A Faithful Binary Circuit Model

Matthias Függer, Robert Najvirt, Thomas Nowak, and Ulrich Schmid

***Abstract*—Függer *et al.* (2016) proved that no existing digital circuit model, including those based on pure and inertial delay channels, faithfully captures glitch propagation: for the short-pulse filtration (SPF) problem similar to that of building a one-shot inertial delay, they showed that every member of the broad class of bounded single-history channels either contradicts the unsolvability of SPF in bounded time or the solvability of SPF in unbounded time in physical circuits. In this article, we propose binary circuit models based on novel involution channels that do not suffer from this deficiency. Namely, in sharp contrast to bounded single-history channels, SPF cannot be solved in bounded time with involution channels, whereas it is easy to provide an unbounded SPF implementation. Hence, binary-valued circuit models based on involution channels allow to solve SPF precisely when this is possible in physical circuits. Additionally, using both SPICE simulations and physical measurements of an inverter chain instrumented by high-speed analog amplifiers, we demonstrate that our model provides good modeling accuracy with respect to real circuits as well. Consequently, our involution channel model is not only a promising basis for sound formal verification but also allows to seamlessly improve existing dynamic timing analysis.**

***Index Terms*—Binary circuit models, glitch propagation.**

I. Introduction

MODERN digital circuit design relies heavily on fast timing analysis techniques. For synchronous designs, state-of-the-art *static* timing analysis tools like Synopsis prime time are able to very accurately predict the timing behavior of a given circuit design, and to identify setup/hold-violations and other timing-related problems. Such tools are based on elaborate timing prediction models like CCSM [1] and ECSM [2]. These models characterize the delay of a cell via (typically manufacturer-supplied) technology data. This data can include tabulated input/output current waveforms for varying parameters, such as input slew rate and output capacitive load [3]. However, the timing predictions provided by static timing analysis tools do not involve any dynamic (signal trace-related) considerations.

By contrast, *dynamic* timing analysis techniques rely on signal traces generated by a circuit, in response to appropriately setup test vectors. The "golden standard" here are fully fledged analog simulations, e.g., using SPICE [4], which are based on detailed analog models of all elements of a digital standard-cell library. Since SPICE simulation times of even moderately complex circuits are prohibitively excessive. However, designers have to resort to digital timing analysis/simulation tools for Mentor Graphics ModelSim, Cadence NC-Sim, or Synopsis VCS for those parts of a circuit where, e.g., the presence of glitch trains may severely affect the correctness and power consumption. Such tools are based on *discrete-value* (typically binary) circuit models augmented by continuous-time delays. More specifically, gate and wire delay estimates obtained via CCSM or ECSM, for example, are used to parameterize pure or inertial delay [5] channels (e.g., in VHDL-Vital or Verilog timing libraries). The resulting executable HDL simulation models are then used in subsequent simulation and dynamic timing analysis runs. Clearly, the precomputed delays are constants here, i.e., remain the same throughout these runs. More accurate results can be expected from the degradation delay model (DDM), introduced by Bellido-Díaz *et al.* [6], [7], which allow channel delays to vary dynamically in a trace.

However, binary-valued circuit models do not only facilitate accurate performance and power estimation [8], [9] of complex circuits at early design stages: they also pave the way to formal verification of complex circuits. A main driver for verification of digital circuits is its potential to uncover race conditions, hazardous glitches and other corner-case effects relevant for timing-closure analysis. Obviously, such capabilities rest critically on suitable foundations for a rigorous and complete timing analysis of complex circuits. The first thing to note in this context is that statements about the correctness of a circuit *in a model* are meaningful only if they also imply correctness of the corresponding *real* circuit implementation. We call a model *realistic*, if a given problem can be solved in the model if and only if it can be solved by a real circuit. A model is *faithful* if it is both realistic and provides accurate timing predictions. The first question to ask is whether existing binary circuit models are faithful.

A. Short-Pulse Filtration

Függer *et al.* [10] studied the faithfulness of existing binary circuit models with respect to glitch propagation. More specifically by their ability to solve the simple short-pulse filtration (SPF) problem, which is essentially the problem of building a one-shot variant of an inertial delay channel: as for inertial delay channels, no short pulses may appear at the SPF output. In the case of long input pulses,

Manuscript received August 16, 2018; revised February 18, 2019 and March 25, 2019; accepted July 26, 2019. Date of publication August 28, 2019; date of current version September 18, 2020. This work was supported in part by the Austrian Science Fund (FWF) through Project SIC under Grant P26436, Project RiSE under Grant S11405, and Project DMAC under Grant P32431, in part by the Centre National de la Recherche Scientifique Project PEPS DEMO, and in part by DigiCosme (Working Group HicDiesMeus). This article was recommended by Associate Editor Y. Shi. *(Corresponding author: Thomas Nowak.)*

M. Függer is with the Centre National de la Recherche Scientifique, Laboratoire Spécification et Vérification, ENS Paris-Saclay, Univeristé Paris-Saclay and Inria, 94235 Cachan, France (e-mail: mfuegger@lsv.fr).

R. Najvirt and U. Schmid are with the Embedded Computing Systems Group, Vienna University of Technology, 1040 Vienna, Austria (e-mail: rnajvirt@ecs.tuwien.ac.at; s@ecs.tuwien.ac.at).

T. Nowak is with the Laboratoire de Recherche en Informatique, Centre National de la Recherche Scientifique, Université Paris-Sud, 91405 Orsay, France (e-mail: thomas.nowak@lri.fr).

Digital Object Identifier 10.1109/TCAD.2019.2937748

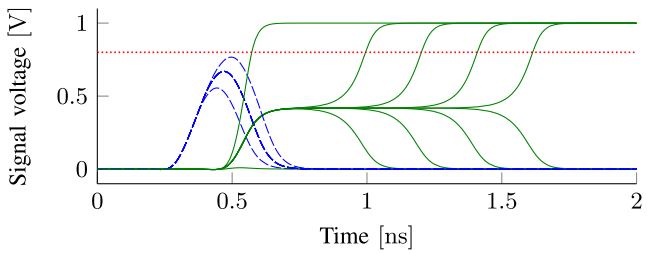

Fig. 1. Analog simulation traces of a CMOS SPF, implemented as a storage loop followed by a high-threshold buffer. The dashed (blue) curves represent the input signal, the solid (green) ones give the output of the storage loop. The horizontal line at 0.8 V marks the threshold level.

however, they need not be passed unaltered. In particular, the SPF output may settle at logical 1 even if the input does not. The stronger variant of *bounded* SPF requires the output to settle in bounded time.

