On-Chip Electric Waves: An Analog Circuit Approach to Physical Uncloneable Functions

György Csaba Xueming Ju Qingqing Chen Wolfgang Porod
Jürgen Schmidhuber Ulf Schlichtmann Paolo Lugli Ulrich Rührmair

March 11, 2009

1 Abstract

This paper proposes the use of Cellular Non-Linear Networks (CNNs) as physical uncloneable functions (PUFs). We argue that analog circuits offer higher security than existing digital PUFs and that the CNN paradigm allows us to build large, unclonable, and scalable analog PUFs, which still show a stable and repeatable input–output behavior.

CNNs are dynamical arrays of locally-interconected cells, with a cell dynamics that depends upon the interconnection strengths to their neighbors. They can be designed to evolve in time according to partial differential equations. If this equation describes a physical phenomenon, then the CNN can simulate a complex physical system on-chip. This can be exploited to create electrical PUFs with high relevant structural information content.

To illustrate our paradigm at work, we design a circuit that directly emulates nonlinear wave propagation phenomena in a random media. It effectively translates the complexity of optical PUFs into electrical circuits.

Keywords Physical Uncloneable Functions (PUF), Analog circuits, Cellular Nonlinear Networks (CNN)

2 Introduction

Physical unclonable functions (PUFs) are emerging as a new, powerful approach to cryptography and security applications [1]. In the widest sense, a PUF is a mathematical function that is derived from the behavior of a complex, disordered physical object or device. Whenever the object is excited via external stimuli or challenges C_i, it reacts with corresponding responses R_i, meaning that we can regard its input–output behavior as a function from challenges to responses. Due to the high degree of nanoscale disorder or structural entropy inherent in the object, and due to its complicated internal interactions, it should both be hard to (i) rebuild or clone the PUF-object physically, and (ii) simulate or predict the PUF's input-output behavior numerically, even if a large number of challenge-response pairs (C_i, R_i) are known.

It is possible to develop a variety of cryptographic protocols which rest on the two said properties of PUFs. One of the assets of such protocols is that they can avoid many of the usual unproven

assumptions in cryptography (hardness of factoring/discrete log), albeit they rest on other, independent assumptions. Another advantage is that they can avoid the long-term storage of binary keys in non-volatile memory, where such keys can often be located and read out invasively.

Similar to many other cryptographic primitives, well-designed PUFs should combine some seemingly conflicting requirements in their behavior. On the one hand, they should be stable upon multiple measurements, robust against temperature variations, aging and environmental influences while still easy to manufacture and measure. On the other hand, their input–output behavior should be highly complex and sensitive against minuscule manufacturing variations. Even small variations should lead to a detectable change in many (ideally all) outputs. Local changes ideally should perpetuate and influence the behavior of the PUF globally.

The historically first attempt to resolve this tension between complexity, sensitivity and stability was the optical PUF suggested in [1]. This is a transparent object with many randomly distributed scatterers. Small changes in the position of only a few scatterers alter the entire interference pattern detectably. Stability, on the other hand, is guaranteed by the thermally inert behavior of the optical structure. Unfortunately, the described PUF requires on an expensive, bulky and delicate measurement apparatus.

Integrated, on-chip electrical PUFs [2, 3] with electrical inputs and outputs have been investigated soon after said optical PUF, since they promise higher practicality. Unfortunately, the implementations realized so far have been reverse engineered and imitated successfully by machine learning techniques [3, 4], breaking their security.

It is not difficult to see where the significant challenge in the design of highly complex electrical PUFs comes from. Electrical signals are highly susceptible to noise, temperature variations, voltage fluctuations and crosstalk. Thus, in order to maintain reproducibility, only relatively simple circuits were designed and built so far. They do not exhibit suffuciently high amount of relevant information content or highly complex internal interactions.

In order to circumvent said problems, we propose analog circuits with a Cellular Nonlinear Network (CNN) architecture in order to build scalable, highly complex electrical PUFs. Analog signals are very sensitive to the individual device characteristics of the circuit components, meaning that they lead to a larger amount of relevant random structural information per chip area than their digital counterparts. Their complex analog internal dynamics also strongly masks and obfuscates the internal random parameters, and hence provides a high degree of immunity against reverse engineering. Invasive attacks and microprobing can render analog signals unrecognizable. At the same time, the special circuit architecture of CNNs allows to large scalable analog arrays, while maintaining stability.

In particular, we investigate CNN designs that solve partial differential equations (PDEs) describing complex physical systems. Our motivation is that such PUFs will inherit their complexity from the suitably chosen PDEs. In other words, we use complex physical PDEs as a design guideline for highly secure, on-chip PUFs. To illustrate this concept at work, we discuss CNNs whose design is derived from Maxwell's equation, and which – electrically and on-chip – imitate the behavior of optical PUFs.

