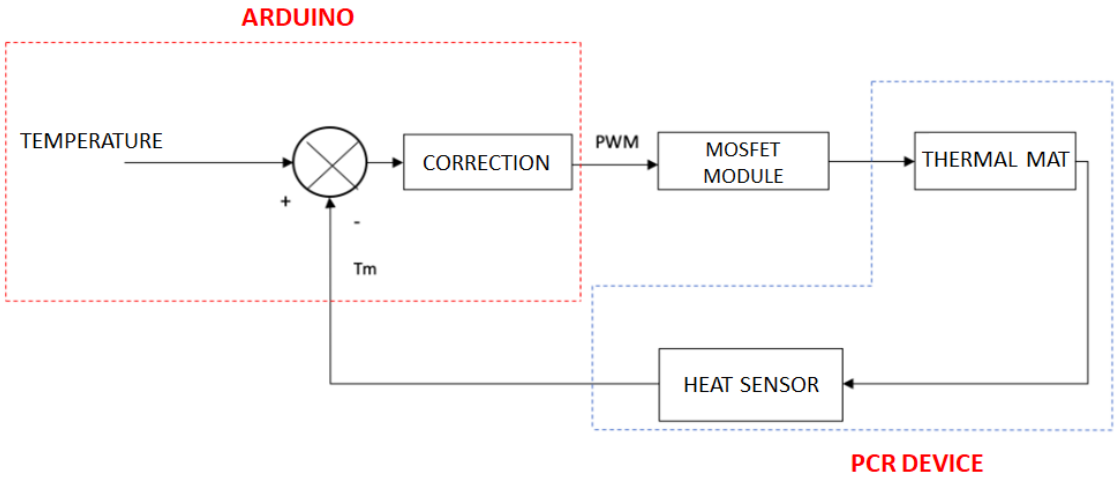


Figure 78. Block diagram of temperature maintenance of the PCR system.

3.5. Temperature regulation

Figure 79 shows the final setup of the microfluidic PCR with the temperature control system.

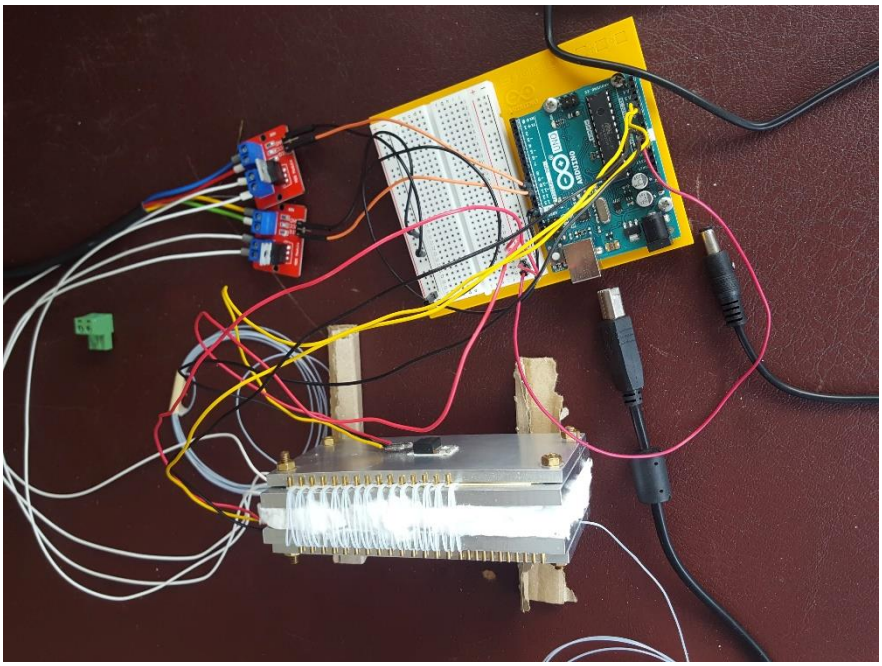


Figure 79. Final setup for temperature regulation tests.

The results under all-or-none regulation are showed in Figures 80 and 81. The pairing and extension temperatures used to test the thermal model were chosen based on the literature. Thus, the study at 72°C was based on a general approximation of the temperature values in the literature.

The provided samples had an average pairing + extension temperature of 65°C. Therefore, a low temperature of 65°C was tested to suit the restrictions of the pre-mix sample. This sample is presented in paragraph 3.6.1.

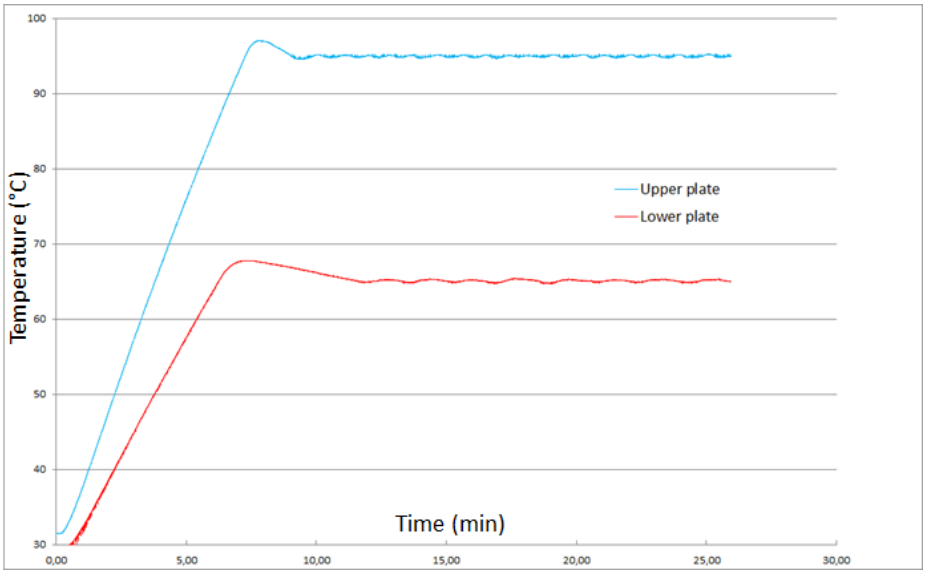


Figure 80. Graph of temperature monitoring in true-life conditions with water in the microfluidic channels.

Figure 80 shows that the initial heating system could be improved to avoid overheating during the initial temperature rise. This could be done by gradually decreasing initial heating as one approaches the pre-set temperature which loses about 6 min on starting. However, the starting time remained below the time we imposed in the specifications (15 min).

Figure 81 shows temperature variation around an average temperature of 94.96 °C and a standard deviation of 0.12 °C.

The results obtained during the fixed temperature phase were satisfactory enough for us to consider tests with genetic material to validate the proof of concept.

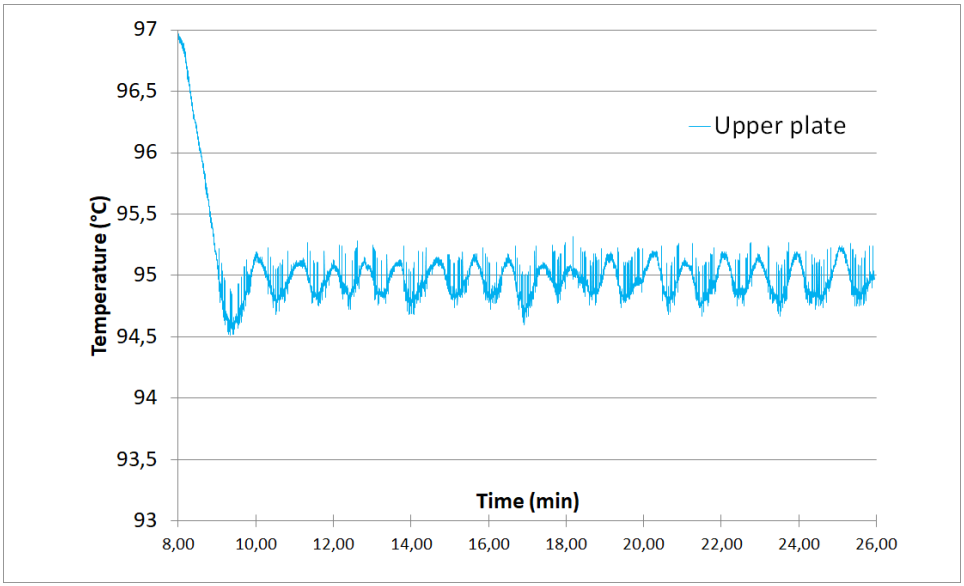


Figure 81. Zooming on the zone of temperature maintenance at 90 °C.

3.6. Experimental results

3.6.1. Protocol

These tests were carried out in collaboration with the research team of the Laboratory of Biotechnology and Cellular Signaling of Strasbourg (Pr. Bruno Chatton), who provided us with a premix containing the genetic material needed to perform PCRs. Phase 1 was the denaturation phase at 95°C, and phase 2 was the pairing + extension phase at 65°C.

Table 4. Synoptic table of the experimental protocol of the microfluidic PCR.

Number of cycles	25
Phase 1 time (min)	1
Phase 2 time (min)	2
µFlu channel length Phase 1 (mm)	58
µFlu channel length Phase 2 (mm)	116
Channel section (µm²)	50,670
Fluid flow (µL/min)	2.93

Table 5 shows the parameters of the experimental protocol that we applied for the PCR test with our device. The same time and temperature values were used in a "classical" PCR to compare the results.