The (un)solvability of (bounded) SPF is indeed a suitable test for a model's ability to faithfully model glitch propagation with respect to physical circuits: on the one hand, Marino [11] formally proved that problems like SPF cannot be solved in a physical model when the output is required to stabilize in bounded time [10]. On the other hand, a simple storage loop with a high-threshold buffer at its output (see Fig. 5) solves SPF in unbounded time: as shown in the SPICE simulation traces in Fig. 1, sufficiently large input pulses (largest blue dashed one) just cause the storage loop to change its state (to 1) instantaneously (left-most green solid one), very small input pulses (smallest blue dashed one) do not affect the storage loop (bottom green solid one). Critical input pulses (middle blue dashed ones, overlapping, therefore appearing as if they were one pulse) cause the storage loop to become metastable for an unbounded time, eventually resolving to either state 0 or 1. Therefore, appending a high-threshold buffer with a threshold (marked by the red dotted line) clearly above the metastability region results in a clean (= nonmetastable) output signal, which either remains at 0 or makes a single (possibly delayed) transition to 1. Hence, with real circuits, SPF is solvable, while its stronger, bounded, variant is not.

B. Single-History Channels

The circuit model used in [10] combines zero-time Boolean gates with *single-history channels* that model circuit delays. They are primarily characterized by a delay function δ that maps a transition occurring at the channel input at time t to its corresponding output transition at time $t + \delta(T)$, where T is the previous-output-to-input delay. Fig. 2 shows two examples. Note that single-history channels not only allow to model decaying pulse propagation, but also vanishing pulses: if two succeeding input transitions would, according to $\delta(T)$, occur at the output in reversed order, they cancel each other. Furthermore, single-history channels allow for different rising and falling transition delays, specified by two delay functions $\delta_\uparrow$ and $\delta_\downarrow$, respectively.

Well-known instances of single-history channels are pure delay channels and inertial delay channels [5]; a more advanced example are DDM channels [6], [7]. They are all *bounded single-history channels*, where the delay functions are upper- and lower-bounded. Függer *et al.* [10] proved that

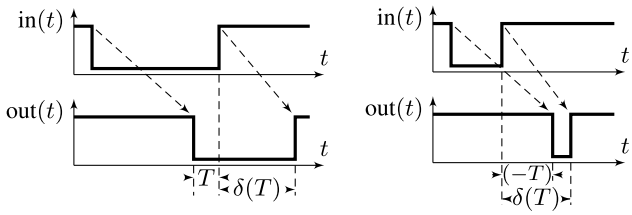

Fig. 2. Left: input/output signal of a single-history channel, involving the previous-output-to-input delay T and the resulting input-to-output delay $\delta(T)$. Right: input transition with $T < 0$.

no bounded single-history channel can be faithful: binary circuit models based on channels with pure (= constant) delays do not allow to solve unbounded SPF. Bounded single-history channels with nonconstant delays, including inertial delay and DDM channels, allow to design circuits that solve bounded SPF. Since this contradicts reality, as argued above, no existing binary circuit model is faithful.

C. Main Contributions and Paper Organization

In this article, we propose a class of single-history channels with *unbounded* delay functions: like their bounded counterparts, their delay is upper bounded; however, it is not bounded from below. These negative delays turn out to be crucial for accurately modeling glitch suppression. We coined the term *involution channels* for them, as we require their negative delay functions to be involutions, i.e., $-\delta(T)$ must form its own inverse. To increase the coverage of our class of involution channels, we actually allow the delay functions $\delta_\uparrow$ and $\delta_\downarrow$ for rising and falling transitions to be different, and require $-\delta_\downarrow(-\delta_\uparrow(T)) = T$ and $-\delta_\uparrow(-\delta_\downarrow(T)) = T$. We prove that the solvability/unsolvability border of SPF in a binary-valued circuit model based on our involution channels is the same as in reality, and that the resulting model also accurately captures the behavior of real circuits.

1) In Section II, we demonstrate that the standard first-order model used, e.g., in [12] actually gives rise to a simple instance of general involution channels. They are introduced formally in Section IV. Assuming delay functions to be involutions is hence neither artificial nor (as our simulations and experiments reveal) inaccurate.
2) In Section III, we provide our binary circuit model, as well as the SPF problem. In Section V, we explain how to use our model to explicitly construct output and intermediate signals of a circuit given the input signals, i.e., how to perform circuit simulation.
3) In Section VI, we prove that the simple circuit consisting of a storage loop and a high-threshold buffer solves unbounded SPF in the involution channel model.
4) In Section VII, we show that bounded SPF is impossible to solve with involution channels. In a nutshell, our proof inductively constructs an execution that can determine the final output only after some unbounded time. It exploits an important continuity property of the output of an involution channel with respect to the presence/absence of glitches at the channel input, which is due to the involution property (unboundedness) of our delay functions.
5) In Section VIII, we briefly report on the results of the experimental evaluation [13] of the accuracy of the predictions of our involution model for a real circuit, which used both simulations and measurements.

Together, as we conclude in Section IX, the above results reveal that our binary circuit model in conjunction with involution channels indeed allows to solve SPF precisely when this is possible in physical circuits. Moreover, some (limited) experimental evaluation also revealed good accuracy. Consequently, to the best of our knowledge, our involution model seems to be the very first candidate for a model that indeed guarantees faithful glitch propagation.

D. Related Work

Whereas there is a wealth of research devoted to the analog modeling of digital circuits (see [4], [14]–[17] for a few references), none addressed the issue of characterizing delay functions with respect to solvability of problems. On the other hand, digital circuit models have been proposed as a general approach for modeling asynchronous sequential switching circuits long time ago: Unger [5] introduced the well-known pure and inertial delay channels, which have been heavily used both in research and in industrial timing simulators since then. Brzozowski and Ebergen [18] formally proved that it is impossible to implement Muller C-elements and other state-holding components using only zero-time logical gates interconnected by wires without timing restrictions. Bellido-Díaz *et al.* [6] proposed the PID model, and justified its appropriateness both analytically and by comparing model predictions against SPICE simulations. In [19], the PID model [later renamed to delay degradation model (DDM)] was generalized from inverters to NAND and NOR gates. Thanks to considerable efforts like [19], [20] spent on the question of how to extract the DDM model parameters from technology parameters, the DDM model has already made its way into digital timing analysis tools [7].

II. Analog Models Versus Involution Channels

Restricting delay functions to satisfy the involution property $-\delta_\uparrow(-\delta_\downarrow(T)) = -\delta_\downarrow(-\delta_\uparrow(T)) = T$ might raise concerns about whether such an assumption makes sense at all in real circuits, and whether/how it fits to existing analog models [4], [14]–[17]. In this section, we will show that involution channels are indeed well-suited for modeling physical circuits, in the sense that they arise naturally in a (generalized) standard analog model.

More specifically, we will show that, for any given involutions $\delta_\uparrow$, $\delta_\downarrow$, there is an analog channel model that has $\delta_\uparrow$, $\delta_\downarrow$ as its corresponding delay functions. It consists of a pure delay component, a slew-rate limiter with generalized switching waveforms, and an ideal comparator (see Fig. 3). Note carefully, though, that we do not claim that this is the only analog model that leads to involution delay functions. There may of course be many others as well. Vice versa, the fact that some well-known analog model leads to involutions does not at all make our results incremental: besides the fact that, to the best of our knowledge, no analog modeling paper [4], [14]–[17] addressed the properties of corresponding delay functions, it is of course not possible to generalize results obtained for some *particular* involution to involutions in general.