The rest of this paper is organized as follows: Section 3 formulates general requirements for highly secure circuit-based PUF implementations. Section 4 gives a brief overview of CNNs and shows how and why they are suited for our cause. Section 5 demonstrates that a CNN simulating a nonlinear wave equation fulfills the requirements stated in 3. Section 6 shows how this circuit can be built from elementary device components within a small chip area. The last section 7 discusses

the security features of this new type of PUF.

3 Specification of secure circuit PUFs

Based on the discussion in the introduction, we stipulate the following design goals for a highly secure circuit-based PUF:

1. It carries as much as possible structural information per chip area. As much of this information as possible is relevant in the chip's electrical behavior, meaning that it influences the responses R_i for most or many C_i.
2. A small and localized change in circuit parameters should globally alter the circuit behavior.
3. The characteristics of the circuit elements are difficult to measure invasively, and difficult to deduce from known challenge–response pairs (C_i, R_i).
4. There is a strong, non-linear, complex interaction between the different PUF-subcomponents.
5. The circuit is scalable to large sizes. The scaling should increase both the information content and the complexity of circuit operation, while maintaining the circuit stability.
6. The circuit operation should be stable and repeatable over time and relatively insensitive to temperature variations, noise, aging, power fluctuations.
7. The circuit is physically unique, and it is infeasible to build a clone which is accurate enough to show the same challenge–response behavior as the original.

Several of the requirements seem to be contradictory, or at least difficult to achieve simultaneously. They require a special circuit architecture that – despite being highly sensitive to its circuit parameters – still leads to a stable behavior in time. Special types of so-called Cellular Nonlinear Networks meet precisely just these requirements, which is why they represent a promising approach to highly secure and complex electrical PUFs.

4 Cellular Nonlinear Networks

4.1 General Introduction

Cellular Nonlinear Networks (or Cellular Neural Networks, CNNs) are analog computing arrays [5, 6] originally proposed as a realizable (scalable) alternative to Hopfield networks. They are built from locally interconnected circuit units or cells, which are arranged on a regular grid. The grid is typically two-dimensional, and all cells are connected via analog connections to their neighbors. Each cell is characterized by a dynamical state variable, which obeys an ordinary differential equation (ODE).

A schematic illustration of the CNN architecture is given in Fig. 1a. The time evolution of the state variable is described by the the following ODE:

$$\dot{x}_{ij} = -x_{ij} + \sum_{k,l} \mathbf{A}_{i,j,k,l} y_{kl} + \sum_{k,l} \mathbf{B}_{i,j,k,l} u_{kl} + z_{ij}$$

i.e. the time derivative of the state variable (for cell with i, j indices) depends on the y output of the neighboring cells (denoted by the k, l indices) via a the **A** cloning templates. Each cell has a bias (z) and inputs, which are coupled by the **B** template to the dynamical equation. The CNN behavior can be programmed by choosing appropriate templates. If only nearest neighbors are coupled, then the **A** and **B** templates are 3×3 matrices.

The output of each cell is sigmoid-like (saturating) function of the state variable, for example:

$$y_{ij} = f(x_{ij}) = \frac{1}{2}|x_{ij} + 1| - |x_{ij} + 1|$$

CNNs often have multiple layers and these layers are also coupled to each other via **B** templates.

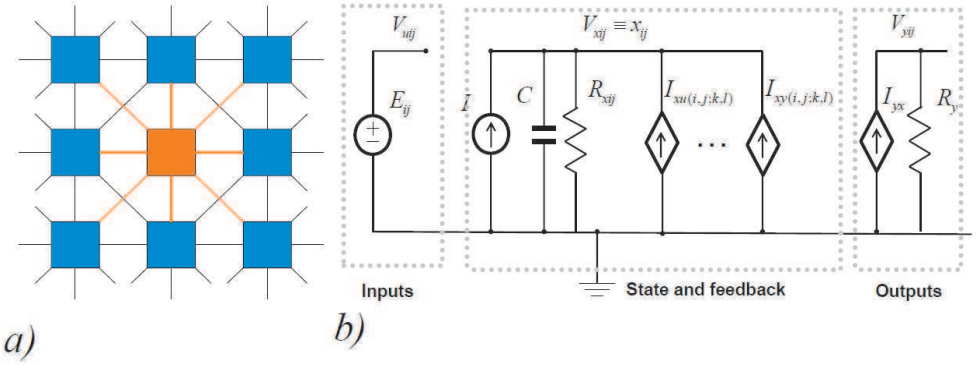

Figure 1: Panel (a) shows the topology of a CNN circuit with nearest-neighbor couplings. Panel (b) is the circuit schematics of a single cell, realized by controlled current and voltage sources.

As a mathematical model, CNNs are very general. For example, cellular automata [7] can be interpreted as a special CNN which operates on discrete variables in discrete time (and where rules replace the ODE-based description). CNNs are also known to be Turing-complete [8]. Furthermore they have an extraordinary computational efficiency: for many tasks (typically those which involve large number of local operations) they are several orders of magnitude faster than standard Boolean microprocessors [9], since the time evolution of thousands of cells goes on in parallel.

