Stochastic Computing Using Droplet-Based Microfluidics

Werner Haselmayr*, Andreas Grimmer‡, Robert Wille‡

*Institute for Communications Engineering and RF-Systems
‡Institute for Integrated Circuits
Johannes Kepler University Linz, Austria
{werner.haselmayr, andreas.grimmer, robert.wille}@jku.at

Abstract. In this work, we consider the realization of stochastic computing in the microfluidic domain. To this end, we exploit the fact that both, the bit streams and the operations required for stochastic computing can be realized in microfluidic systems through droplet streams and microfluidic gates. Simulating the trajectory of the individual droplets through the microfluidic gates confirmed the validity of our approach.

Keywords: Droplet-based microfluidics, microfluidic computing, microfluidic gates, stochastic computing

1 Introduction

Droplet-based microfluidic systems refer to systems, where tiny volumes of fluids, so-called *droplets*, flow in channels of micrometer scale [1]. Currently, such systems are frequently used as platform for the realization of *Labs-on-Chip* (LoC) devices, where droplets contain biological/chemical samples and undergo several processes to execute certain laboratory experiments, e.g., DNA sequencing, cell analysis or drug discovery [1, 2]. But beyond that, droplet-based microfluidics recently also found interest in domains such as information transmission and simple computing.

For example, a simple droplet-based communication system was proposed for the first time in [3]. This idea was later extended in [4] by introducing different methodologies for information encoding using droplets, e.g., the presence/absence of droplets or the distance between two consecutive droplets. With respect to using microfluidic systems for computing, initial work has been conducted in [5, 6]. Here, the presence/absence of a droplet is used to represented Boolean values, while their flow through a dedicated microfluidic system realizes the desired Boolean functions.

In this work, we further extend these concepts in order to realize *stochastic computing* (SC, [7]) in the microfluidic domain. In SC, real numbers between 0 and 1 are represented by a stochastic bit stream which allows to realize usually complex arithmetic operations (such as a multiplication) through very simple

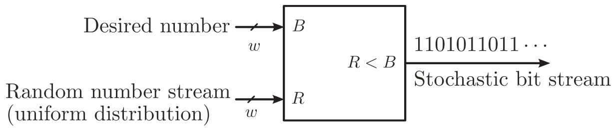


Fig. 1. Stochastic bit stream generation.

logic gates (such as an AND gate) – yielding substantially more compact realizations of circuits. The fact that both concepts (the bit streams and the operations) can be realized in microfluidic systems (namely as droplet streams and microfluidic gates [5,6], respectively) motivates a more detailed consideration of SC using droplet-based microfluidic systems.

To this end, the remainder of this work provides the following contributions: First, we give a brief introduction to SC and review the principle of droplet-based microfludics as well as microfluidic gates in Sec. 2 and Sec. 3, respectively. Afterwards, we propose a concept for realizing stochastic arithmetic operations in droplet-based microfluidic systems by adopting the SC approach for the microfludic domain in Sec. 4 – including a validation of the proposed concepts through simulations based on the duality between microfluidic systems and time-varying electrical circuits. Finally, Sec. 5 concludes the paper and briefly discusses future work.

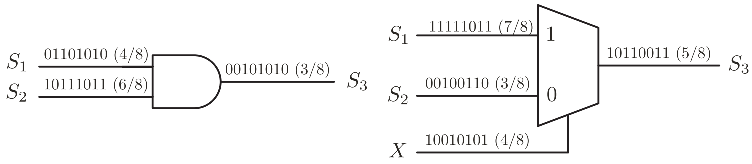
2 Stochastic Computing

In stochastic computing [7], a real number s in the unit range ($s \in [0,1]$) is represented as a serial stochastic bit stream S. The desired number corresponds to the ratio of 1's included in the bit stream to the bit stream length, i.e., the probability for each bit in the stream to be 1 is given by $\Pr(S=1)=s$. For example, the value $s=7/10$ can be represented by a stochastic stream $S \mathrel{\widehat{=}} 1101011011$ with $\Pr(S=1)=7/10$. It is important to note that the positions of the 1's in the stream is not prescribed and, thus, many different streams for the same value exists.