The results of the two PCRs were characterized by electrophoresis, which is a gold standard of DNA characterization [95]. A pH buffer was added to the DNA solution to charge the sample negatively; the buffer also prevented potential modification of the number of charges on the DNA following pH variation. Each sample was deposited on an agarose gel. Then, an electric field was applied to either side of the gel, causing the DNA to migrate to the anode. The gel through which DNA migrated acted as a sieve: larger DNA strands migrated more slowly to the anode than shorter strands did. This made it possible to characterize the size of the DNA strands in a sample. Once migration was complete, the gel was immersed in an ethidium bromide bath. This DNA intercalator fluoresces under ultraviolet radiation and reveals the DNA present in the gel.

3.6.2. Results

The results are visible on the photograph of the electrophoresis gel (Figure 82). In this photograph, each of the white bands corresponds to a DNA sample of the same length. The molecular weight ladder is a mixture of different strands of DNA of known length; it allowed us to determine the size of the DNA strands in our sample. Figure 82 groups together two electrophoresis results performed separately (one following microfluidic PCR and one following PCR in a conventional thermal cycler). This explains why the molecular weight ladder did not migrate in the same way on the left / in the middle and on the right.

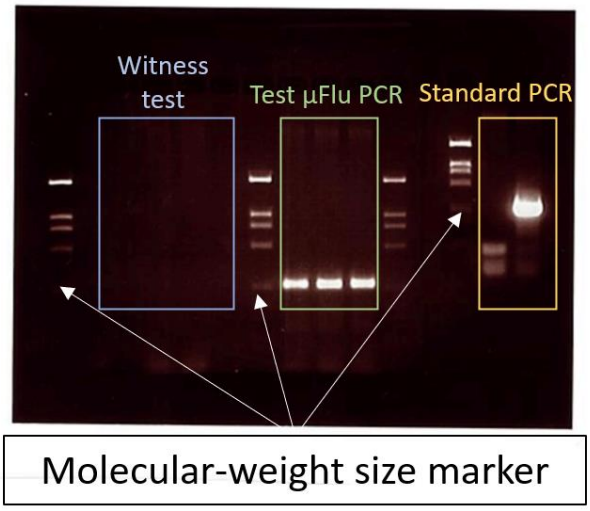


Figure 82. Photograph of the electrophoresis gels. Blue box, negative control; green box, microfluidic PCR results; yellow box, "classical PCR" results.

The light intensity of each of the white bands represents a DNA concentration. We did three tests with our microfluidic PCR method and one test with the classical method to compare results. DNA migrated up to the same scale of the molecular weight ladder in both cases. This proved the efficiency of the microfluidic PCR amplification. Yet, the intensity of the bands was much weaker in microfluidics PCR than it was in conventional PCR, suggesting that

amplification was less efficient. We estimated the loss of amplification gain to range between 100 and 1,000.

4. Modeling of the PCR reaction

The second modeling step concerned the biological reaction. We wanted to create a dynamic temperature-dependent model to describe the PCR reaction [96].

4.1. Balance sheet equations and dynamic models

Below are the equations of the different PCR phases before presenting the general equations.

4.1.1. Denaturation phase

As a reminder, this phase separates double-stranded DNA into single-stranded DNA. Its chemical equation reads:

$$D \underset{k_{-1}}{\overset{k_1}{\rightleftarrows}} 2\,S \tag{78}$$

where D is double-stranded DNA (mol/L) and S single-stranded DNA (mol/L), k_1 (s^{-1}) the rate of the reaction denaturing double-stranded DNA into two single-stranded DNAs, and k_{-1} (s^{-1}) the reaction rate of the hybridization of two single-stranded DNAs into one single-stranded DNA.

The dynamics of this reaction was described by the following differential equations:

$$\frac{d[D]}{dt} = -k_1 \cdot [D] + k_{-1} \cdot [S]^2 \tag{79}$$

$$\frac{d[S]}{dt} = 2 \cdot (k_1 \cdot\!* [D] - k_{-1} \cdot\!* [S]^2) \tag{80}$$

4.1.2. Annealing phase

This phase corresponds to the annealing of DNA primers on single DNA strands to initiate the complementary synthesis of the single strand. This reaction was written:

$$S + P \underset{k_{-2}}{\overset{k_2}{\rightleftarrows}} SP \tag{81}$$

where P is the DNA primer and SP single-stranded DNA complexed with the bound primer, k_2 ($mol^{-1}.s^{-1}.L$) the pairing reaction rate of the SP complex, and k_{-2} (s^{-1}) the dissociation rate of the SP complex.

The dynamic model associated with this reaction was:

$$\frac{d[P]}{dt} = -k_2 \cdot [S] \cdot [P] + k_{-2} \cdot [SP] \tag{82}$$

$$\frac{d[S]}{dt} = -k_2 \cdot [S] \cdot [P] + k_{-2} \cdot [SP] \tag{83}$$

$$\frac{d[SP]}{dt} = k_2 \cdot [S] \cdot [P] - k_{-2} \cdot [SP] \tag{84}$$

4.1.3. Extension phase

During this phase, the polymerase enzyme binds to the complex formed by the primer and single-stranded DNA. Then, the enzyme reconstitutes the DNA strand complementary to the single strand, using the nucleotides necessary for polymerization. Therefore, two successive reactions occur. First, the binding of the enzyme to the primer + single-stranded DNA complex is represented by:

$$SP + E \underset{k_{-3}}{\overset{k_3}{\rightleftharpoons}} SPE \quad (85)$$

where E is the enzyme (mol/L) and SPE the primer + single-stranded DNA + polymerase complex, k_3 ($mol^{-1}.s^{-1}.L$) the rate of the reaction forming the SPE complex, and k_{-3}(s^{-1}) the dissociation rate of the SPE complex.

The second reaction is the extension of the DNA strand. This reaction is actually a series of small reactions that repeat themselves successively. Assuming that the binding of a nucleotide is a fast reaction in comparison to all the others, we used a simplified model for extension assuming that double-stranded DNA was formed using the SPE complex and p nucleotides. This reaction releases the polymerase. We also assumed that this reaction was irreversible and occurred at a rate k_4 depending on the nucleotide concentration. Based on this last hypothesis, we modeled a possible nucleotide deficiency at the end of the reaction:

$$SPE + p \cdot N \overset{k_4}{\longrightarrow} D + E \quad (86)$$

The combination of these two reactions translated numerically as:

$$\frac{d[SP]}{dt} = -k_3 \cdot [SP] \cdot [E] + k_{-3} \cdot [SPE] \quad (87)$$

$$\frac{d[E]}{dt} = -k_3 \cdot [SP] \cdot [E] + \left(k_{-3} + k_4([N])\right) \cdot [SPE] \quad (88)$$

$$\frac{d[SPE]}{dt} = k_3 \cdot [SP] \cdot [E] - \left(k_{-3} + k_4([N])\right) \cdot [SPE] \quad (89)$$

$$\frac{d[N]}{dt} = -p \cdot k_4([N]) \quad (90)$$

$$\frac{d[D]}{dt} = k_4([N]) \quad (91)$$

4.1.4. Complete biological model

Put end to end, equations 78 to 91 led to the dynamic model of the PCR in the form of a system of 7 differential equations coupled with 7 unknowns:

$$\begin{cases} \dfrac{d[D]}{dt} = -k_1 \cdot [D] + k_{-1} \cdot [S]^2 + k_4([N]) \\ \dfrac{d[S]}{dt} = 2 \cdot (k_1 \cdot [D] - k_{-1} \cdot [S]^2) - k_2 \cdot [S] \cdot [P] + k_{-2} \cdot [SP] \\ \dfrac{d[P]}{dt} = -k_2 \cdot [S] \cdot [P] + k_{-2} \cdot [SP] \\ \dfrac{d[SP]}{dt} = k_2 \cdot [S] \cdot [P] - k_{-2} \cdot [SP] - k_3 \cdot [SP] \cdot [E] + k_{-3} \cdot [SPE] \\ \dfrac{d[E]}{dt} = -k_3 \cdot [SP] \cdot [E] + \left(k_{-3} + k_4([N])\right) \cdot [SPE] \\ \dfrac{d[SPE]}{dt} = k_3 \cdot [SP] \cdot [E] - \left(k_{-3} + k_4([N])\right) \cdot [SPE] \\ \dfrac{d[N]}{dt} = -p \cdot k_4([N]) \end{cases} \quad 92$$

4.2. Coupling to the thermal model

The reaction rates were all temperature dependent, so the model was coupled to the thermal model. For example, considering Equation 78 at 95 °C, the reaction was more directed towards the formation of two single DNA strands from double-stranded DNA, while double-stranded DNA did not separate into two single DNA strands at 65°C.