It is possible to construct a relatively simple circuit that solves the CNN equation 'by hardware'. This circuit is sketched in Fig. 1b. The state variable is represented by the voltage (or charge) on a capacitor. Controlled current and voltage sources are charging this capacitor and generate the outputs according to the CNN equation.

4.2 Physical Uniqueness and the Role of Mismatches

A CNN cell, which is realized from solid-state electronic components does not follow precisely the ideal CNN equations. Parameter deviations of the integrated circuit elements [10] result in the deviation of templates from their nominal values. These deviations can be introduced by the fabrication process, noise, temperature, device degradation and noisy input signals [11]. Such mismatches represent a critical problem in scaling analog circuits in the sub-micrometer range.

Templates, which are called *robust* in the CNN literature are tolerating certain amount mismatch. A CNN with not robust templates may show a unique challenge-response behavior and

serve as a circuit-PUF.

4.3 Read-Out Formalism for CNN-PUFs

Our proposed CNN-PUF is built from nominally identical, fixed-template cells (with or without an output/bias). The excitation vector C_i is applied as an input or fixed value on some cells. It triggers an excitation (wave) that propagates through the circuit and possibly bounces back and forth between the boundaries. The circuit may go to a stationary state after some time or oscillate indefinitely, depending on the choice of the **A** and **B** templates. The steady-state values or, alternatively, the averaged oscillating voltages can serve as the response R_i.

The resulting response vector R_i should be sensitive to the deviations of the templates from their nominal value (cell mismatches) and consequently carries a signature from the individual device characteristics from each cell of the CNN. Note that even if the cells are only locally connected, far-away cells influence each other indirectly due to the propagation effects. Quantitative values for this influence can be obtained by simulations, and will be discussed in the next sections.

4.4 CNNs and Partial Differential Equations

CNNs are intimately related to Partial Differential Equations (PDEs), too [12, 13], a fact which we are going to exploit in this paper. Given a two-dimensional, continuous, one variable, time dependent PDE which was discretized in space (but not in time) on an $n \times m$ lattice, there are standard methods to design CNNs to directly simulate this discretized system – the CNN can be tuned to physically evolve according to a certain PDE. More precisely, we can derive a CNN template from the PDE such that the state variables of the CNN cells evolve over time in the same way to the states of the lattice points in the discretized $n \times m$ lattice. The necessary CNN templates, which enforce the right CNN behavior, can be directly determined from the finite difference approximation of the PDE.

We will demonstrate this procedure exemplarily for the case of the Maxwell equations and nonlinear optical behavior in Section 5, deriving a concrete template that enforces a time evolution according to Maxwell's equations. As we will exemplify in Section 6, it is furthermore possible to translate this abstract template architecture into a concrete, relatively simple circuit block on the transistor level. This leads to cellular PUF-circuits that can be commercially ordered on demand or fabricated in large volumes [14].

5 Case Study: Wave Propagation on a Chip

There is an infinite number of possible CNN templates, but there is no general, systematic method how to construct templates for a given set of requirements. As said earlier, one promising possibility to meet our specifications of Section 3 is to employ CNN templates which realize well-known PDEs describing physical systems. Then the CNN inherits its complex dynamics from the complex behavior described by these PDEs.

As we discussed in Section 2, the appeal of optical PUFs is coming from the 'global' interactions that define the interference patterns. Therefore we will now investigate a CNN which solves a linear wave equation known from electromagnetic theory, aspiring that it will inherit the typically optical

feature of non-local interactions. For weak excitations, the CNN solves a linear scattering problem (i.e. the original PDE). For stronger excitations, the nonlinear cell-cell dynamics comes into play.

5.1 The Maxwell equation for TE waves

Our CNN design is based on a simple numerical technique that rests on Zuse's seminal work [15], [16] for solving Maxwell's equations in two dimensions for TE (transversal electric) waves [17]. Using $\mu_0 = \epsilon_0 = 1$, Maxwell's equations take the form:

$$\begin{aligned} \operatorname{curl} \mathbf{H} &= \frac{d\mathbf{E}}{dt} \\ \operatorname{curl} \mathbf{E} &= -\frac{d\mathbf{E}}{dt} \\ \operatorname{div} \mathbf{E} &= 0 \\ \operatorname{div} \mathbf{H} &= 0 \end{aligned}$$