The generation of a stochastic bit stream can be accomplished through a comparator as shown in Fig. 1. The comparator compares a w-bit random natural number R and the value $B=[s \times 2^w]$, with $[x]$ as the nearest integer function. The number R is drawn from an uniform distribution and the number B corresponds to the mapping of the desired real number s to an w-bit natural number. If $R<B$ the output of the comparator is 1 and, thus, the probability of a 1 appearing at the output of the comparator is given by $\Pr(R<B)=\Pr(S=1)=[s \times 2^w]/2^w \approx s$.

The main benefit of SC is that arithmetic operations can be performed with simple logic circuits. In the following, we briefly discuss the realization of multiplication and addition[1].

[1] We refer to [8] for a description of the implementation for division and subtraction.



(a) Multiplication (AND gate) (b) Addition (Multiplexer)

Fig. 2. Logic circuits for performing arithmetic operations in SC [7].

– **Multiplication:** A multiplication can be implemented by an AND gate as shown in Fig. 2(a). Let us consider two independent stochastic bit streams S_1 and S_2 at the input of an AND gate. According to the AND gate behavior the output stream S_3 is only 1, if both input streams are equal to 1. More formally this can be written as

$$s_3 = \Pr(S_3 = 1) = \Pr(S_1 = 1 \wedge S_2 = 1) = \Pr(S_1 = 1)\Pr(S_2 = 1) = s_1 s_2. \quad (1)$$

Thus, the AND gate can be used to compute the product of s_1 and s_2, which are represented by the input stochastic streams S_1 and S_2, respectively.

– **Addition:** A scaled addition can be implemented using a multiplexer as shown in Fig. 2(b). Let us consider two independent stochastic streams S_1 and S_2 at the input of the multiplexer and a stochastic stream X that selects the input to be forwarded to the output. If $X = 1$ or $X = 0$ the actual bit value of the output stream S_3 corresponds to the bit value of S_1 or S_2, respectively. More formally this can be written as

$$\begin{aligned} s_3 = \Pr(S_3 = 1) &= \Pr(X = 1)\Pr(S_1 = 1) + \Pr(X = 0)\Pr(S_2 = 1) \\ &= \Pr(X = 1)\Pr(S_1 = 1) + (1 - \Pr(X = 1))\Pr(S_2 = 1) \\ &= xs_1 + (1 - x)s_2. \end{aligned} \quad (2)$$

Thus, a multiplexer can be used to compute the scaled addition of s_1 and s_2, where s_1 and s_2 are weighted by x and $1 - x$.

In addition to the benefit of realizing arithmetic operations using simple logic gates, SC has an inherent fault tolerance and requires no synchronization among the streams. This is because, in a stochastic bit stream, the information on the number to be represented is included in the stream properties (number of 1's) and not in the individual bits. Thus, all bits have a similar weight and no higher order bits exists as in the conventional binary format. For example, a bit flip changes a stochastic stream from 000101010 to 100101010 which changes the corresponding number from 3/8 to 4/8 [7]. Thus, only a small error occurs due to the bit flip. In contrast, considering the number 3/8 in conventional binary format 0.011, a bit flip of a higher order bit causes a huge error, i.e., if 0.011 changes to 0.111 the numbers change from 3/8 to 7/8.

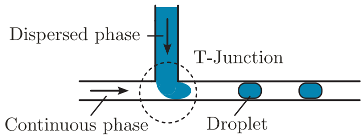


Fig. 3. Generation of droplets using a T-junction.

One major drawback of SC is that, for high accuracy, long stochastic bit streams are required – resulting in a high latency for the computation. For example, to increase the accuracy from 4 to 8 bits requires an increase of the stream length from 16 to 256 bits [7]. In a recent work, it is proposed to generate the stochastic streams deterministically rather than randomly (cf. Fig. 1), which leads to significantly lower latency while keeping the advantage of inherent fault tolerance [9].

Nevertheless, despite this drawback and motivated by the huge benefits discussed above, SC has been successfully used in various applications – including neural networks [8], control systems [10], image processing [11] and decoding of error-correcting codes [12, 13].

3 Droplet-Based Microfluidics and Microfluidic Gates

In droplet-based microfludic systems [1], tiny volumes of fluids, so-called droplets, flow in closed microchannels, triggered by some external sources (e.g., pressure or syringe pump). Typically, the droplets are generated using a T-junction, where a fluid (dispersed phase) in form of droplets is dispersed into another immiscible fluid (continuous phase) acting as carrier fluid (cf. Fig. 3).