4.3. Simulation Results

4.3.1. Denaturation

Hybridization of DNA molecules is often expressed as the ratio θ of the total number of double-stranded DNA to the total number of DNA molecules in the solution. θ is expressed as follows:

$$\theta = \frac{2 \cdot [D]}{[S] + 2 \cdot [D]} \quad 93$$

In steady state and based on Equation 78, we wrote:

$$[D] = \frac{k_{-1} \cdot [S]^2}{k_1} \quad 94$$

so

$$\theta = \frac{2 \cdot k_{-1} \cdot [S]^2}{k_1 \cdot [S] + 2 \cdot k_{-1} \cdot [S]^2} = \frac{1}{1 + \dfrac{k_1}{2 \cdot k_{-1} \cdot [S]}} \quad 95$$

This function obviously depended on temperature (*via* parameters k_1 and k_{-1}) and presented a sigmoidal profile most of the time [97]. By identifying $\theta(T)$ with a sigmoid, we deduced that:

$$\frac{k_1}{k_{-1}} = 2 \cdot [S] \cdot \mathrm{e}^{\alpha_D \cdot (T - T_D)} \quad 96$$

where α_D is an empirical parameter determining the steepness of the transition between single-stranded DNA and double-stranded DNA and T_D is the denaturation temperature, *i.e.*, the temperature threshold at which more than half of DS DNA is denatured. We first set T_D at 85 °C, *i.e.,* 10 °C below the temperature at which denaturation occurs in a conventional PCR.

The constant α_D was adjusted so that θ dropped to 1% at 95 °C. The calculation gave α_D = 0.46. Finally, the time constant k_1 (k_{-1}) corresponded, according to Equation (79), to the half-life constant of double-stranded (single-stranded) DNA when it was at much higher (lower) temperature than T_D. This fixed the dynamics of the denaturation reaction. We arbitrarily set the sum k_1+ k_{-1} to 0.05, which represented a characteristic time of 20 seconds.

4.3.2. Primer hybridization

We applied the same reasoning with constants k_2 and k_{-2} since the binding of the primer with single-stranded DNA can be likened to hybridization. However, as the hybridization temperatures depend on the length of the sequences, k_2 and k_{-2} differed from k_1 and k_{-1}. The ratio was also reversed because k_1 represents a denaturation rate while k_2 represents a hybridization rate. Using the same sigmoidal approximation as before, we wrote:

$$\frac{k_{-2}}{k_2 \cdot [P]} = \mathrm{e}^{\alpha_H \cdot (T - T_H)} \qquad 97$$

Where α_H plays the same role as α_D and where T_H is the primer hybridization temperature. This time, we set T_H at 70 °C, 10 °C higher than the usual temperature of this step in a conventional PCR. α_D was adjusted as before so that the hybridization efficiency decreased to 99% for a temperature 10 °C below T_H, which again gave 0.46. Finally, k_2 (k_{-2}) corresponded to the rate of formation (the half-life) of the single-stranded DNA + primer complex when the temperature was more than 10 °C below (above) T_H. These values represented the dynamics of primer hybridization that we assumed to be faster than the dynamics of denaturation because the DNA strands were shorter. We arbitrarily set the sum k_2+ k_{-2} to 2 s^{-1}, which represented a time constant of 0.5 seconds.

4.3.3. Extension

Extension can be thought of as an enzymatic reaction. Therefore, we simplified the model by using a Michaelis-Menten equation, which avoided having to use k_3, k_{-3} and k_4. The approach was the same as in [98]. In this case, equations (88) and (89) were assumed to be in a permanent steady state, and v_{cat}, which was both the substrate consumption rate – the SP complex formed by the primer + single stranded DNA – and the production rate of double-stranded DNA, was written as:

$$v_{cat} = V_{max} \frac{[SP]}{K_M + [SP]} \qquad 98$$

where K_M is the Michaelis constant, *i.e.,* the concentration of SP below which the reaction rate is divided by 2, and V_{max} the maximum reaction rate. In principle, $V_{max} = k_4 \cdot [E]$ in the Michaelis equation. However, k_4 depends on the concentration of available nucleotides. We modeled this dependence by a Hill equation:

$$k_4 = k_{max} \cdot \frac{[N]^q}{K_N^q + [N]^q} \qquad 99$$

where k_{max} is the maximum speed of the reaction when the nucleotide concentration is not limiting, K_N is the nucleotide concentration below which the rate of the reaction is divided by 2, and q is an empirical parameter illustrating the steepness of the transition from

maximum production to zero production. As we were dealing with a hybridization phenomenon once again, we assumed that this k_{max} also depended on temperature, and applied a sigmoid law model depending on the two parameters α_E and T_E:

$$k_{max}(T) = \frac{{k'}_{max}}{1 + e^{\alpha_E \cdot (T - T_E)}} \quad 100$$

The complete equation for v_{cat} was written:

$$v_{cat} = \frac{{k'}_{max}}{1 + e^{\alpha_E \cdot (T - T_E)}} \cdot [E] \cdot \frac{[N]^q}{K_N^q + [N]^q} \cdot \frac{[SP]}{K_M + [SP]} \quad 101$$

The extension temperature T_E was set 10 °C above the extension temperature in conventional PCR, *i.e.*, 85 °C. α_E was calculated in the same way as before. ${k'}_{max}$ was the maximum rate at which the polymerase extended a DNA strand. This value was set at 0.05, which corresponds to an average extension time of 20 seconds. The other three values were also fixed arbitrarily so that i) the limitation of the reaction rate due to the SP concentration was not prohibitive at the start of the PCR when the SP concentration was low, and ii) the limitation of the reaction rate due to the amount of nucleotides only occurred at the end of the PCR.

By using the Michaelis model, we reduced the dynamic model of the PCR to 5 differential equations depending on 5 variable quantities, *i.e.*, the concentrations of single-stranded DNA, double-stranded DNA, primers, nucleotides, and single-stranded DNA bound to the primers. The model parameters are summarized in Table 6.

Table 6. Summary of the different parameters used for the simulation.

$k_{1,max}$ and $k_{-1,max}$	DNA half-life constant under optimal temperature conditions	0.05 s^{-1}
$k_{2,max}$ and $k_{-2,max}$	Hybridization rate of the primer on single-stranded DNA under optimal temperature conditions	2 s^{-1}
${k'}_{max}$	Maximum polymerase extension rate	0.05 s^{-1}
K_M	SP complex concentration at which the DNA extension rate is divided by 2	1 nM
K_N	Nucleotide concentration at which the DNA extension rate is divided by 2	10 nM
q	Steepness of the transition when nucleotide deficiency limits the extension rate	2.5
α_D	Steepness of the transition of the denaturation curve	0.46
α_H	Steepness of the transition of the hybridization curve	0.46

α_E	Steepness of the transition of the elongation curve	0.46
T_D	Denaturation temperature	95 °C
T_H	Hybridization temperature	65 °C
T_E	Extension temperature	75 °C
p	Number of nucleotides in the sequence to be amplified	50

4.3.4. Thermal model

We considered two scenarios for these simulations. In the first scenario, the sample was placed in contact with an aluminum plate whose temperature varied over time to mimic the cycles. We assumed that the heat exchange between the aluminum plate and the sample was perfect, as in the thermal simulations presented in section 3. The temperature of the sample was modeled by a first-order system:

$$C_{th} \cdot \frac{dT}{dt} - h \cdot S \cdot (T - T_{ambiante}) = \phi(t) \qquad 102$$

where C_{th} is the thermal capacity of the aluminum plate, h the convection coefficient with air, S the contact area between the aluminum plate and air, and $\phi(t)$ the thermal power supplied to the plate. The aluminum plate studied in paragraph 3 had a characteristic time of 1,200 seconds. This was not suitable for our system because it would have taken almost one hour for the temperature to stabilize between each cycle. Consequently, we considered a smaller heating system with a time constant of 0.1 second. The simulated cycle consisted of 3 steps: 2 minutes at 95 °C, 2 minutes at 60 °C, and 2 minutes at 75 °C.

In a second step, we considered a system closer to the one we manufactured. The sample circulated in contact with an aluminum plate whose temperature was maintained at 95°C, then passed through the insulation layer to come into contact with the other plate at 65°C. The contact time with the hot plate was 56 seconds, the contact time with the cold plate 116 seconds, and the crossing of the insulation layer lasted 4 seconds each time. Based on the thermal simulation carried out in section 3, we considered the temperature gradient in the insulation layer as linear.

4.3.5. Simulations

The two PCR models were implemented in Python. Unless otherwise indicated, the simulations were performed with the parameters listed in Table 1. The initial concentrations were 1 µM for the polymerase, 10 µM for the primers, 1 mM for the nucleotides, and 1 pM for DNA. The simulations were all carried out over 20 cycles, with an expected amplification in the order of one million.

4.4. Case of "classical" PCR

The simulation results for classical PCR are given in Figure 83 for the ideal case. We observed the alternated denaturation and hybridization phases, with alternate predominance of single-stranded and double-stranded DNA each time. The simulated amplification after 20 cycles was 991,000, consistent with theory. By modifying the parameters of the model, we observed a

strong dependence of the PCR performance on the k_{-1} / k_2 ratio, due to the model itself. Indeed, when single-stranded DNA is cooled down to 60 °C, two reactions are in competition: i) hybridization of the primer (whose constant is k_{-1}), and ii) hybridization between two DNA strands (whose constant is k_2). To succeed in amplifying DNA, it is necessary to prevent single-stranded DNA from cross-hybridizing, and therefore to have $k_2 \gg k_{-1}$.