For a TE field (propagating in the x-y plane), the electric field vector has only an E_z component, while the magnetic field bears the H_x and H_y components. A continuous time, spatially discretized form of the wave equations can be obtained by using a second-order, central finite difference approximation for the spatial deriatives:

$$\begin{aligned} \frac{dH_x^{i,j}}{dt} &= -\frac{E_z^{i,j+1} + E_z^{i,j-1} - 2E_z^{i,j}}{\Delta y^2} \\ \frac{dH_y^{i,j}}{dt} &= \frac{E_z^{i+1,j} + E_z^{i-1,j} - 2E_z^{i,j}}{\Delta x^2} \\ \frac{dE_z^{i,j}}{dt} &= -\left(\frac{H_y^{i+1,j} + H_y^{i-1,j} - 2H_y^{i,j}}{\Delta x^2} - \frac{H_x^{i,j+1} + H_x^{i,j-1} - 2H_x^{i,j}}{\Delta y^2} \right) \end{aligned}$$

Here Δx and Δy are the step width of the spatial discretization. We used $\Delta x = \Delta y = 5$ for all the example simulations. The 'CNN-hardware' that evolves according to the above equations (and thereby computes its solution) must be implemented as a three-layer CNN: One layer for each of the E_z, H_x and H_y variables. Comparing the above equations with the CNN dynamical equations, the templates can be directly determined. The H_x and H_y layers are bi-directionally coupled to the E_z layer, and there is no direct coupling between H_x and H_y layers. The corresponding templates are:

$$\mathbf{B}^{E_z \to H_x} = \frac{1}{(\Delta x)^2} \begin{bmatrix} 0 & 0 & 0 \\ -1 & 2 & -1 \\ 0 & 0 & 0 \end{bmatrix} \quad \mathbf{B}^{H_x \to E_z} = \frac{1}{(\Delta x)^2} \begin{bmatrix} 0 & 0 & 0 \\ 1 & -2 & 1 \\ 0 & 0 & 0 \end{bmatrix}$$

$$\mathbf{B}^{E_z \to H_y} = \frac{1}{(\Delta y)^2} \begin{bmatrix} 0 & 1 & 0 \\ 0 & -2 & 0 \\ 0 & 1 & 0 \end{bmatrix} \quad \mathbf{B}^{H_y \to E_z} = \frac{1}{(\Delta y)^2} \begin{bmatrix} 0 & -1 & 0 \\ 0 & 2 & 0 \\ 0 & -1 & 0 \end{bmatrix}$$

All self-feedback templates are:

$$\mathbf{A} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

5.2 Simulation of CNN-dynamics: sensitivity and complexity

We numerically studied the behavior of larger CNNs by custom-built Matlab and C++ based simulators. We used a simple Runge-Kutta scheme for calculating the CNN dynamics integrating the above ODEs [18].

If all the templates are fixed at their nominal value and the excitation vector is a single, fixed-value cell placed at the center, then the CNN generates a wave, which is illustrated in Fig. 2a). Initially, all state variables of the circuit were zero and the circuit does not cross into the nonlinear regime.

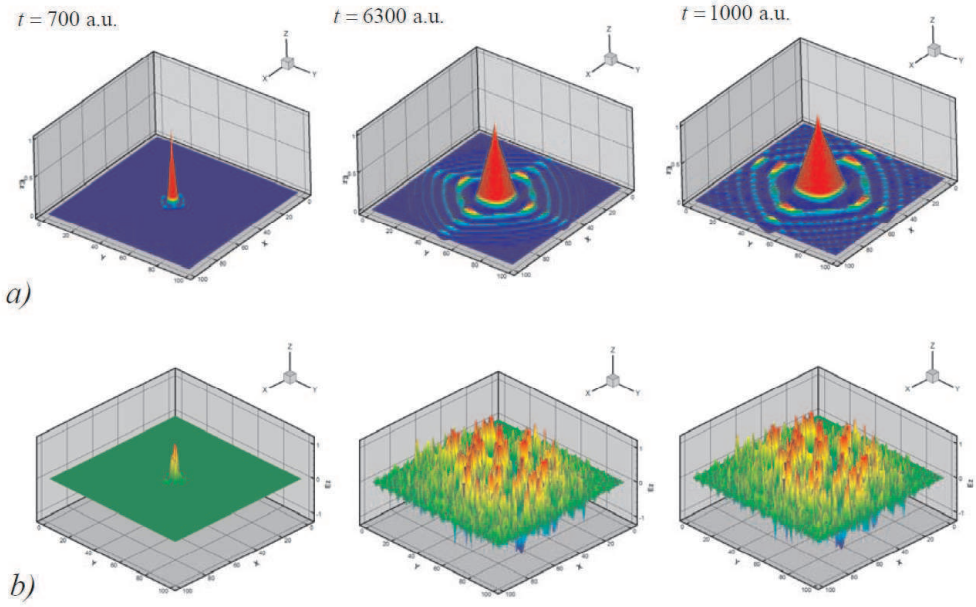

Figure 2: Simulation results of the wave-CNN, with a single excitation of the center. For identical cells, a characteristic wave pattern is formed (panel a)). For mismatched templates complex, non-symmetric interferences appear (panel b)).