Originally, droplet-based microfluidic systems were used as a platform for the realization of LoCs, which execute certain laboratory experiments by including biological/chemical samples in the droplet [2]. But in the past few years, also several approaches for employing droplet-based microfluidics for information transmission or computing (microfludic gates) were proposed [3–6]. In this section, we briefly describe the principle of microfluidic gates [5, 6] which will provide the basis for the proposed realization of SC in the microfluidic domain.

The working principle of microfluidic gates is based on two observations: First, droplets arriving at a junction flow along the channel with the lowest hydrodynamic resistance. Second, droplets increase the hydrodynamic resistance in a channel. For a rectangular channel with length L_c, width w_c and height h_c, the hydrodynamic resistance is given by [14]

$$R_c = \frac{\alpha \mu_c L_c}{w_c h_c^3}, \tag{3}$$

where μ_c denotes the dynamic viscosity of the carrier fluid and the dimensionless parameter α is given by $\alpha = 12[1 - 192h/(\pi^5 w_c)\tanh(\pi w_c/(2h_c))]^{-1}$.

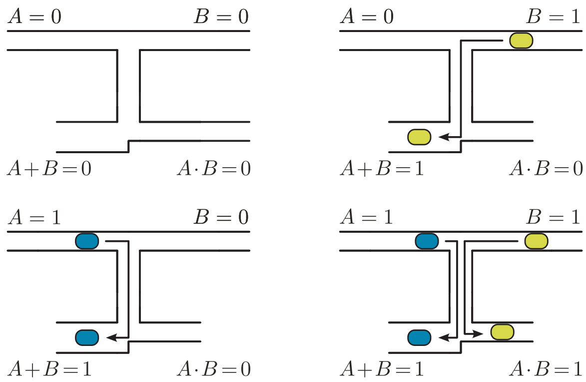


Fig. 4. Working principle of a microfluidic AND/OR gate [6].

The increase of the resistance due to a droplet in a channel is given by [15]

$$R_\mathrm{d} = \frac{(\mu_\mathrm{d} - \mu_\mathrm{c})\alpha L_\mathrm{d}}{w_\mathrm{d} h_\mathrm{d}^3}, \tag{4}$$

with L_d, w_d and h_d being the length, width and height of the droplet, respectively. Moreover, μ_d denotes the dynamic viscosity of the dispersed phase. Thus, the total hydrodynamic resistance of a channel which includes a droplet is given by $R = R_\mathrm{c} + R_\mathrm{d}$.

For the first time, microfluidic AND/OR and AND/NOT gates were proposed in [6] and are shown in Figs. 4 and 5, respectively. The channel dimensions of the AND/OR gate are chosen such that a droplet from input A or B arriving at the bottom junction flows into the OR branch ($A + B$) due the lower hydrodynamic resistance (wider channel). However, if a droplet arrives at the junction and the OR branch is occupied by a previously sent droplet (increasing the resistance of the OR branch to be higher than the AND branch), it is directed to the AND branch. This behavior corresponds to an AND and OR gate behavior[2]. The channel dimensions of the NOT gate are set such that, if there is no droplet from input A, the droplet from input B flows into the branch $\bar{A} \cdot B$. By introducing a droplet into the input channel A, the flow towards the lower channel is reduced and the droplet from input B flows towards $A \cdot B$. Thus, by providing a droplet train for input B, the gate shown in Fig. 5 realizes a NOT gate.

4 Stochastic Computing Using Microfluidic Systems

Using microfluidic systems as reviewed in the previous section allows to realize SC. To this end, a serial stochastic bit stream as well as corresponding microfluidic gates need to be realized. The latter is already available as discussed

[2] To allow for a correct functionality for $A = 1$ and $B = 1$, the droplets must enter the gate with a slight time delay.

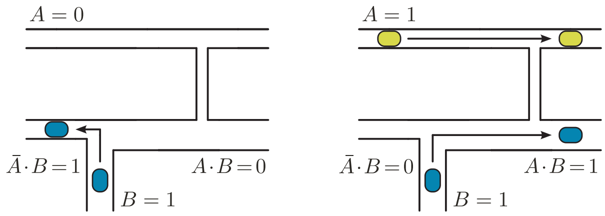


Fig. 5. Working principle of a microfluidic AND/NOT gate [6].