As a first observation, note that the timing behavior of involution channels is fully determined by either one of the delay functions, as $\delta_\uparrow(T) = -\delta_\downarrow^{-1}(-T)$ (and similarly for $\delta_\downarrow$). To better understand how our delay functions integrate the behavior of both transitions, consider the ansatz $\delta_\uparrow(T) = -f_\uparrow^{-1}(f_\downarrow(T))$ and $\delta_\downarrow(T) = -f_\downarrow^{-1}(f_\uparrow(T))$, where $f_\uparrow$ resp. $f_\downarrow$ are strictly increasing resp. decreasing functions. Intuitively, we would like $f_\uparrow$ and $f_\downarrow$ to represent the continuous switching waveforms of the output of the generalized slew rate limiter upon the occurrence of a rising, respectively, falling transition at its input. In the above formula, e.g., at a rising transition, $\delta_\uparrow(T)$ returns the time by which $f_\uparrow$ has to be shifted so that the output signal remains continuous with respect to the output caused by the previous falling transition. For realistic switching waveforms, we further need $f_\uparrow(0) = 1-f_\downarrow(0) = 0$ and $\lim_{t\to\infty} f_\uparrow(t) = 1-\lim_{t\to\infty} f_\downarrow(t) = 1$, which requires to augment our ansatz with some additive terms, resulting in

$$\delta_\uparrow(T) = -f_\uparrow^{-1}\left(f_\downarrow\left(T+\delta_\infty^\downarrow\right)\right) + \delta_\infty^\uparrow \quad \text{and}$$
$$\delta_\downarrow(T) = -f_\downarrow^{-1}\left(f_\uparrow\left(T+\delta_\infty^\uparrow\right)\right) + \delta_\infty^\downarrow \tag{1}$$

where $\delta_\infty^\uparrow = \lim_{T\to\infty}\delta_\uparrow(T)$ and $\delta_\infty^\downarrow = \lim_{T\to\infty}\delta_\downarrow(T)$.

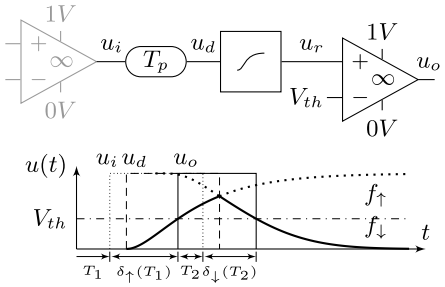

Fig. 3. Simple analog channel model.

Fig. 3 shows a block diagram of an idealized analog circuit corresponding to so constructed involution channels, and a sample waveform. The pure delay time-shifts the binary-valued input u_i by some T_p. The slew rate limiter exchanges the step functions of the resulting u_d with instances of $f_\uparrow$ and $f_\downarrow$, shifting them in time such that the output u_r is continuous and switches between strictly increasing and decreasing exactly at u_d's switching times. The comparator generates u_o by again discretizing the value of this waveform comparing it to the threshold voltage V_{th}, effectively adding $f_\uparrow^{-1}(V_{\text{th}})$ resp. $f_\downarrow^{-1}(V_{\text{th}})$ to the instantiation times of $f_\uparrow$ resp. $f_\downarrow$. The input–output delay of a perfectly idle channel (the last output transition was at time $-\infty$), i.e., $\delta_\infty^\uparrow$ and $\delta_\infty^\downarrow$ for rising, respectively, falling transitions, is the sum of the pure delay and the time the switching waveform needs to reach the threshold voltage V_{th}

$$\delta_\infty^\uparrow = T_p + f_\uparrow^{-1}(V_{\text{th}}) \quad \text{and} \quad \delta_\infty^\downarrow = T_p + f_\downarrow^{-1}(V_{\text{th}}). \tag{2}$$

This equation and (1) can be used to transform the parameters of the model in Fig. 3 to the corresponding δ functions. As a special case, consider a slew rate limiter implemented as a first-order *RC* low pass filter; the switching waveforms are $f_\downarrow(t) = 1 - f_\uparrow(t) = e^{-t/\tau}$ here, with τ being the *RC* time constant. Inserting these functions and their inverses

into (1) and (2), we obtain what we refer to as *exp-channels* in the remainder of this article

$$\delta_\uparrow(T) = \tau \ln\Big(1 - e^{-(T+T_p-\tau\ln(V_{\text{th}}))/\tau}\Big) + T_p - \tau\ln(1-V_{\text{th}})$$
$$\delta_\downarrow(T) = \tau \ln\Big(1 - e^{-(T+T_p-\tau\ln(1-V_{\text{th}}))/\tau}\Big) + T_p - \tau\ln(V_{\text{th}}). \tag{3}$$

Conversely to the above, given any $\delta_\downarrow$, $\delta_\uparrow$, there is a combination of switching waveforms $f_\uparrow$ and $f_\downarrow$, pure delay T_p, and threshold V_{th}, such that the circuit in Fig. 3 behaves exactly like the corresponding involution channel. For example, one could choose $f_\downarrow(t) = e^{-t}$, $f_\uparrow(t) = e^{\delta_\downarrow(t-\delta_\infty^\uparrow)-\delta_\infty^\downarrow}$, T_p such that $\delta_\downarrow(-T_p) = \delta_\uparrow(-T_p) = T_p$, and $V_{\text{th}} = e^{T_p-\delta_\infty^\downarrow}$. However, this choice is of course not unique.

III. Binary Circuit Model

We next formally define the binary-value continuous-time circuit model used in this article. Except for the involution channels introduced in Section IV, it is essentially the same as the model introduced in [10].

A. Signals

A *falling transition* at time t is the pair $(t, 0)$, a *rising transition* at time t is the pair $(t, 1)$. A *signal* is a (finite or infinite) list of alternating transitions such that the following.

- S1) The initial transition is at time $-\infty$; all other transitions are at times $t \geq 0$.
- S2) The sequence of transition times is strictly increasing.
- S3) If there are infinitely many transitions in the list, then the set of transition times is unbounded.

To every signal s corresponds a function $\mathbb{R} \to \{0, 1\}$ whose value at time t is that of the most recent transition. We follow the convention that the function already has the new value at the time of a transition, i.e., the function is constant in the half-open interval $[t_n, t_{n+1})$ if t_n and t_{n+1} are two consecutive transition times. A signal is uniquely determined by such a function.

B. Circuits

Circuits are obtained by interconnecting a set of input ports and a set of output ports, forming the external interface of a circuit, and a set of combinational gates via channels. We constrain the way components are interconnected in a natural way by requiring that any gate input, channel input, and output port is attached to only one input port, gate output or channel output.