The bottom panel of Fig. 2 shows snapshots from simulations of a mismatched CNN. All the elements of the **B** templates (matrices) were varied by $\sigma = 0.03$. The initial similarity to the non-mismatched case quickly disappears, and non-periodic nonlinear waves appear in the structure.

We found that for mismatched templates, the dynamic behavior never stays in the linear range, as even for very weak excitations signals rapidly amplify. The mismatches automatically lead into a non-linear regime, which is desirable in order to increase the system's complexity. The sigmoid-type nonlinearity has the further positive effect that it forces all the signals (circuit variables) to stay in the $[-1; +1]$ range.

Physically, the mismatched **B** templates essentially are equivalent to a nonlinear, non-isotropic and optically active gain material. This 'world' is much more rich than the linear scattering effects used in [1]. On the other hand, the CNN model is only two-dimensional unlike the 3D token described there [1]. A more direct CNN analogy of the linear scattering PUF could also be built by placing fix zero-valued cells (obstacles, scatterers) at some (random) positions but this 'randomization strategy' realizes only lower information content and density, and is to some extent unnatural in the circuit implementation case, since there are better sources of random information and uniqueness there.

Simulation results confirm the sensitivity of this CNN for both the initial (input) values and the templates.

A particular simulation result for the input sensitivity is sketched in Fig. 3. We placed one source (a fixed-value cell) in the center of the CNN array - all the other cells started from a zero initial value. The averaged absolute value ($|E_z|$) of some randomly picked cell outputs is plotted as a function of this source value. The function is oscillating with a high amplitude even far away from the source and the curves belonging to nearby cells seem to be uncorrelated. It strongly indicates that this structure fulfills the criteria we set up for the input sensitivity.

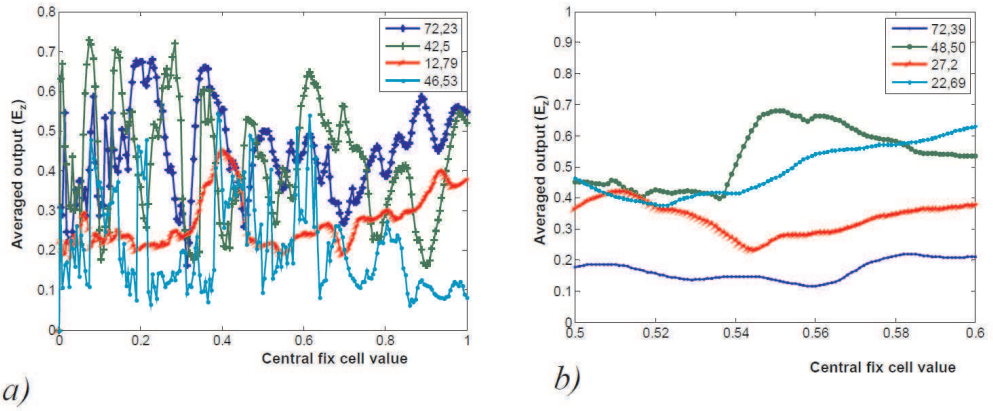

Figure 3: Sensitivity of the CNN for one particular input. The averaged output value is shown as a function of the applied fixed input. A close-up view (from a different simulation) shows that despite the 'chaotic' appearance of the signals on panel a), they are changing continously.

The simulations of Fig. 4 confirm that this structure sensitively responds to the change of circuit parameters (templates) as well. Changing only a single template at a particular position (denoted by **B** in the figure), even far away from an input (marked as *In* in the figure) going to alter the global behavior of the circuit detectably.

These simulation results confirm that this circuit inherits the 'global sensitivity' property from optical PUFs, which make those devices so appealing. The circuit behavior also appears to be complex, as it is expected from a nonlinear dynamic system with many degrees of freedom. Work is in progress to set up machine learning experiments on simulated data to confirm that our circuit indeed withstands such types of attacks.

[1] Three-dimensional CNNs can be straightforwardly defined, but they cannot be realized on a large scale by planar IC technology.