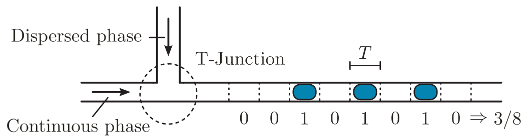


Fig. 6. Droplet stream representing the number 3/8.

before by means of Figs. 4 and 5. The stochastic bit stream can be realized by a droplet stream that can be generated by a T-junction as shown in Fig. 3. More precisely, the droplet transmission is divided into *time intervals* of duration T – generating a single or no droplet depending on whether a bit 1 or 0 occurs in the stochastic stream (cf. Fig. 6).

In the following, we validate the realization of SC in the microfludic domain through the example of a multiplication of two numbers. To this end, we evaluate the trajectory of individual droplets through a microfluidic system using the event-based simulator proposed in [15]. This simulator models microfluidic systems as time-varying electrical circuits.

We implemented an AND/OR gate as shown in Fig. 7 using the channel dimensions and fluid properties as specified in Tab. 1. We convert the numbers to be multiplied into two serial stochastic bit streams, which are used for the droplet generation. One bit stream represents the droplets which are injected into input A and the other represents the droplets which are injected into input B of the AND/OR gate. Depending on the value of the bits in the streams, a droplet is injected or not (presence/absence of a droplet for bit 1/0). More precisely, in case that both bit streams have length N, every time nT ($1 \leq n \leq N$), a bit of both streams is injected in form of droplets. It is important to note that, in order to prevent droplets merging, the injection time of input A and B are slightly time-delayed.

Using these streams as input to the AND/OR gate, we can observe whether a droplet enters the OR or the AND branch. Note that the droplet stream exiting the AND branch represents the results of the multiplication.

As discussed in Sec. 2, a longer stochastic bit stream results in a higher accuracy. Thus, given a fixed amount of time, it is desirable to have a small time interval T. However, in order to ensure a correct behavior of the AND/OR gate, T cannot be arbitrarily small. A correct behavior can only be ensured, when

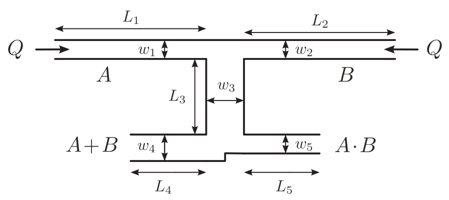


Fig. 7. Layout of AND/OR gate.

Flow rate Q	$0.25\,\mu$l/s
Dyn. viscosity cont. phase μ_c	1.002 mPa·s
Dyn. viscosity disp. phase μ_d	5.511 mPa·s
Channel height h	$70\,\mu$m
Channel lengths L_1, L_2	$125\,\mu$m
Channel lengths L_3, L_4, L_5	$62.5\,\mu$m
Channel widths w_1, w_2	$25\,\mu$m
Channel width w_3	$50\,\mu$m
Channel width w_4	$30\,\mu$m
Channel width w_5	$20\,\mu$m

Table 1. Simulation parameters.

droplets injected at time nT do not influence droplets injected at time $(n+1)T$ (e.g., by changing the resistances). In our simulations, we decreased the time interval T until droplets flow into wrong branches at the bottom junction and, thus, produced a wrong result. Considering the system specified in Fig. 7 and Tab. 1, the smallest droplet injection time interval which still guarantees a correct behavior has been determined as being $T = 2.9$ ms.

As an example, a video that shows the simulation result for the multiplication of 0.7×0.9 is available at `www.jku.at/iic/eda/sc`. This video confirms the working principle of the AND/OR gate (cf. Sec. 3). More precisely, it shows the injection of two bit streams of length $N = 10$. The droplets injected into input A have a probability $\Pr(A = 1) = 7/10$ and droplets injected into input B have a probability $\Pr(B = 1) = 9/10$. The result of the multiplication $\Pr(A=1)\Pr(B=1)$ is obtained by counting the number of droplets flowing into the AND branch (i.e. to the right bottom channel) and dividing this number by the stream length N. It is important to note that the result of the multiplication is 6/10, since we have only used stochastic bit streams of length $N = 10$. In order to improve the accuracy, the bit stream length must be increased as discussed in Sec. 2. We successfully conducted similar simulations using other inputs as well as other operations – confirming the validity of the proposed approach.