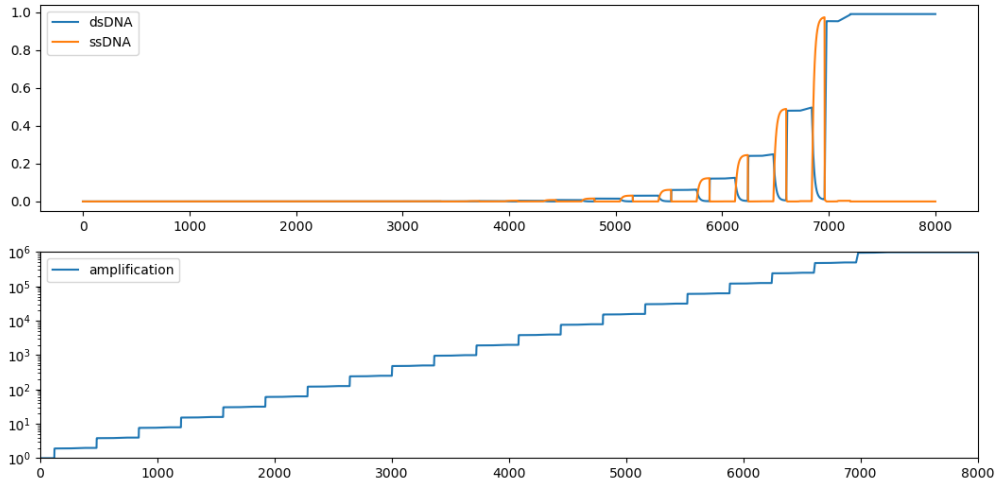


Figure 83. Normal case.

We studied the case of a limiting number of available primers (Figure 84) by decreasing the initial primer concentration to 10 nM. The simulation showed a sharp stop of the PCR when the primer stock was depleted (during the 12th cycle). Amplification decreased 200-fold. A similar effect was observed when the number of available nucleotides was limiting.

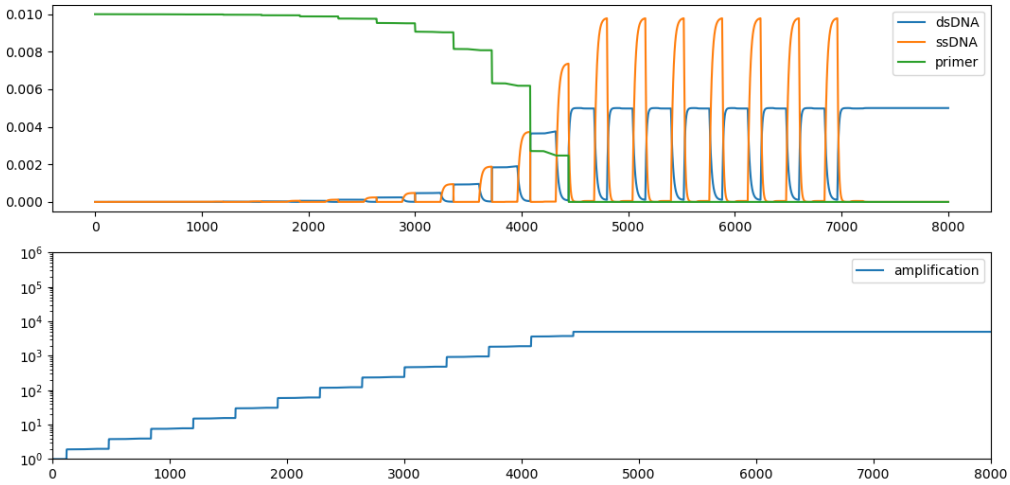


Figure 84. Simulation with a reduced number of primers.

Likewise, we simulated the behavior of the PCR when the number of polymerases was limiting. Figure 85 shows a simulation for a polymerase concentration of 10 nM. Amplification

decreased at the end of the PCR because double-stranded DNA was not formed during the extension step when the number of single-stranded DNA + primer complexes was higher than the number of polymerases. In this example, we observed a five-fold decrease of the PCR gain.

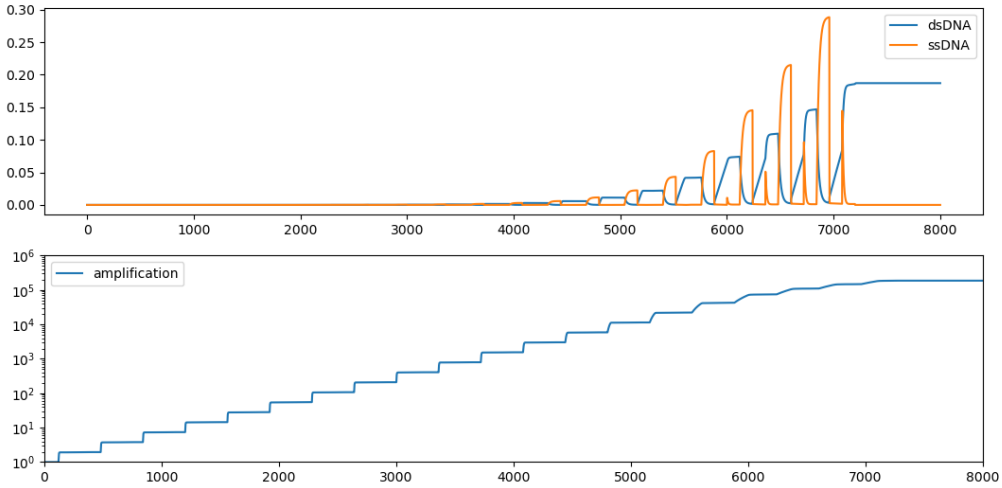


Figure 85. Simulation with a reduced number of polymerases.

We also used the model to simulate a two-step PCR (2 minutes at 95°C, then 4 minutes at 65°C). The impact was a decreased efficiency of each cycle leading to an overall two-fold gain loss. Finally, we analyzed the impact of the time constant of the heater on the PCR performance. For example, we simulated the result of a PCR with a device with time constant of 10 seconds instead of 0.1 seconds. Again, the impact was reduced PCR efficiency at each cycle, leading to a 100-fold decrease of the overall gain.

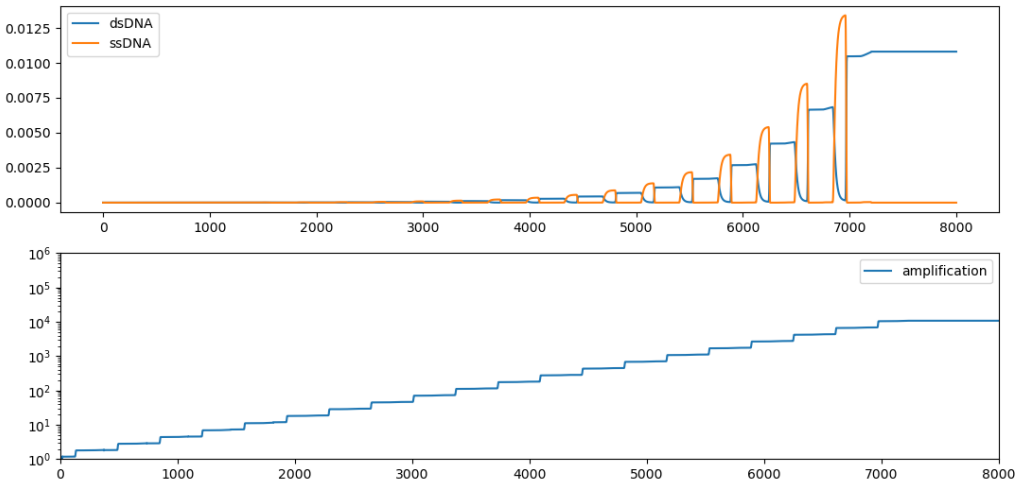


Figure 86. Simulation in optimal conditions but with a thermal device whose time constant was 10 seconds instead of 0.1 second.

4.5. Case of micro-PCR

We simulated a µPCR device under the same conditions as a conventional PCR device, with the same temperature stages as in the device we developed (Figure 87). Amplification seemed efficient, but its gain was only 66,000. Denaturation of double stranded DNA did not reach its asymptote with the chosen parameters and cycle times because the denaturation time was too short. We performed a new simulation with a denaturation time of 4 minutes instead of 1. This time the gain was very close to the one obtained using the conventional device with the two-step PCR, that is to say approximately half of the expected theoretical gain.

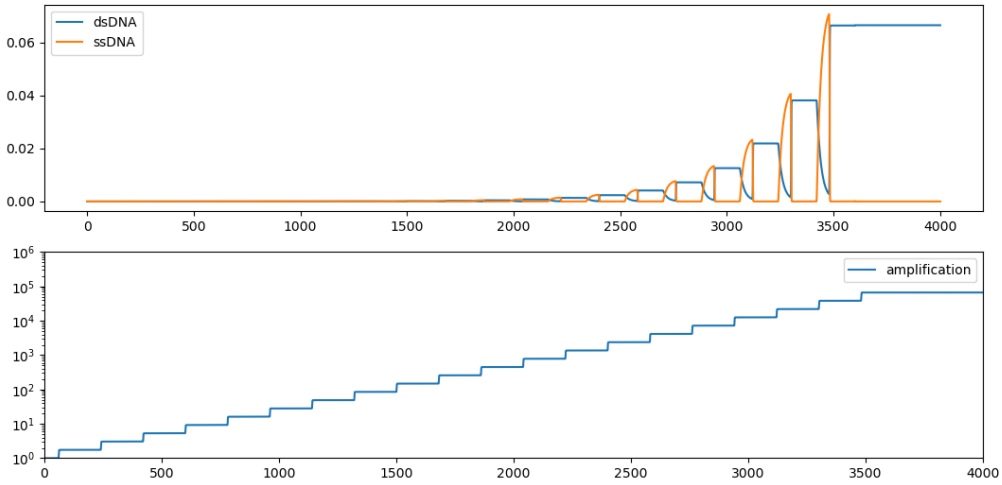


Figure 87. Simulation of the micro-PCR device.