Formally, a *circuit* is described by a directed graph where:

- C1) vertices are partitioned into *input ports*, *output ports*, and *gates*;
- C2) channels are edges with a channel function that maps an input signal to an output signal. Multiple edges between two vertices are allowed. Section IV specifies the properties of the channel function for our involution channels. For simplicity of analysis, we use 0-delay channels as edges from input ports and to output ports;
- C3) input ports have no incoming channels;
- C4) output ports have exactly one incoming channel and no outgoing channel;
- C5) every gate is assigned a Boolean gate function $\{0, 1\}^d \to \{0, 1\}$, where d is the number of incoming channels, and an initial value in $\{0, 1\}$;
- C6) there is a fixed order on the incoming channels of every gate.

C. Executions

An *execution* of circuit $\mathcal{C}$ is a collection of signals s_Γ for all components Γ (vertices and channels) of $\mathcal{C}$ that respects the channel functions, Boolean gate functions, and initial values. Formally, the following properties hold.

- E1) If I is an input port, then there are no restrictions on s_I.
- E2) If O is an output port, then $s_O = s_C$, where C is the unique incoming channel of O.
- E3) If C is a channel departing from vertex V, then $s_C = f_C(s_V)$, where f_C is the channel's function.
- E4) If B is a gate with d incoming channels $C_1, \ldots, C_d$, ordered according to the fixed order of condition (C6), gate function f_B, and initial value I_B, then for all times $t < 0$, $s_B(t) = I_B$, and for $t \geq 0$, $s_B(t) = f_B(s_{C_1}(t), s_{C_2}(t), \ldots, s_{C_d}(t))$.

D. Short-Pulse Filtration

A *pulse* of length $\Delta > 0$ at time $T \geq 0$ has initial value 0, one rising transition at time T, and one falling transition at time $T + \Delta$. A signal *contains a pulse* of length Δ at time T if it contains a rising transition at time T, a falling transition at time $T + \Delta$ and no transition in between. The *zero signal* has the initial transition $(-\infty, 0)$ only.

A circuit *solves SPF*, if it fulfills the following conditions.

- F1) The circuit has exactly one input port and exactly one output port. *(Well-formedness.)*
- F2) If the input signal is the zero signal, then so is the output signal. *(No generation.)*
- F3) There exists an input pulse such that the output signal is not the zero signal. *(Nontriviality.)*
- F4) There exists an $\varepsilon > 0$ such that for every input pulse the output signal never contains a pulse of length less than or equal to ε. *(No short pulses.)*

Note that we allow the circuit to behave arbitrarily if the input signal is not a single pulse or the zero signal.

A circuit *solves bounded SPF* if additionally the following condition holds.

- F5) There exists a $K > 0$ such that for every input pulse the last output transition is before time $T+K$, where T is the time of the last input transition. *(Bounded stabilization time.)*

IV. Involution Channels

Intuitively, a channel propagates each transition of the input signal to a transition at the output happening after some *input-to-output* delay $\delta(T)$, which depends on the *previous-output-to-input* delay T. Note that T can be negative if two input transitions are close together, as in Fig. 2 (right).

Formally, an *involution channel* is characterized by two strictly increasing concave *delay functions* $\delta_\uparrow : (-\delta_\infty^\downarrow, \infty) \to (-\infty, \delta_\infty^\uparrow)$ and $\delta_\downarrow : (-\delta_\infty^\uparrow, \infty) \to (-\infty, \delta_\infty^\downarrow)$ such that both $\delta_\infty^\uparrow = \lim_{T\to\infty} \delta_\uparrow(T)$ and $\delta_\infty^\downarrow = \lim_{T\to\infty} \delta_\downarrow(T)$ are finite and

$$-\delta_\uparrow\big(-\delta_\downarrow(T)\big) = T \text{ and } -\delta_\downarrow\big(-\delta_\uparrow(T)\big) = T \tag{4}$$

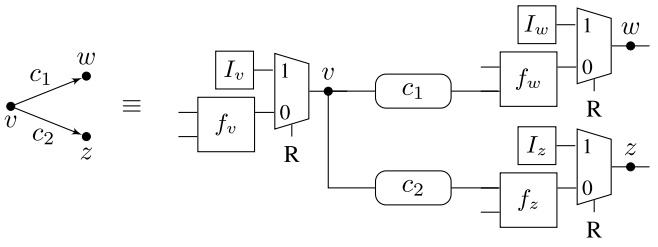

Fig. 4. Circuit (graph) with gates v, w, z, and channels c_1 and c_2 (on the left) and the physical equivalent (on the right). Reset R switches from 1 to 0 at time 0.

Algorithm 1 Channel Algorithm, Up to Time τ

$x \leftarrow$ value at $t = -\infty$ in Input
add $(-\infty, x)$ to Output
Prev $\leftarrow (-\infty, x)$
$t \leftarrow$ earliest time of a pending transition, otherwise $+\infty$
while $t \le \tau$ **do**
  move (t, x) from Pending to Output
  $(t', x') \leftarrow$ Prev
  if $x = 1$ **then** $\delta \leftarrow \delta_\uparrow(t - t')$ **else** $\delta \leftarrow \delta_\downarrow(t - t')$ **endif**
  Prev $\leftarrow (t + \delta, x)$
  if $t + \delta \le t'$ **then**
    remove (t', x') from Pending(C)
  else
    add $(t + \delta, x)$ to Pending(C)
  end if
  $t \leftarrow$ earliest time of a pending transition, otherwise $+\infty$
end while
return Output

for all applicable T. All such functions are necessarily continuous and, for simplicity, we will also assume them to be differentiable; δ being concave thus implies that its derivative δ' is continuous and monotonically decreasing.

The behavior of an involution channel is defined by Algorithm 1, which maps channel input signal s with event list Input to channel output signal $f_C(s)$ with event list output.

Definition 1: An involution channel is *strictly causal* if $\delta_\uparrow(0) > 0$, which is equivalent to the condition $\delta_\downarrow(0) > 0$ due to (4) and the functions being strictly increasing.

Lemma 1: An exp-channel is strictly causal if and only if $T_p > 0$.

The next lemma identifies an important parameter $\delta_{\min}$ of a strictly causal involution channel.

Lemma 2: A strictly causal involution channel has a unique $\delta_{\min}$ defined by $\delta_\uparrow(-\delta_{\min}) = \delta_{\min} = \delta_\downarrow(-\delta_{\min})$, which is positive. For exp-channels, $\delta_{\min} = T_p$.

For the derivative, we have $\delta'_\uparrow(-\delta_\downarrow(T)) = 1/\delta'_\downarrow(T)$ and hence $\delta'_\uparrow(-\delta_{\min}) = 1/\delta'_\downarrow(-\delta_{\min})$.