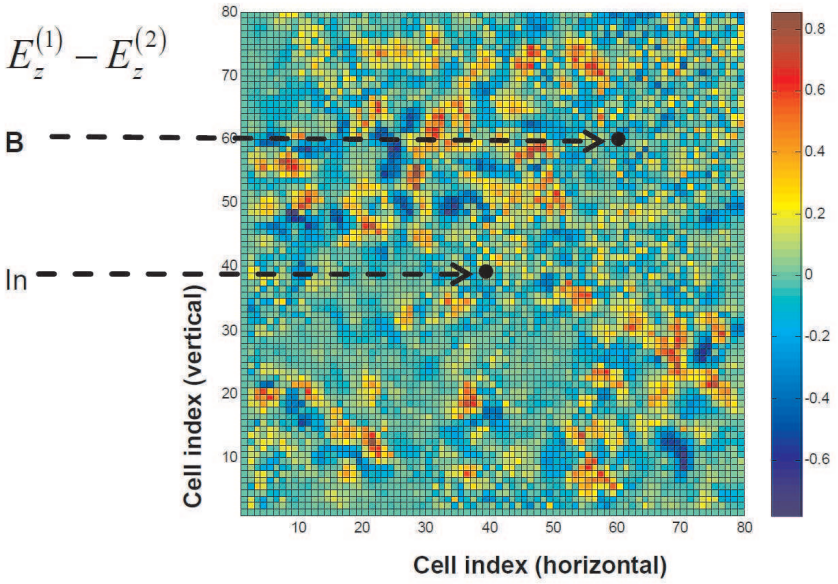

Figure 4: Sensitivity of the CNN to change in a particular template - the contour plots gives the difference of E_z if a single template is altered.

Another very important characteristics of our circuit that its behavior is sensitive, but is not chaotic. Chaotic circuits are well known [19] and several CNN templates are known to realize chaos [20] [21] [22]. The time trajectories of a chaotic system are irreproducable in a real physical environment and seem to be unsuitable as a PUF.

5.3 Effect Propagation and Read-Out Speed

The interactions in the described cellular structures are propagating with a finite speed. If the dynamics of the system is interrupted after a short time then only cells lying within a finite neighborhood can influence each other. Changing the length of the time evolution gives a possibility to balance 'global sensitivity' against error tolerance and robustness of the circuit.

Strongly depending on the template that we choose, the development of the full pattern *can* take considerable time. Assuming that the time constant of a single cell is Δt, and the circuit is composed of $n \times n$ cells, the entire pattern (with the signals bouncing back and forth between the boundaries) can take several times $n\Delta t$ time to develop.

This provides us with the possibility to design CNN-PUFs which take *intendedly* long time in order to develop robust outputs, and even to set the read-out speed to an intrinsic, predefined value.

This leads the way for an extra security feature of CNN-PUFs: it limits the rate at which data and information can be gathered from the circuit, making the task of reverse engineering or machine learning more difficult.

6 Circuit implementation of CNN-PUFs

So far, we described the behavior of CNNs on an abstract, template-based level. We now investigate which design on a transistor level is necessary to enforce this behavior in a real circuit. This will allow us to estimate the information content per chip area and the scalability of the structure.

A non-programmable (fixed-template) CNN cell can be built using three operational amplifiers. Additional passive elements (resistors) set the cell to cell coupling via the cloning templates, and the feedback. The circuit schematics is shown in Fig. 5.

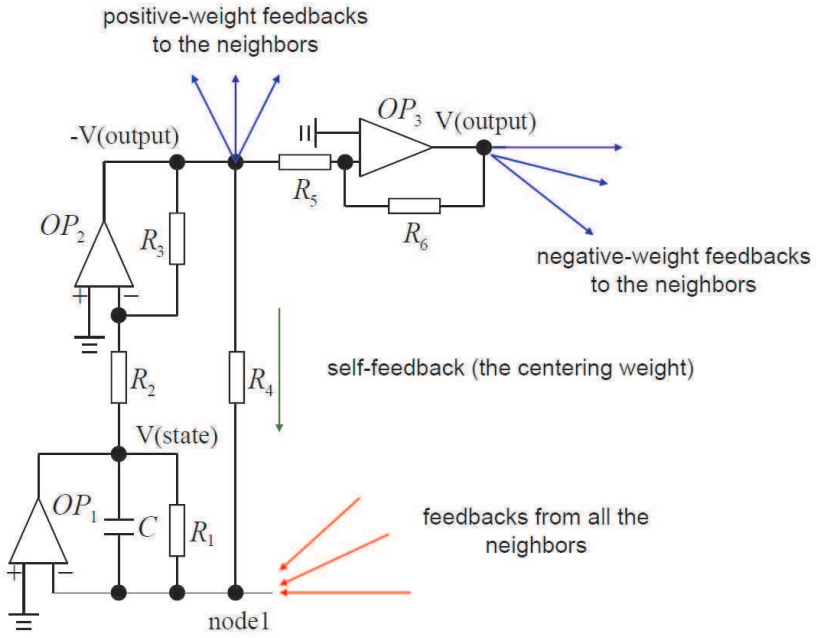

Figure 5: Circuit schematics of a non-programmable CNN cell.

The state variable is represented by the voltage of the capacitor. OP_1 is used for limiting the state voltage and connecting one end of the capacitor (node1) to virtual ground. OP_2 , R_2 and R_3 work as an inverting amplifier which can realize the full nonlinear transfer characteristic to map V(state) nonlinearly to -V(output). R_3 is usually several times larger than R_2 so as to have the OP_2 driven to the saturation region. Another inverting amplifier (OP_3, R_5, R_6) is used to provide the actual V(output). The weights in the **A** template can be controlled by changing the resistances of feedback resistors, e.g. R_4. The weight of the **B** templates are set by resistor values connecting from the neighboring cells to node$_1$ Kirchoff's current law applied for node$_1$ can exactly present the CNN state equation we have discussed in Section 4.