5. Prospects

5.1. Full lab-on-a-chip modeling

The general principle of a lab-on-a-chip is to detect a specific gene in a DNA sample and allow continuous quantitative monitoring of the PCR. Our design and simulation tools will allow us to optimally design the PCR as well as the entire device.

In the diagram of the model in Figure 88, we can see a central controller for managing all the input parameters of the lab-on-a-chip, *i.e.*,:

- The concentration of the species present in the PCR premix. We can also see a fluorophore in the diagram, to monitor the PCR continuously. Its function is detailed in section 5.2.
- The concentration of the marker antibody of the gene to be detected by the lab-on-a-chip.
- The inflow rate of each species.
- The temperature of the PCR zones, as seen in 3.5.

It will also process the signals measured at the output to possibly set up feedbacks, *i.e.*,:

- Temperature monitoring with suitable sensors and adjustment of the powers sent to the heating systems (already described in paragraph 3.1)

- Monitoring of the PCR with an optical biosensor to adapt the supply of reagents or to stop when the equilibrium state of the PCR is reached (detailed in Chapter 4.2.)
- Measure the response of a biosensor aimed at detecting a gene of interest to possibly stop the reaction as soon as the detection threshold is reached and thus speed up the diagnosis.

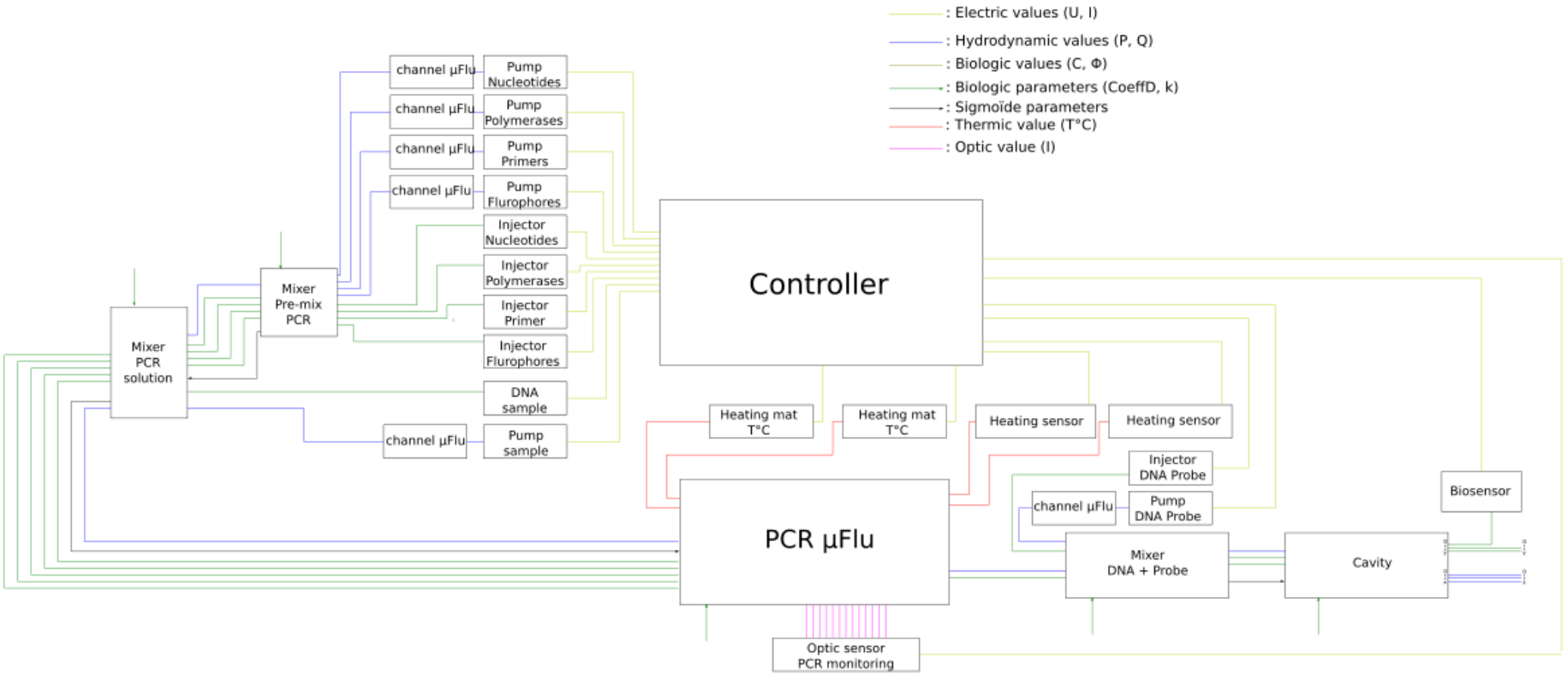


Figure 88. Block diagram of the final lab-on-a-chip. Each block can be simulated using tools developed in the previous sections.

5.2. qPCR device

Our microfluidic PCR tool could be improved by measuring DNA concentrations over the cycles. This technique is already present in biology and is called qPCR or real-time PCR [99].

The principle of real-time PCR is a standard PCR reaction carried out in the presence of a fluorophore (*e.g.*, a SYBR dye [100]) which fluoresces when it is intercalated in the DNA helix. Fluorescence increases as the amount of PCR product increases and is quantified after each completed PCR cycle [101].

Fluorescence is detected by widely used sensors such as phototransistors, photodiodes or photomultiplier tubes [100]. Optical devices can capture the fluorescent light emitted by fluorophores and measure light intensity [90]. We can imagine a detection device adapted to our system, as schematically described in Figure 89.

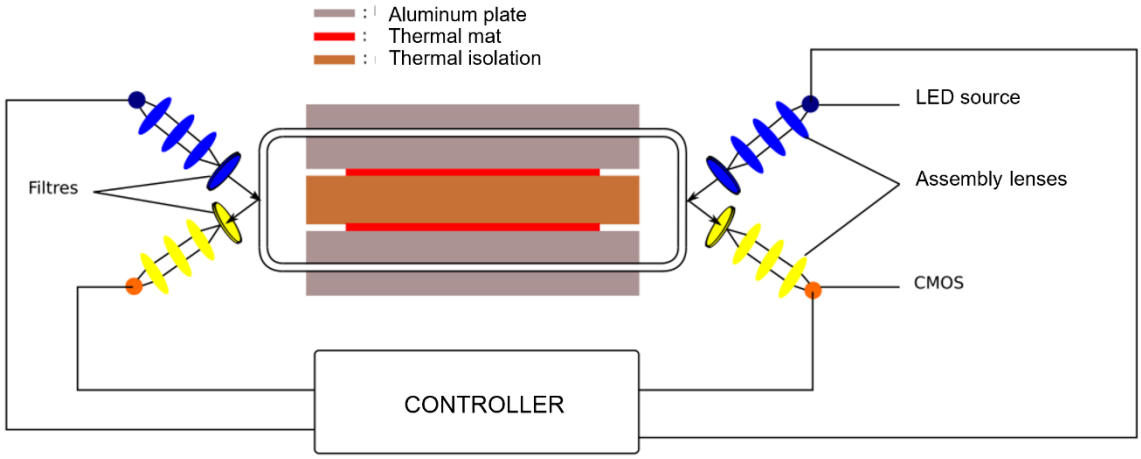


Figure 89. Schematic representation of the continuous-control device of conventional PCR as adapted to microfluidic PCR.

5.3. Gene detection

Many biosensors have been developed to detect specific genes in a DNA strand [102]. Almost all types of transducers (see Chapter 1) have their own method of gene detection.

We can imagine a mixture with a marker of the gene we want to detect, *e.g.*, a specific antibody. Then, in a microfluidic cavity, we would immobilize anti-DNA antibodies on the floor of the cavity, and these antibodies would immobilize the labeled DNA at the level of the biosensor transducer.

Modeling this analysis chamber using the tools developed in Chapter 4 allowed us to optimize its design.

5.4. Suitable mixers

5.4.1. PCR premix mixer

As seen earlier (1.2), we need a specific mixture of reagents with the DNA sample to perform a PCR. This preparation needs to be mixed properly. To design this mixer, we used the techniques seen in Chapter 2. Our mixer was converted into an equivalent electrical diagram (Figure 90).

The resistances were calculated according to the proportion of each element (nucleotides, primers, polymerases, fluorophore, and DNA sample) present in the pre-PCR preparation.