Proof: Set $f(T) = -T + \delta_\uparrow(-T)$. This function is continuous and strictly decreasing, since $\delta_\uparrow$ is continuous and strictly increasing. Because $f(0) = \delta_\uparrow(0)$ is positive and the limit of $f(T)$ as $T \to \delta^\downarrow_\infty$ is $-\infty$, there exists a unique $\delta_{\min}$ between 0 and $\delta^\downarrow_\infty$ for which $f(\delta_{\min}) = 0$. Hence, $\delta_\uparrow(-\delta_{\min}) = \delta_{\min}$. The second equality follows from $\delta_{\min} = \delta_\downarrow(-\delta_\uparrow(-\delta_{\min})) = \delta_\downarrow(-\delta_{\min})$ according to (4).

The second part of the lemma follows by differentiating (4). ∎

We next show that $\delta_{\min}$ indeed deserves its name: a particular consequence of the following lemma is that the channel delay for any noncanceled transition is larger than $\delta_{\min}$.

Lemma 3: Let t_n and t_{n+1} be the times of the nth and $(n+1)$th input transitions. The following are equivalent.

1) The nth and $(n+1)$th pending output transitions cancel.
2) $t_{n+1} \le t_n + \delta_n - \delta_{\min}$.
3) $\delta_{n+1} \le \delta_{\min}$.

Proof: Let δ be either $\delta_\uparrow$ or $\delta_\downarrow$, depending on whether t_{n+1} is a rising or falling transition. By definition, the two transitions cancel if and only if

$$\delta_{n+1} = \delta(t_{n+1} - t_n - \delta_n) \le -(t_{n+1} - t_n - \delta_n). \tag{5}$$

Set $T = t_{n+1} - t_n - \delta_n$. By Lemma 2, equality holds in (5) if and only if $T = -\delta_{\min}$. Because the left-hand side of (5) is increasing in T and the right-hand side is strictly decreasing in T, (5) is equivalent to $T \le -\delta_{\min}$. This in turn is equivalent to $t_{n+1} \le t_n + \delta_n - \delta_{\min}$ and $\delta(T) \le \delta_{\min}$. ∎

In the rest of this article, we assume all channels to be strictly causal involution channels unless noted otherwise.

V. Simulating Executions of Circuits

The definition of an execution of a circuit as given in Section III is "existential," in the sense that it only allows to check for a given collection of signals whether it is an execution or not. This also includes the involution channel algorithm: it specifies the channel output signal, given a *fixed* input signal. *A priori*, this does not give an algorithm to construct executions of circuits with feedback loops. However, we show in Theorem 1 that executions always exist and are unique for circuits with strictly causal involution channels. We can therefore give a *deterministic simulation algorithm* for arbitrary circuits (that may include feedback loops).

The simulation algorithm takes as input a time $\tau \ge 0$ up to which the circuit should be simulated, and a list of transitions $\text{Fixed}(I) = s_I$ for every input port I. We denote by $\text{Init}(I)$ the value of the initial transition in this list (at time $-\infty$). When the algorithm terminates, it outputs a list of transitions $\text{Fixed}(\Gamma)$ up to time τ for every component Γ (vertices and channels) of the circuit.

During the execution of the algorithm, it distinguishes *pending* and *fixed* transitions. Pending transitions are stored in the variable $\text{Pending}(\Gamma)$, while fixed ones are transferred to $\text{Fixed}(\Gamma)$; they are said to be *marked fixed* when doing so. We will show (Lemma 6) that pending transitions can still be canceled by other transitions. Fixed transitions on the other hand are guaranteed to occur in the constructed execution.

For a channel C, we write $\text{Incoming}(C)$ for its predecessor and $\text{Delay}(C)$ for the pair of its delay functions. Furthermore, we store its last generated output transition (ordered by the time of the corresponding input transitions), whether it is canceled or not, in the variable $\text{Prev}(C)$. For a gate B, we write $\text{Incoming}(B)$ for the collection of its incoming channels, $\text{Init}(B)$ for its initial value, and f_B for its Boolean function. For an output port O, $\text{Incoming}(O)$ is its unique incoming channel. For an input port I, $\text{Outgoing}(I)$ is its unique outgoing channel.

The simulation algorithm is given in Algorithm 2. It uses the function $\text{Latest}(\Gamma, t)$ for a component Γ and a time t, which is equal to the Boolean value of the most recent fixed or pending transition for component Γ before or at time t (ordered by their

Algorithm 2 Circuit Simulation Algorithm, Until Time τ

```
1: for all input ports I do
2:    C ← Outgoing(I)
3:    copy all finite-time transitions from Fixed(I) to Pending(C)
4: end for
5: for all channels C to gate B do
6:    V ← Incoming(C)
7:    add (−∞, Init(V)) to Fixed(C)
8:    Prev(C) ← (−∞, Init(V))
9: end for
10: for all gates B do
11:    add (−∞, Init(B)) to Fixed(B)
12: end for
13: t ← 0
14: while t ≤ τ do
15:    for all comp. Γ with a pending transition (t, x) at time t do
16:       move (t, x) from Pending(Γ) to Fixed(Γ)
17:    end for
18:    for all gates B do
19:       (C_1, ..., C_d) ← Incoming(B)
20:       v ← f_B(Latest(C_1, t), ..., Latest(C_d, t))
21:       if v ≠ Latest(B, t) then add (t, v) to Fixed(B) endif
22:    end for
23:    for all channels C from gate B_1 to gate B_2 do
24:       (δ_↑, δ_↓) ← Delay(C)
25:       if ∃ a transition (t, x) in Fixed(B_1) at time t then
26:          (t', x') ← Prev(C)
27:          if x = 1 then δ ← δ_↑(t − t') else δ ← δ_↓(t − t') endif
28:          Prev(C) ← (t + δ, x)
29:          if t + δ ≤ t' then
30:             remove (t', x') from Pending(C)
31:          else
32:             add (t + δ, x) to Pending(C)
33:          end if
34:       end if
35:    end for
36:    t ← earliest time of a pending transition, otherwise +∞
37: end while
38: for all output ports O do
39:    C ← Incoming(O); V ← Incoming(C)
40:    copy Fixed(V) to Fixed(C) and to Fixed(O)
41: end for
```

transition times). Note the special handling of 0-delay channels from input ports and to output ports, which is outside of the main loop and just copies the transition lists.

If the Init(·)-values of the incoming neighbors are not compatible with a gate's own initial value, then line 21 generates a transition at time $t = 0$ in the first loop iteration. Thereupon, the algorithm iteratively looks at the earliest pending transitions, declaring them as fixed, and propagating their effect through gates and channels. We highlight two noteworthy properties of the algorithm: 1) the delay $\delta(T)$ is a function of the previous-output-to-input delay $T = t - t'$ (see line 27) and 2) a pending output transition of a channel is removed if a later input transition causes an output transition that occurs earlier (code line 29). In this case, the two transitions cancel at the channel output (pulse cancellation).