The simplest CMOS operational amplifiers can be built using seven transistors, so the circuit of Fig. 5 requires 21 transistors. Considering that a state of the art CPU contains a few times $\approx 10^7$ circuit elements, an approximately 1000×1000 cell fixed-template CNN could be reasonable to build [23]. Each template carries at least a few bytes of information in its deviation from the nominal value. Obviously, some part of this information could be lost because of averaging / error correction, but the information content of the CNN-PUF should be comparable or even higher as

the optical PUF of [1].

Using a modest workstation and standard SPICE distribution [24], we could verify the operation of few-hundred cell (few thousand transistor) circuits, and the results agree with the results from the template-based description. Only highly parallelized, research-distributions of SPICE, running on supercomputers (such as Xyce [25]) could deal with the few-million transistor circuits that could eventually be envisaged as CNN-PUFs. This is a strong indication of the security of our CNN-PUF approach: Even if all the parameters of the circuit are known, it still takes hours for supercomputer to simulate the few microsecond or millisecond behavior of the CNN-PUF. Reverse engineering of such a circuit would thus be formidably difficult.

Error Correction and Stability. An important benefit of circuit-based PUFs is that there are efficient circuit solutions to minimize the output instability of the circuit. For example, bandgap references can provide temperature-independent voltage sources, albeit they make the circuit more complicated and slightly reduce the achievable information content per chip area.

We could not yet perform extensive simulations on large-scale circuits to estimate the effects of temperature, noise, power supply fluctuations, etc. on the circuits, as it requires extreme amount of computer power. It is known, however that cell to cell mismatch in CNN circuits dominates over temperature effects [11, 26], which is particularly important in our context. The templates are set by resistance ratios, so if those close-by resistors are at the same temperature, the temperature dependence of the templates will become very small. As the template-based simulations in Fig. 3 and Fig. 4 suggest, a one-percent change in circuit parameters does not make the circuit dynamics unrecognizable.

In addition, the response vector R_i is read-out as a result of a stationary process. While R_i depends on the internal dynamics / timing of the circuit, noise, glitches may be averaged out, increasing the stability of the circuit.

7 Conclusions: Security Assessment of CNN-PUFs

This paper proposed CNN circuits with non-robust templates (i.e. CNNs which are sensitive to uncontrollable variations in their circuit parameters) as promising circuit implementations of PUFs. We argued that (i) analog circuits, in general, yield to higher security than digital ones (ii) the CNN paradigm (or a similar cellular structure) is among the very few viable possibilities to build scalable analog arrays. Based on an physical analogy, we designed a template that inherits the benefits of optical PUFs (such as high sensitivity, no averaging out effects, global interactions) and, on top of that, also displays nonlinear behavior. The complex internal interactions probably eliminate the possibility to construct a simple computationally non-intensive model of such a circuit.

Full characterization of the circuit by a faker is further complicated by the fact that the stationary (steady state) behavior of the CNN can be designed in such a manner that it takes time to develop; we could call such a circuit a *'slow read-out CNN-PUF'*. A brief analysis showed that the read-out time for one CRP can easily be put in the order of several milliseconds. This feature can make it complicated for a faker to obtain the large number of challenge-response pairs that he might need for reverse engineering or machine learning.

We could not give 'hard' computational limits on the difficulty of reverse engineering and simulating the behavior of a random (mismatched) CNN with the described template. Note, however,

that providing such provably hard limits may be beyond the current state of complexity theory anyway. We refer in this context to the unsettled NP vs. P question and the general difficulty of giving hard, meaningful and non-linear boundaries for natural problems in NP.

Nevertheless, it can be argued convincingly that, based on the proven computational power of CNN chips [9] [27], [28] their large internal information content and their parameter sensitivity, the use of CNNs as PUFs seems very promising. It may eventually yield to the highest security achievable by circuit-based PUFs with interacting components.

8 Acknowledgments

The authors are grateful for enjoyable discussions and excellent suggestions to Martin Stutzmann, Peter Vogl, Tamas Roska and Arpad Csurgay.

References

[1] R. Pappu, B. Recht, J. Taylor, N. Gershenfeld, *Physical One-Way Functions*, Science, vol. 297, pp. 2026-2030, 20 September 2002.

[2] Blaise Gassend, Daihyun Lim, Dwaine Clarke, Marten van Dijk, Srinivas Devadas: *Identification and authentication of integrated circuits.* Concurrency and Computation: Practice & Experience, pp. 1077 - 1098, Volume 16, Issue 11, September 2004.

[3] Daihyun Lim; Lee, J.W.; Gassend, B.; Suh, G.E.; van Dijk, M.; Devadas, S., *Extracting secret keys from integrated circuits,* IEEE Transactions on Very Large Scale Integration (VLSI) Systems, vol.13, no.10, pp. 1200-1205, Oct. 2005

[4] —: *Cryptoanalysis of Physical Unclonable Functions by Machine Learning Techniques*, Manuscript in Preparation, 2009.