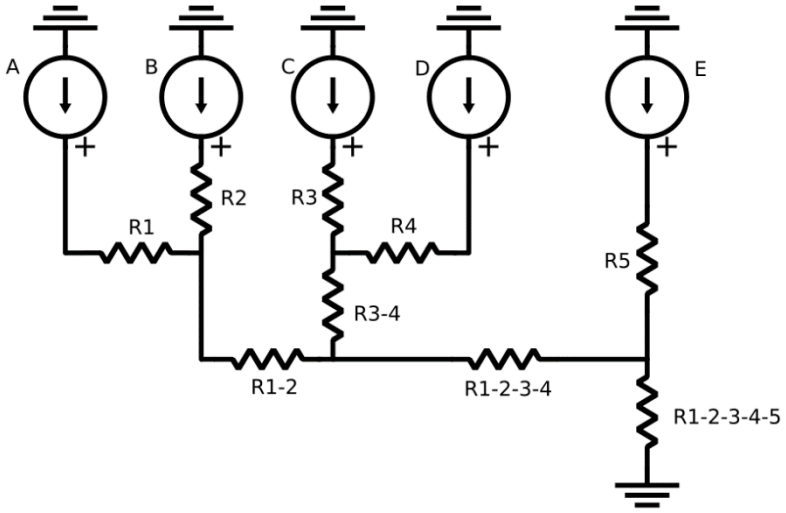


Figure 90. Equivalent electrical diagrm of a pre-PCR mixer. (A), (B), (C) and (D) Pumps dedicated to the pre-mix. (E) Pump dedicated to the DNA sample.

5.4.2. PCR output mixer

At the output of the PCR is a mixture of the amplified DNA sample + a marker of the gene to be detected.

In this case, the mixture involves a reaction between the two mixed species. To be able to model this mixer as best as possible, we need the tools developed in Chapter 3, namely the compact microfluidic mixer model.

6. Conclusion

In this chapter, we saw how to create a microfluidic PCR tool that allowed us to add temperature modeling and the modeling of a complex biological reaction.

Microfluidic PCR was tested experimentally to validate our thermal model. The tool will make it possible to create a complex lab-on-a-chip in the follow-up of the project.

We presented the modeling of a complex lab-on-a-chip using the different tools developed in all the chapters of this thesis.

General conclusions and prospects

Throughout this thesis we strengthened our belief in the future of labs on chips. We are convinced that it will depend on task specialization, with design on the one hand and manufacturing on the other hand.

In the first chapter of the manuscript, we explained the complexity of labs on chips by showing the scientific plurality that results from it. We saw that modeling tools existed, but none of them allowed for the modeling of an entire lab-on-a-chip in a single simulation environment. To solve this issue, we decided to create a design tool for labs on chips in a microelectronics simulation environment.

In the second chapter, we saw that microfluidic circuits could already be modeled through the analogy between electronics and microfluidics. This modeling is particularly powerful and already used to model many microfluidic circuits. However, it can be problematic in the event of concentration gradients within a passive mixer.

To answer this new issue, we studied passive mixing in microfluidics in the third chapter of the manuscript. The equations relating to the mixture were solved by converting them into Fourier series. Thus, we developed a compact mixer model of the concentration gradients within a passive microfluidic mixer.

In the fourth chapter, we developed a low-abstraction modeling tool to simulate certain areas of the laboratory on a more complex chip deserving a more precise simulation. Previous work by the team had showed that 3D modeling of the diffusion of a molecule in a microelectronics simulation environment was possible. By resuming this work, we added advection to adapt this simulator to a more general use of labs on chips.

Finally, in the last chapter, we developed a microfluidic PCR tool, the starting point of a more complex lab-on-a-chip. We included thermal modeling into our modeling tool. We also developed a model describing the different biological reactions of a PCR. Finally, we designed a complete model of a theoretical lab-on-a-chip using the set of models developed throughout this thesis.

We ultimately developed a complex multiphysics modeling tool allowing the modeling of labs on chips. The following improvements can be made:

- Add the biological reaction to the compact mixer model described in Chapter 3.
- Validate our finite difference simulator, presented in Chapter 4, in 3D with the refining of the meshes for more precision in certain zones.
- Test our tools in existing labs on chips to improve them with experimental comparisons.

A final improvement to our simulator would be to make it more user-friendly for external users, as it currently remains a tool that can only be used by insiders.

This work is a substantial first step towards the development of a design support tool for labs on chips. These tools are expected to be developed at the same dazzling speed as microelectronics was, and this work will be helpful for future designers.

References

[1] Y. Xia and G. M. Whitesides, 'Soft Lithography', *Angew. Chemie - Int. Ed.*, vol. 37, pp. 550–575, 1998.

[2] G. F. Christopher and S. L. Anna, 'Microfluidic methods for generating continuous droplet streams', *J. Phys. D. Appl. Phys.*, vol. 40, no. 19, pp. R319–R336, 2007.

[3] M. Abdelgawad, P. Park, and A. R. Wheeler, 'Optimization of device geometry in single-plate digital microfluidics', pp. 29–31, 2009.

[4] S. Gupta, K. Ramesh, S. Ahmed, and V. Kakkar, 'Lab-on-Chip Technology : A Review on Design Trends and Future Scope in Biomedical Applications', *Int. J. Bio-Science Bio-Technology*, vol. 8, no. 5, pp. 311–322, 2016.

[5] X. Chen and C. L. Ren, 'A microfluidic chip integrated with droplet generation, pairing, trapping, merging, mixing and releasing', *RSC Adv.*, pp. 16738–16750, 2017.

[6] L.-H. Hung, K. M. Choi, W.-Y. Tseng, Y.-C. Tan, K. J. Shea, and A. P. Lee, 'Alternating droplet generation and controlled dynamic droplet fusion in microfluidic device for CdS nanoparticle synthesis', *Lab Chip*, vol. 6, no. 2, p. 174, 2006.

[7] Y. Zhang and H. Jiang, 'A review on continuous- flow micro fluidic PCR in droplets : Advances , challenges and future', *Anal. Chim. Acta*, vol. 914, pp. 7–16, 2016.

[8] A. Weltin *et al.*, 'Cell culture monitoring for drug screening and cancer research: a transparent, microfluidic, multi-sensor microsystem', *Lab Chip*, vol. 14, no. 1, pp. 138–146, 2014.

[9] S. S. Terekhov *et al.*, 'Microfluidic droplet platform for ultrahigh-throughput single-cell screening of biodiversity', *Proc. Natl. Acad. Sci.*, vol. 114, no. 10, pp. 2550–2555, 2017.

[10] H. Pennemann, V. Hessel, and H. Löwe, 'Chemical microprocess technology — from laboratory-scale to production', vol. 59, pp. 4789–4794, 2004.

[11] P. Watts, S. J. Haswell, P. Watts, and S. J. Haswell, 'Continuous flow reactors for drug discovery', vol. 8, no. 13, pp. 586–593, 2003.

[12] G. Li and D. Lee, 'An advanced selective liquid-metal plating technique for stretchable biosensor applications', *Lab Chip*, vol. 17, no. 20, pp. 3415–3421, 2017.

[13] Z. Altintas, M. Akgun, G. Kokturk, and Y. Uludag, 'Biosensors and Bioelectronics A fully automated micro fluidic-based electrochemical sensor for real-time bacteria detection', *Biosens. Bioelectron.*, vol. 100, no. July 2017, pp. 541–548, 2018.

[14] C.-W. Huang *et al.*, 'Efficient SNP Discovery by Combining Microarray and Lab-on-a-Chip Data for Animal Breeding and Selection', *Microarrays*, vol. 4, no. 4, pp. 570–595, 2015.

[15] Y. Zhang and P. Ozdemir, 'Microfluidic DNA Amplification - a Review', vol. 44, no. 0, pp. 1–26.

[16] C. Delattre *et al.*, 'Macro to microfluidics system for biological environmental monitoring', *Biosens. Bioelectron.*, vol. 36, no. 1, pp. 230–235, 2012.

[17] A. D. Leach and J. R. Oaks, 'The Utility of Single Nucleotide Polymorphism (SNP) Data in Phylogenetics', no. November 2017, 2018.

[18] C. Berggren, P. Stalhandske, J. Brundell, and G. Johansson, 'A feasibility study of a capacitive biosensor for direct detection of DNA hybridization', *Electroanalysis*, vol. 11, pp. 156–160, 1999.

[19] H. Li, X. Liu, L. Li, X. Mu, R. Genov, and A. J. Mason, 'CMOS electrochemical instrumentation for biosensor microsystems: A review', *Sensors (Switzerland)*, vol. 17, no. 1, 2017.

[20] J. Ahn *et al.*, 'Electrical immunosensor based on a submicron-gap interdigitated electrode and gold enhancement', *Biosens. Bioelectron.*, vol. 26, no. 12, pp. 4690–4696, 2011.

[21] S. MacKay, P. Hermansen, D. Wishart, and J. Chen, 'Simulations of interdigitated electrode interactions with gold nanoparticles for impedance-based biosensing applications', *Sensors*, vol. 15, no. 9, pp. 22192–22208, 2015.

[22] A. Bonanni, I. Fernández-Cuesta, X. Borrisé, F. Pérez-Murano, S. Alegret, and M. del Valle, 'DNA hybridization detection by electrochemical impedance spectroscopy using interdigitated gold nanoelectrodes', *Microchim. Acta*, vol. 170, no. 3, pp. 275–281, 2010.

[23] K. M. M. Kabir, Y. M. Sabri, A. E. Kandjani, S. J. Ippolito, and S. K. Bhargava, 'Development and comparative investigation of Ag-sensitive layer based SAW and QCM sensors for mercury sensing applications', *Analyst*, vol. 141, no. 8, pp. 2463–2473, 2016.

[24] B. Kuswandi, Nuriman, J. Huskens, and W. Verboom, 'Optical sensing systems for microfluidic devices: A review', *Anal. Chim. Acta*, vol. 601, no. 2, pp. 141–155, 2007.