We now show that this algorithm indeed constructs an execution of a circuit $\mathcal{C}$ (up to time τ). Let t_ℓ be the value of t at the beginning of iteration $\ell \geq 1$ of the algorithm (set in line 13 or 36). Denote by $\delta^{\mathcal{C}}_{\min} > 0$ the minimal $\delta_{\min}$ of all channels in circuit $\mathcal{C}$ except for the 0-delay channels from inputs and to outputs.

Lemma 4: For all iterations $\ell \geq 1$: 1) no transition (s, x) with $s \neq t_\ell$ is newly marked fixed in the iteration; 2) a transition (s, x) added during iteration ℓ either has time $s = t_\ell$ or $s > t_\ell + \delta^{\mathcal{C}}_{\min}$; and 3) every transition at time t_ℓ is fixed at the end of the iteration.

Proof: Statement 1) is implied by the fact that transitions are only marked fixed in lines 16 and 21, which act on transitions at time t_ℓ only.

For 2), assume by contradiction that a transition (s, x) with $s \leq t_\ell + \delta^{\mathcal{C}}_{\min}$ but different from t_ℓ was added in iteration ℓ. Such a transition can only be added via line 32. By our assumption and line 27, $\delta(t_\ell - t') \leq \delta^{\mathcal{C}}_{\min} \leq \delta_{\min}$ must have held, where $\delta \in \{\delta_\uparrow, \delta_\downarrow\}$ is the applicable channel function with minimal delay $\delta_{\min}$ and t' is the time of the channel's last output transition. However, by Lemma 3, this requires $t_\ell + \delta(t_\ell - t') \leq t_\ell + \delta_{\min} \leq t'$ which is in contradiction with the (negated) condition in line 29 necessary to reach line 32.

For 3), assume by contradiction that, at the end of iteration ℓ, there exists a nonfixed transition (t_ℓ, x). Since line 16 marks all transitions at time t_ℓ fixed and line 21 adds only fixed transitions at time t_ℓ, the nonfixed transition must have been newly added by line 32. However, as in 2), we know that this requires $\delta(t_\ell - t') \leq \delta_{\min}$, again contradicting line 29. ∎

From an inductive application of Lemma 4, we obtain that the sequence of iteration start times $(t_\ell)_{\ell \geq 1}$ is strictly increasing without bound.

Lemma 5: Either the number of transitions generated by the simulation algorithm is finite or for all non-negative integers $k \geq 0$, there exists some iteration $\ell \geq 1$ such that $t_\ell \geq k \cdot \delta^{\mathcal{C}}_{\min}$.

Proof: We prove the lemma by induction. The base case $k = 0$ is trivial since $t_\ell \geq 0$ for all iterations ℓ.

For the induction step, let $t_\ell \geq k \cdot \delta^{\mathcal{C}}_{\min}$. Let N be the number of distinct times of pending transitions in the interval $[t_\ell, (k+1) \cdot \delta^{\mathcal{C}}_{\min})$ at the start of iteration ℓ. By Lemma 4, all pending transitions added in iterations from ℓ onward are at times $s > t_\ell + \delta^{\mathcal{C}}_{\min} \geq (k+1) \cdot \delta^{\mathcal{C}}_{\min}$. This means that all pending transitions before time $(k+1) \cdot \delta^{\mathcal{C}}_{\min}$ are either removed or fixed at the end of iteration $\ell + N - 1$. Hence, $t_{\ell+N} \geq (k+1) \cdot \delta^{\mathcal{C}}_{\min}$, which concludes the proof. ∎

The following lemma proves that the generated transition lists are well-defined, in the sense that no later iteration can remove transitions that may have generated causally dependent other transitions already.

Lemma 6: Consider Algorithm 2 with line 30 changed to remove from set Pending$(C) \cup$ Fixed(C), i.e., both pending and fixed transitions. Then, no fixed transition would ever be canceled.

Proof: Assume by contradiction that some iteration $\ell \geq 1$ is the first in which a fixed transition is canceled. Thus, there exists a transition at time t_ℓ that generated a new transition at some time $t = t_\ell + \delta(t_\ell - t')$ that results in the cancellation of a fixed transition at time t', i.e., $t \leq t'$. From the fact that the transition at time t' is already fixed at iteration ℓ, Lemma 4 1), and the fact that the sequence of t_ℓ, $\ell \geq 1$, generated by the modified algorithm is increasing, we obtain $t' \leq t_\ell$.

However, the condition in line 29 requires $t_\ell + \delta(t_\ell - t') \leq t'$. Since $t_\ell - t' \geq 0$ and the channel is strictly causal (see Definition 1), $\delta(t_\ell - t') > 0$ yields a contradiction. ∎

This allows us to use the original line 30.

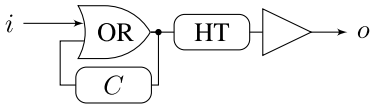

Fig. 5. Circuit solving unbounded SPF, consisting of an OR-gate fed-back by channel C, and an exp-channel HT implementing a high-threshold buffer.

We are now ready for the main result of this section, which asserts the existence of a unique execution of our circuit $\mathcal{C}$.

Theorem 1: For any $0 \leq \tau < \infty$, the execution construction algorithm applied to circuits with strictly causal involution channels always terminates. At the end of iteration $\ell \geq 1$, the collection of signals s_σ, restricted to time $[-\infty, t_\ell]$, is the unique execution of circuit $\mathcal{C}$ restricted to time $[-\infty, t_\ell]$. If the algorithm terminates at the beginning of iteration ℓ, then this collection of signals is the unique execution of circuit $\mathcal{C}$.

Proof: From Lemma 5, we deduce that there is an iteration $\ell \geq 1$ such that $t_\ell > \tau$, thus the algorithm terminates. From Lemma 4, we know that the algorithm does not add transitions with times $< t_\ell$ during iteration ℓ. By Lemma 6 in conjunction with the fact that lines 23–35 implement the output transition generation algorithm of Section IV, Algorithm 2 correctly computes channel outputs. To prove uniqueness of the execution, assume by contradiction that there is a second execution and consider the first differing transition. This contradicts either E3) or E4). ∎

VI. Possibility of Unbounded SPF

In this section, we show that unbounded SPF is solvable in our circuit model with strictly causal involution channels. We do this by verifying that the circuit shown in Fig. 5 indeed solves SPF. The circuit was inspired by the physical solution that provided Fig. 1, which consists of a fed-back OR-gate forming the storage loop and a subsequent high-threshold buffer. The high-threshold buffer is implemented by a (nonsymmetric) exp-channel, with appropriately chosen parameters. Unless otherwise noted, $\delta_\infty^\uparrow$, $\delta_\infty^\downarrow$, and $\delta_{\min}$ will refer to the parameters of the feedback channel.