[5] L. O. Chua and T. Roska: *Cellular Neural Networks and Visual Computing: Foundations and Applications* Cambridge University Press 2005

[6] Chua, L.O. and L. Yang: *Cellular Neural Networks: Theory*, IEEE Transactions on Circuits and Systems, vol. 35, pp. 1257-1272.

[7] S. Wolfram: *Statistical mechanics of cellular automata*, Rev. Mod. Phys. 55, 601 - 644 (1983)

[8] Roska, T.; Chua, L.O., *The CNN universal machine: An analogic array computer.* Circuits and Systems II: IEEE Transactions on Analog and Digital Signal Processing, vol.40, no.3, pp.163-173, Mar 1993

[9] Chua, L.O.; Roska, T.; Kozek, T.; Zarandy, A., *CNN universal chips crank up the computing power* Circuits and Devices Magazine, IEEE , vol.12, no.4, pp.18-28, Jul 1996

[10] Bhavnagarwala, A.J.; Xinghai Tang; Meindl, J.D., *The impact of intrinsic device fluctuations on CMOS SRAM cell stability* Solid-State Circuits, IEEE Journal of , vol.36, no.4, pp.658-665, Apr 2001

[11] S. Xavier-de-Souza, M. Yalcin, J. Suykens, and J. Vandewalle, Toward CNN Chip-Specific Robustness, IEEE Trans. On Circuits And Systems - I, 51(5): 892-902, 2004.

[12] T. Roska, L.O. Chua, D. Wolf, T. Kozek, R. Tetzlaff, F. Puffer: *Simulating nonlinear waves and partial differential equations via CNN-Part I: Basic techniques*, IEEE Transaction on Circuits and Systems-I, vol. 42, pp. 807-815, 1995.

[13] Serpico, C., Setti, G., and Thiran, P. 1997. *Analogies between cellular neural networks and partial differential equations*. In: Advances in intelligent Systems, F. C. Morabito, (Ed.) IOS Frontiers In Artificial Intelligence And Applications Series, vol. 41. IOS Press, Amsterdam, The Netherlands, 157-162.

[14] For example, see: http://www.anafocus.com/

[15] Konrad Zuse: *Calculating space*. MIT Technical translation, orig: K. Zuse: *Rechnender Raum*, Schriften zur Datenverarbeitung, 1 Friedr. Vieweg & Sohn, Braunschweig, 1969.

[16] Wolfgang Porod, Henry K. Harbury, and Craig S. Lent, *Study of Wave Phenomena in Physically- Coupled Device Arrays Using the Helmholtz Equation as a Model*, Fourth Workshop on Physics and Computation - PhysComp96, Boston, Massachusetts, November 1996.

[17] J. D. Jackson: *Classical electrodynamics* Wiley, 1998

[18] William H. Press, Brian P. Flannery, Saul A. Teukolsky, William T. Vetterling: *Numerical Recipes in C: The Art of Scientific Computing* Cambridge University Press; 2 edition (October 30, 1992) or www.nr.com/

[19] Kennedy, M.P., *Three steps to chaos. II: A Chua's circuit primer*, IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications, vol.40, no.10, pp.657-674, Oct 199

[20] Zou, F. and J.A. Nossek, *A chaotic attractor with cellular neural networks*, IEEE Transaction on Circuits and Systems, Vol. 38, pp. 811-812, 1991.

[21] Maciej J. Ogorzalek, Zbigniew Galias, Andrzej M. Dqbrowski, Wladyslaw R. Dqbrowski: *Chaotic Waves and Spatio-Temporal Patterns in Large Arrays of Doubly-Coupled Chua' s Circuits,* IEEE Transactions on Circuits and Systems-I: Fundamental Theory and Applications, Vol. 42, No. 10, October 1995

[22] M. Gomez-Gesteira, M. de Castro, V. Perez-Villar, L. O. Chua: *Experimental Chua's Circuit Arrays As an Autowave Simulator,* IEEE Transactions on Circuits and Systems-I: Fundamental Theory and Applications, Vol. 46, No. 4, April 1999.

[23] Cellular wave computers for nano-tera-scale technology - beyond spatial- temporal logic in million processor devices ELECTRONICS LETTERS 12th April 2007 Vol. 43 No. 8

[24] http://www.linear.com/designtools/software/

[25] http://xyce.sandia.gov/

[26] Tamas Roska, private communication

[27] L. O. Chua: *CNN: A paradigm for complexity* World Scientific Pub. Co Juni 1998

[28] M. Ercsey-Ravasz, T. Roska, Z. Neda: Cellular Neural Networks for NP-Hard Optimization EURASIP Journal on Advances in Signal Processing Volume 2009, Article ID 64697