5 Conclusions

In this work, we proposed the realization of stochastic computing using droplet-based microfluidic systems. To this end, we represented stochastic bit streams as droplet streams (presence/absence of droplets for bit 1/0) and utilized existing realization of microfluidic gates to realize stochastic operations (e.g., an AND gate for a multiplication). We confirmed the validity of the proposed approach through evaluating the trajectory of the droplets in a microfluidic system. As future work, we will investigate the effect of imperfect droplet generation (e.g., different droplet volumes and injection time variation) and the implementation of more complex operations. Furthermore, we will consider ways to conduct corresponding simulations on discrete models such as proposed in [16].

References

1. D. Mark, S. Haeberle, G. Roth, F. von Stetten, and R. Zengerle, "Microfluidic lab-on-a-chip platforms: requirements, characteristics and applications," *Chemical Society Reviews*, vol. 39, no. 3, pp. 1153–1182, 2010.
2. S.-Y. Teh, R. Lin, L.-H. Hung, and A. P. Lee, "Droplet microfluidics," *Lab on a Chip*, vol. 8, pp. 198–220, 2008.
3. M. J. Fuerstman, P. Garstecki, and G. M. Whiteside, "Coding/decoding and reversibility of droplet trains in microfluidic networks," *Science*, vol. 315, no. 5813, pp. 828 – 832, 2007.
4. E. D. Leo, L. Galluccio, A. Lombardo, and G. Morabito, "Networked labs-on-a-chip (NLoC): Introducing networking technologies in microfluidic systems," *Nano Communication Networks*, vol. 3, no. 4, pp. 217 – 228, 2012.
5. L. F. Cheow, L. Yobas, and D.-L. Kwong, "Digital microfluidics: Droplet based logic gates," *Applied Physics Letters*, vol. 90, no. 5, pp. 054 107–1 – 054 107–3, 2007.
6. M. Prakash and N. Gershenfeld, "Microfluidic bubble logic," *Science*, vol. 315, no. 5813, pp. 832–835, 2007.
7. A. Alaghi and J. P. Hayes, "Survey of stochastic computing," *ACM Trans. Embed. Comput. Syst.*, vol. 12, no. 2s, pp. 92:1–92:19, May 2013.
8. B. Brown and H.Card., "Stochastic neural computation I: Computational elements," *IEEE Trans. on Computers*, vol. 50, no. 9, pp. 891–905, 2001.
9. D. Jenson and M. Riedel, "A deterministic approach to stochastic computation," in *Proc. IEEE/ACM Int. Conf. Computer-Aided Design*, 2016, pp. 1–8.
10. A. Dinu, M. N. Cirstea, , and M. McCormick, "Stochastic implementation of motor controllers," in *Proc. IEEE Int. Symp. Industrial Electronics*, 2002, pp. 639–644.
11. T. Hammadou, M. Nilson, A. Bermak, and P. Ogunbona, "A 96x64 intelligent digital pixel array with extended binary stochastic arithmetic," in *Proc. IEEE Int. Symp. Circuits and Systems*, 2003, pp. IV–772–IV–775.
12. V. C. Gaudet and A. C. Rapley, "Iterative decoding using stochastic computation," *Electronics Letters*, vol. 39, no. 3, pp. 299–301, 2003.
13. S. S. Tehrani, W. J. Gross, and S. Mannor, "Stochastic decoding of LDPC codes," *IEEE Commun. Lett.*, vol. 10, no. 10, Oct. 2006.
14. M. J. Fuerstman, A. Lai, M. E. Thurlow, S. S. Shevkoplyas, H. A. Stone, and G. M. Whitesides, "The pressure drop along rectangular microchannels containing bubbles," *Lab Chip*, vol. 7, pp. 1479–1489, 2007.
15. A. Biral, D. Zordan, and A. Zanella, "Modeling, simulation and experimentation of droplet-based microfluidic networks," *Trans. on Molecular, Biological, and Multi-scale Communications*, vol. 1, no. 2, pp. 122–134, 2015.
16. A. Grimmer, W. Haselmayr, A. Springer, and R. Wille, "A discrete model for Networked Labs-on-Chips: Linking the physical world to design automation," in *Proc. Design Automation Conference*, 2017.