We consider a pulse of length $\Delta > 0$ at time 0 at the input and reason about the behavior of the feed-back loop. Then, we show that its behavior can be translated to a legitimate SPF output by using a high-threshold buffer. We start by identifying two extremal cases: if Δ is too small, then the pulse is filtered by the channel in the feed-back loop. If it is too large, the pulse is captured by the storage loop, leading to a stable output 1.

Lemma 7: If the input pulse's length Δ satisfies $\Delta \geq \delta_\infty^\uparrow$, then the OR output has a unique rising transition at time 0.

Proof: Assigning the channel output s_C a single rising transition at time $\delta_\infty^\uparrow$ is part of a consistent execution, in which the OR's output has a single rising transition at time 0. The lemma now follows from uniqueness of executions. ∎

Lemma 8: If the input pulse's length Δ satisfies $\Delta \leq \delta_\infty^\uparrow - \delta_{\min}$, then the OR output contains only the input pulse.

Proof: Channel C's input signal contains only two transitions: one at time $t_1 = 0$ and one at time $t_2 = \Delta \leq \delta_\infty^\uparrow - \delta_{\min}$. Since $\delta_1 = \delta_\infty^\uparrow$ and hence $t_2 \leq t_1 + \delta_1 - \delta_{\min}$, the two pending transitions of C's output cancel by Lemma 3, and no further transitions are generated at the OR gate's output. ∎

Now suppose that the input pulse length satisfies $\delta_\infty^\uparrow - \delta_{\min} < \Delta_0 < \delta_\infty^\uparrow$. For these pulse lengths Δ_0, the OR output signal will contain the input pulse Δ_0, followed by a series of pulses of lengths $\Delta_1, \Delta_2, \ldots$ For all but one Δ_0, this series will turn out to be either decreasing or increasing and finite, causing the output signal to be eventually 0 or eventually 1. To compute these pulse lengths, we define the auxiliary function

$$f(\Delta) = \delta_\downarrow\big(\Delta - \delta_\uparrow(-\Delta)\big) + \Delta - \delta_\uparrow(-\Delta) \tag{6}$$

which gives $\Delta_n = f(\Delta_{n-1})$ for all $n \geq 2$. To see this, note that Δ_{n-1} at the channel input is also present at the channel output, so the rising resp. falling transition is delayed by $\delta_\uparrow(-\Delta_{n-1})$ resp. $\delta_\downarrow(\Delta_{n-1} - \delta_\uparrow(-\Delta_{n-1}))$. The first generated pulse starts from a zero channel input and thus

$$\Delta_1 = \Delta_0 - \delta_\infty^\uparrow + \delta_\downarrow\Big(\Delta_0 - \delta_\infty^\uparrow\Big). \tag{7}$$

The procedure stops if either $f(\Delta_n) \leq 0$ (pulse canceled; the output is constant 0 thereafter) or if

$$f(\Delta_n) \geq \delta_{\min} > 0 \tag{8}$$

(pulse captured; the output is constant 1 thereafter).

The only case in which the procedure does not stop is if $f(\Delta_1) = \Delta_1$. There is a unique $\Delta_1 > 0$ with this property, denoted $\tilde{\Delta}_1$: with $\kappa = \delta_\uparrow(-\tilde{\Delta}_1)$ and hence $\tilde{\Delta}_1 = \delta_\downarrow(-\kappa)$ by the involution property, (6) reads $\delta_\downarrow(\tilde{\Delta}_1 - \kappa) = \kappa$ and hence $\tilde{\Delta}_1 - \kappa = -\delta_\uparrow(-\kappa)$. Note carefully that κ is the period of the resulting periodic signal, and $\gamma = \tilde{\Delta}_1/\kappa < 1$ its duty cycle (0.5 in the case of symmetric channels, as $\kappa = 2\tilde{\Delta}_1$ here). Since the left-hand side of the resulting equation $\delta_\downarrow(-\kappa) + \delta_\uparrow(-\kappa) - \kappa = 0$ is positive for $\kappa \to 0$ but negative for $\kappa \to \min\{\delta_\infty^\uparrow, \delta_\infty^\downarrow\}$, there is indeed a unique $\kappa > 0$ and a corresponding $\tilde{\Delta}_1 = \delta_\downarrow(-\kappa) < \kappa$.

Given the upper bound on Δ_0 from Lemma 7 and (7), we must have $\tilde{\Delta}_1 < \delta_\downarrow(0)$. Since $\Delta_1 \to \delta_\downarrow(0)$ as $\Delta_0 \to \delta_\infty^\uparrow$ and $\Delta_1 \to 0$ as $\Delta_0 \to \delta_\infty^\uparrow - \delta_{\min}$, there exists a unique Δ_0 such that $\Delta_1 = \tilde{\Delta}_1$. Denote it by $\tilde{\Delta}_0$.

The following lemma shows that the procedure indeed stops if and only if $\Delta_1 \neq \tilde{\Delta}_1$, and can be used to bound the number of steps until it stops.

Lemma 9: For $f(.)$ given in (6) with fixed point $\tilde{\Delta}_1$, we have $|f(\Delta_n) - \tilde{\Delta}_1| \geq (1 + \delta'_\uparrow(0)) \cdot |\Delta_n - \tilde{\Delta}_1|$ for all $n \geq 1$ if $\Delta_n > 0$.

Proof: Differentiation of (6) provides $f'(\Delta_n) = (1 + \delta'_\uparrow(-\Delta_n)) \cdot \delta'_\downarrow(\Delta_n - \delta_\uparrow(-\Delta_n)) + 1 + \delta'_\uparrow(-\Delta_n) \geq 1 + \delta'_\uparrow(0)$, because $\delta'_\uparrow(-\Delta_n) \geq \delta'(0)$ and $\delta'_\downarrow(T) > 0$ for all T as $\delta_\downarrow(.)$ is concave and increasing. The mean value theorem of calculus now implies the lemma. ∎

Theorem 2: The fed-back OR gate with a strictly causal involution channel has the following output when the input pulse has length Δ_0.

1) If $\Delta_0 > \tilde{\Delta}_0$, then the output is eventually constant 1.
2) If $\Delta_0 < \tilde{\Delta}_0$, then the output is eventually constant 0.
3) If $\Delta_0 = \tilde{\Delta}_0$, then the output after the initial pulse Δ_0 is a periodic pulse train with uptime $\tilde{\Delta}_1$, period κ and duty cycle $\gamma = \tilde{\Delta}_1/\kappa < 1$.

Furthermore, the stabilization time in the first two cases is in the order of $\log_a(1/|\Delta_0 - \tilde{\Delta}_0|)$ with $a = 1 + \delta'_\uparrow(0)$.

Proof: If $\Delta_0 \geq \delta_\infty^\uparrow$ or $\Delta_0 \leq \delta_\infty^\uparrow - \delta_{\min}$, then Lemmas 7 and 8 show the theorem.