[25] A. Nakae, K. Kamon, T. Hanawa, K. Moriizumi, and S. Nakao, 'Precision Improvement in Optical Proximity Correction by Optimizing Second Illumination Source Shape', *Jpn. J. Appl. Phys.*, vol. 35, no. 12S, p. 6395, 1996.

[26] J. Léonard *et al.*, 'High-throughput time-correlated single photon counting', *Lab Chip*, vol. 14, no. 22, pp. 4338–4343, Nov. 2014.

[27] A. C. Croce, G. Bottiroli, and C. Unit, 'Autofluorescence spectroscopy and imaging : a tool for biomedical research and diagnosis Autofluorescence ':, vol. 58, 2014.

[28] N. Skivesen, A. Têtu, M. Kristensen, and D.-K. Lyngby, 'Photonic-crystal waveguide biosensor', *Opt. Soc. Am.*, vol. 15, no. 6, pp. 3169–3176, 2007.

[29] X. Fan, I. M. White, S. I. Shopova, H. Zhu, J. D. Suter, and Y. Sun, 'Sensitive optical biosensors for unlabeled targets: A review', *Anal. Chim. Acta*, vol. 620, no. 1–2, pp. 8–26, 2008.

[30] T. Lieske, W. Uhring, N. Dumas, A. I. Skilitski, J. Léonard, and D. Fey, 'Embedded Fluorescence Lifetime Determination for High-Throughput, Low-Photon-Number Applications', *J. Signal Process. Syst.*, vol. 91, no. 7, pp. 819–831, 2019.

[31] J. Campbell, 'High-Throughput Assessment of Bacterial Growth Inhibition by Optical Density Measurements', *Curr. Protoc. Chem. Biol.*, vol. 2, no. 4, pp. 195–208, 2010.

[32] H.-Y. Lin, W.-H. Tsai, Y.-C. Tsao, and B.-C. Sheu, 'Side-polished multimode fiber biosensor based on surface plasmon resonance with halogen light.', *Appl. Opt.*, vol. 46, no. 5, pp. 800–806, 2007.

[33] G. M. Whitesides, 'The origins and the future of microfluidics.', *Nature*, vol. 442, no. 7101, pp. 368–373, Jul. 2006.

[34] P. Tabeling, *Introduction to Microfluidics*. Oxford University press, 2005.

[35] C. H. Kwak *et al.*, 'Customized microfluidic reactor based on droplet formation for the synthesis of monodispersed silver nanoparticles', *J. Ind. Eng. Chem.*, vol. 63, pp. 405–410, 2018.

[36] K. Choi, A. H. C. Ng, R. Fobel, and A. R. Wheeler, 'Digital Microfluidics', 2012.

[37] M. Yamada and M. Seki, 'Hydrodynamic filtration for on-chip particle concentration and classification utilizing microfluidics', *Lab Chip*, vol. 5, no. 11, p. 1233, 2005.

[38] G. Persichetti, I. A. Grimaldi, G. Testa, and R. Bernini, 'Multifunctional optofluidic lab-on-chip platform for Raman and fluorescence spectroscopic microfluidic analysis', *Lab Chip*, vol. 17, no. 15, pp. 2631–2639, 2017.

[39] C. A. E. Little, N. D. Orloff, I. E. Hanemann, C. J. Long, V. M. Bright, and J. C. Booth, 'Modeling electrical double-layer effects for microfluidic impedance spectroscopy from 100 kHz to 110 GHz', *Lab Chip*, vol. 17, no. 15, pp. 2674–2681, 2017.

[40] S. Colin, *Microfluidique*. Hermes Science Publications, 2004.

[41] P. Kim, K. W. Kwon, M. C. Park, S. H. Lee, and S. M. Kim, 'Soft Lithography for Microfluidics : a Review', *Biochip J.*, vol. 2, no. 1, pp. 1–11, 2008.

[42] 'Elveflow(R)'. [Online]. Available: https://www.elveflow.com/microfluidic-reviews/soft-lithography-microfabrication/microfluidic-3d-printer/. [Accessed: 20-Aug-2020].

[43] V. Ananthanarayanan and W. Thies, 'Biocoder : A programming language for standardizing and automating biology protocols', *J. Biol. Eng.*, vol. 4, no. 1, p. 13, 2010.

[44] 'COMSOL Multiphysics® v. 5.4.'

[45] D. Das, K. Biswas, and S. Das, 'A microfluidic device for continuous manipulation of biological cells using dielectrophoresis', *Med. Eng. Phys.*, vol. 36, no. 6, pp. 726–731, 2014.

[46] P. Voigt, G. Schrag, and G. Wachutka, 'Microfluidic system modeling using VHDL-AMS and circuit simulation', vol. 29, pp. 791–797, 1998.

[47] K. W. Oh, K. Lee, B. Ahn, and E. P. Furlani, 'Design of pressure-driven microfluidic networks using electric circuit analogy', *Lab Chip*, vol. 12, no. 3, pp. 515–545, 2012.

[48] E. E. Tsur, 'Computer-Aided Design of Microfluidic Circuits', pp. 285–307, 2020.

[49] C. Dominguez, R. Boelens, and A. M. J. J. Bonvin, 'HADDOCK: A Protein−Protein Docking Approach Based on Biochemical or Biophysical Information', *J. Am. Chem. Soc.*, vol. 125, no. 7, pp. 1731–1737, Feb. 2003.

[50] G. C. P. van Zundert *et al.*, 'The HADDOCK2.2 Web Server: User-Friendly Integrative Modeling of Biomolecular Complexes', *J. Mol. Biol.*, vol. 428, no. 4, pp. 720–725, 2016.

[51] D. Chandrasekaran, M. Kovermann, M. Gopalswamy, and A. Hellmuth, 'Solution structure of the PsIAA4 oligomerization domain reveals interaction modes for transcription factors in early auxin response', no. April, 2015.

[52] Y. Liu, M. Li, N. K. Alham, and S. Hammoud, 'HSim: A MapReduce simulator in enabling Cloud Computing', *Futur. Gener. Comput. Syst.*, vol. 29, no. 1, pp. 300–308, 2013.

[53] L. Michaelis, M. L. Menten, K. A. Johnson, and R. S. Goody, 'The original Michaelis constant: translation of the 1913 Michaelis-Menten paper', *Biochemistry*, vol. 50, no. 39, pp. 8264–8269, Oct. 2011.

[54] C. V Rao, 'Stochastic chemical kinetics and the quasi-steady-state assumption : Application to the Gillespie algorithm Stochastic chemical kinetics and the quasi-steady-state assumption : Application to the Gillespie algorithm', no. August 2002, 2014.

[55] A. Enciso, M. D. Johnston, D. F. Anderson, and D. F. Anderson, 'Stochastic analysis of biochemical reaction networks with absolute concentration robustness', 2014.

[56] T. E. Turner, S. Schnell, and K. Burrage, 'Stochastic approaches for modelling in vivo reactions', vol. 28, pp. 165–178, 2004.

[57] M. Madec, C. Lallement, and J. Haiech, 'Modeling and simulation of biological systems using SPICE language', *PLoS One*, no. Ic, pp. 1–21, 2017.

[58] S. Martinoia, G. Massobrio, and L. Lorenzelli, 'Modeling ISFET microsensor and ISFET-based microsystems : a review', vol. 105, pp. 14–27, 2005.

[59] Q. Liu *et al.*, 'Biosensors and Bioelectronics Impedance sensing and molecular modeling of an olfactory biosensor based on chemosensory proteins of honeybee', *Biosens. Bioelectron.*, vol. 40, no. 1, pp. 174–179, 2013.

[60] A. C. Ward, P. Connolly, and N. P. Tucker, 'Pseudomonas aeruginosa can be detected in a polymicrobial competition model using impedance spectroscopy with a novel biosensor', vol. 9, no. 3, 2014.

[61] A. Manickam, C. A. Johnson, S. Kavusi, and A. Hassibi, 'Interface Design for CMOS-Integrated Electrochemical Impedance Spectroscopy (EIS) Biosensors', pp. 14467–14488, 2012.

[62] F. Pêcheux, C. Lallement, and A. Vachoux, 'VHDL-AMS and Verilog-AMS as Alternative Hardware Description Languages for Efficient Modeling of Multi-Discipline Systems', 2002.

[63] F. Pecheux, B. Allard, C. Lallement, A. Vachoux, and H. Morel, 'Modeling and Simulation of Multi-Discipline Systems Using Bond Graphs and VHDL-AMS'.

[64] E. Rosati, 'Outils d'aide à la conception pour l'ingénierie de systèmes biologiques', UNIVERSITÉ DE STRASBOURG, 2018.

[65] P.-G. De Gennes, F. Brochard-Wyart, and D. Quéré, *Capillarity and wetting phenomena: drops, bubbles, pearls, waves*. 2004.

[66] J. Koo and C. Kleinstreuer, ‘Liquid flow in microchannels: experimental observations and computational analyses of microfluidics effects’, *J. Micromechanics Microengineering*, vol. 13, no. 5, pp. 568–579, 2003.

[67] D. Erickson, ‘Towards numerical prototyping of labs-on-chip: Modeling for integrated microfluidic devices’, *Microfluid. Nanofluidics*, vol. 1, no. 4, pp. 301–318, 2005.

[68] D. Javier, F. Wandosell, and J. Avila, ‘Glycosaminoglycans and B-amyloid, prion and tau peptides in neurodegenerative diseases’, vol. 23, pp. 1323–1332, 2002.

[69] C. Lee, W. Wang, C. Liu, and L. Fu, ‘Passive mixers in microfluidic systems : A review’, *Chem. Eng. J.*, vol. 288, pp. 146–160, 2016.

[70] Y. K. Suh and S. Kang, ‘A review on mixing in microfluidics’, *Micromachines*, vol. 1, no. 3, pp. 82–111, 2010.

[71] F. Su and J. Zeng, ‘Computer-Aided Design and Test for Digital Microfluidics’, pp. 60–70, 2007.

[72] E. E. Tsur and A. Shamir, ‘Computer-Aided Design Computer-aided design of resistance micro-fluidic circuits for 3D’, *Comput. Des.*, vol. 98, pp. 12–23, 2018.

[73] Y. Wang, Q. Lin, and T. Mukherjee, ‘Composable Behavioral Models and Schematic-Based Simulation of Electrokinetic Lab-on-a-Chip Systems’, *IEEE T Comput AID D*, vol. 25, no. 2, pp. 258–273, 2006.

[74] N. S. Lynn Jr *et al.*, ‘Biosensing enhancement using passive mixing structures for microarray-based sensors’, *Biosens. Bioelectron.*, 2013.

[75] Pr. Greenside, ‘Fourier analysis of a 1d diffusion equation’, in *PBOC2 textbook*, no. Section 13.2.3, 2018, pp. 525–529.

[76] M. Madec, A. Bonament, E. Rosati, L. Hebrard, and C. Lallement, ‘Virtual prototyping of biosensors involving reaction- diffusion phenomena’, in *2018 16th IEEE International New Circuits and Systems Conference (NEWCAS)*, 2018, pp. 3–6.

[77] H. Vogt, M. Hendrix, P. Nenzi, and D. Warning, ‘Ngspice User’s Manual Version 34’, vol. 34, 2021.

[78] S. Circuit and S. Reference, ‘Spectre ® Circuit Simulator Reference’, no. January, 2020.

[79] M. Madec, L. Hébrard, J. Kammerer, A. Bonament, E. Rosati, and C. Lallement, ‘Multiphysics Simulation of Biosensors Involving 3D Biological Reaction–Diffusion Phenomena in a Standard Circuit EDA Environment’, *IEEE T CIRCUITS-I*, vol. 66, no. 6, pp. 2188–2197, 2019.

[80] E. M. Lewandowski, K. G. Lethbridge, R. Sanishvili, J. Skiba, K. Kowalski, and Y. Chen, ‘Mechanisms of proton relay and product release by Class A β-lactamase at ultrahigh resolution’, *FEBS J.*, vol. 285, no. 1, pp. 87–100, 2018.

[81] J. Liu, L. Wu, X. Fang, and E. H. Etuk, ‘Using Python to Solve the Navier-Stokes Equations -Applications in the Preconditioned Iterative Methods’, *J. Sci. Res. Reports*, vol. 7, no. 73, pp. 207–217, 2015.

[82] C. Prud’homme, ‘Feelpp’. [Online]. Available: http://www.feelpp.org/.

[83] D. Brune and S. Kim, 'Predicting protein diffusion coefficients', *Biophysics (Oxf).*, vol. 90, no. May, pp. 3835–3839, 1993.

[84] C. Zhang, J. Xu, W. Ma, and W. Zheng, 'PCR microfluidic devices for DNA amplification', vol. 24, pp. 243–284, 2006.

[85] M. Park, J. Won, B. Y. Choi, and C. J. Lee, 'Optimization of primer sets and detection protocols for SARS-CoV-2 of coronavirus disease 2019 (COVID-19) using PCR and real-time PCR', *Exp. Mol. Med.*, pp. 963–977, 2020.

[86] T. Dauxois and M. Peyrard, 'Dynamics and thermodynamics of a nonlinear model for DNA denaturation', *Phys. Rev. Lett.*, vol. 47, 1993.

[87] H. Christensen, Ø. Angen, R. Mutters, J. E. Olsen, and M. Bisgaard, 'DNA – DNA hybridization determined in micro-wells using covalent attachment of DNA', *Int. J. Syst. Evol. Microbiol.*, vol. 50, pp. 1095–1102, 2000.

[88] L. A. Dodson and J. A. Kant, 'Two-temperature PCR and heteroduplex detection: application to rapid cystic fibrosis screening.', *Mol. Cell. Probes*, vol. 5, no. 1, pp. 21–25, Feb. 1991.

[89] J. M. Haber, P. R. C. Gascoyne, and K. Sokolov, 'Rapid real-time recirculating PCR using localized surface plasmon resonance (LSPR) and piezo-electric pumping', *Lab Chip*, vol. 17, pp. 2821–2830, 2017.

[90] Y. Jiang, A. Manz, and W. Wu, 'Fully automatic integrated continuous- flow digital PCR device for absolute DNA quanti fication', *Anal. Chim. Acta*, vol. 1125, pp. 50–56, 2020.

[91] C. Rubber, P. Co, and D. Report, *TABLE OF EMISSIVITY OF VARIOUS SUFACES*. .

[92] M. S. Shell, *Thermodynamics and Statistical Mechanics: An Integrated Approach*. Cambridge University Press, 2015.

[93] W. M. Rohsenow and J. R. Hartnett, *HANDBOOK OF HEAT TRANSFER*. .

[94] 'RDM Simulator'. [Online]. Available: http://www.calculrdm.com/.

[95] D. Rickwood and B. D. Hames, *Gel Electrophoresis of Nucleic Acids: A Practical Approach*. IRL Press, 1982.

[96] S. Mehra and W. Hu, 'A Kinetic Model of Quantitative Real-Time Polymerase Chain Reaction', no. 612, 2005.

[97] N. Crews, C. Wittwer, and B. Gale, 'Product differentiation during continuous-flow thermal gradient PCR', pp. 919–924, 2008.

[98] J. L. Gevertz, S. M. Dunn, and C. M. Roth, 'Mathematical Model of Real-Time PCR Kinetics', pp. 1–10, 2005.

[99] F. Postollec, H. Falentin, S. Pavan, J. Combrisson, and D. Sohier, 'Recent advances in quantitative PCR (qPCR) applications in food microbiology', *Food Microbiol.*, 2011.

[100] M. H. Josefsen, C. Löfström, T. Hansen, E. Reynisson, and J. Hoorfar, 'Instrumentation and Fluorescent Chemistries Used in Quantitative Polymerase Chain Reaction', *Polym. Chain React. Theory Technol.*, pp. 79–104, 2019.

[101] D. Rodríguez-Lázaro and M. Hernández, 'Introduction to the Real-time Polymerase Chain Reaction', *Polym. Chain React. Theory Technol.*, pp. 1–18, 2019.

[102] A. Sassolas *et al.*, 'DNA Biosensors and Microarrays', 2008.

[103] V. Tsouti, C. Boutopoulos, I. Zergioti, and S. Chatzandroulis, 'Capacitive microsystems for biological sensing', *Biosens. Bioelectron.*, vol. 27, no. 1, pp. 1–11, 2011.

Alexi BONAMENT

Outils d'aide à la conception pour les laboratoires sur puce

Résumé

Depuis les années 2000 la demande en outils d'analyses biologiques et chimiques, fiables et portatifs a augmenté dans plusieurs domaines (santé, environnement, agroalimentaire, …). Pour répondre à cette demande, des outils d'analyses miniaturisés, appelés « laboratoires sur puces » se sont développés en particulier sur le plan technique au cours de la dernière décennie. Cependant, les développements industriels à grande échelle des laboratoires sur puces restent encore à découvrir et à être exploités. Cette thèse essaie de donner une réponse au développement industriel des laboratoires sur puce. Notre postulat de base est un mimétique du développement de la microélectronique. En effet, celui-ci a été fulgurant ces 50 dernières années grâce à la séparation des compétences de technologie de fabrication et de celles de modélisation. Cette séparation a permis la libre création d'industries spécialisées soit dans la fabrication, soit dans la modélisation sans interdépendance l'une avec l'autre.

Le but de cette thèse a été de développer différents outils de simulation pour les laboratoires sur puces compatibles avec un environnement de simulation électronique. Nous avons pu développer différentes approches à des niveaux d'abstractions différents afin de permettre une plus grande liberté de réponse à la simulation d'un laboratoire sur puce.

Mots-clefs : Laboratoire sur puce, simulation, modélisation, biocapteur, microfluidique, conception assistée par ordinateur

Abstract

Since the 2000s, the demand for reliable and portable biological and chemical analysis tools has increased in several areas (health, environment, agrifood, etc.). To meet this demand, miniaturized analysis tools, written "lab-on-a-chip" have been developed, in particular on a technical level, over the past decade. However, large-scale industrial developments in lab-on-chips have yet to be discovered and exploited. This thesis tries to give an answer to the industrial development of labs-on-chip. Our basic postulate is a mimetic of the development of microelectronics. Indeed, it has been dazzling over the past 50 years thanks to the separation of the skills of manufacturing technology and those of modeling. This separation allowed the free creation of industries specializing in either manufacturing or modeling without interdependence with each other.

The aim of this thesis was to develop different simulation tools for labs on chips compatible with an electronic simulation environment. We have been able to develop different approaches at different levels of abstractions in order to allow greater freedom of response to the simulation of a laboratory on a chip.

Keywords: Lab-on-chip, simulation, modeling, biosensor, microfluidics, computer-aided design