So let $\Delta_0 \in (\delta_\infty^\uparrow - \delta_{\min}, \delta_\infty^\uparrow)$. By Lemma 9, the number of generated pulses until the procedure stops is in the order of

$\log_a(1/|\Delta_1 - \tilde{\Delta}_1|)$. Setting $g(\Delta_0) = \Delta_0 - \delta_\infty^\uparrow + \delta_\downarrow(\Delta_0 - \delta_\infty^\uparrow)$ such that $\Delta_1 = g(\Delta_0)$, cp. (7), and applying the mean value theorem of calculus to this function, we see analogously as in the proof of Lemma 9 that

$$|\Delta_1 - \tilde{\Delta}_1| \geq \Big(1 + \delta_\downarrow'(0)\Big) \cdot |\Delta_0 - \tilde{\Delta}_0|.$$

Hence, the number of generated pulses is also in the order of $\log_a(1/|\Delta_0 - \tilde{\Delta}_0|)$. Since the period of the generated pulses is trivially upper-bounded by $\delta_\infty^\uparrow$, we have the same asymptotic bound on the stabilization time. ∎

Finally, one can show that a high-threshold buffer with arbitrary threshold can be modeled by an exp-channel with properly chosen V_{th}.

Lemma 10: Let C be an exp-channel with threshold V_{th} and initial value 0, and let $0 \leq \Gamma < V_{\text{th}}$. Then there exists some $\Theta > 0$ such that every finite or infinite pulse train with pulse lengths $\Theta_n \leq \Theta$, $n \geq 0$, and duty cycles $\Gamma_n \leq \Gamma$, $n \geq 1$, is mapped to the zero signal by C.

Proof: Denote channel C's time constant by τ, its pure delay by T_p, and its delay functions by $\delta_\uparrow$ and $\delta_\downarrow$ according to (3). Recall that $\delta_{\min} = T_p$ for C.

Regarding the initial pulse Θ_0, we observe that it is canceled by C whenever

$$\Theta_0 \leq \delta_\uparrow(\infty) - \delta_{\min} = -\tau \ln(1 - V_{\text{th}}) \tag{9}$$

recall (3): since the rising resp. falling transition has delay $\delta_\uparrow(\infty)$ resp. $\delta_\downarrow(\Theta_0 - \delta_\uparrow(\infty))$, we find $\Theta_0 + \delta_\downarrow(\Theta_0 - \delta_\uparrow(\infty)) - \delta_\uparrow(\infty) \leq -\delta_{\min} + \delta_\downarrow(-\delta_{\min}) = 0$.

For choosing an appropriate Θ that also causes a cancellation of Θ_n, $n \geq 1$, we define the function

$$\begin{aligned} f(\theta) &= \delta_\uparrow\left(\frac{1-\Gamma}{\Gamma}\theta - T_p\right) \\ &= \tau \ln\left(1 - V_{\text{th}} e^{-\frac{1-\Gamma}{\Gamma}\cdot\frac{\theta}{\tau}}\right) + T_p - \tau \ln(1 - V_{\text{th}}). \end{aligned} \tag{10}$$

By Lemma 2, we have $f(0) = T_p$. Differentiating this function at $\theta = 0$ gives

$$f'(0) = \frac{1-\Gamma}{\Gamma} \cdot \frac{V_{\text{th}}}{1 - V_{\text{th}}} > 1$$

because $\Gamma < V_{\text{th}}$. This, by continuity of f' and $f'(\theta)$ being decreasing, shows that there is some $\Theta_f > 0$ such that $f'(\theta) \geq 1$ for all $\theta \in [0, \Theta_f]$. Choosing $\Theta = \min\{\Theta_f, -\tau \ln(1 - V_{\text{th}})/2\}$, the mean value theorem of calculus applied to (10) provides

$$\forall \theta \in [0, \Theta]\colon (\theta) \geq T_p + \theta \tag{11}$$

and trivially

$$\Theta \leq \frac{-\tau \ln(1 - V_{\text{th}})}{2} = \frac{\delta_\uparrow(\infty) - \delta_{\min}}{2}. \tag{12}$$

We now prove that this choice of Θ satisfies the statement of our lemma. We know this already for the initial pulse Θ_0, for the remaining pulses Θ_n, $n \geq 1$, let $t_1, t_2, \ldots$ be the transition times in the input pulse train with pulse lengths $\Theta_n \leq \Theta$ and duty cycles $\Gamma_n \leq \Gamma$, i.e., $t_{2n} = t_{2n-1} + \Theta_n$ and $t_{2n+1} = t_{2n} + (1 - \Gamma_n/\Gamma_n)\Theta_n$. Note that indeed the period length of the nth pulse is $\Theta_n + (1 - \Gamma_n/\Gamma_n)\Theta_n = \Theta_n/\Gamma_n$ and the pulse length is Θ_n, which makes its duty cycle $\Theta_n/(\Theta_n/\Gamma_n) = \Gamma_n$.

We prove by induction that

$$\forall n > 0\colon \delta_{2n-1} \geq T_p + \Theta_n \tag{13}$$

where δ_{2n-1} is the delay of the $(2n-1)$st transition. From this, it then follows that all pulses get canceled by Lemma 3 using the equality $\delta_{\min} = T_p$ of Lemma 2 ($\delta_{\min}$ being that of the exp-channel C), because then $\Theta_n \leq \Theta$ together with (13) implies $t_{2n} = t_{2n-1} + \Theta_n \leq t_{2n-1} + \delta_{2n-1} - T_p$. Thus, it only remains to prove (13).

For the base case $n = 1$, we note that the duty cycle of the initial pulse may be arbitrary, hence $t_1 - (t_0 + \Theta_0)$ may be arbitrarily small. However, we find

$$\begin{aligned} \delta_1 &= \delta_\uparrow\big(t_1 - t_0 - \Theta_0 - \delta_\downarrow(\Theta_0 - \delta_\uparrow(\infty))\big) \\ &\geq \delta_\uparrow\big(-\delta_\downarrow(\Theta_0 - \delta_\uparrow(\infty))\big) = \delta_\uparrow(\infty) - \Theta_0 \\ &\geq (\delta_\uparrow(\infty) + T_p)/2 = \Theta + T_p \geq \Theta_1 + T_p \end{aligned}$$

where we used the involution property and (12).

For the induction step, we observe $\delta_{2n} \leq T_p$ by Lemma 3 and the induction hypothesis and thus

$$\delta_{2n+1} = \delta_\uparrow\left(\frac{1-\Gamma_n}{\Gamma_n}\Theta_n - \delta_{2n}\right) \geq f(\Theta_n)$$

where we used $\Gamma_n \leq \Gamma$. By (10), we have

$$\delta_{2n+1} \geq f(\Theta_n) \geq T_p + \Theta_n \tag{14}$$

since $\Theta_n \leq \Theta$, which concludes the proof. ∎

By letting the time constant τ grow, one can hence achieve the following result.

Lemma 11: Let $\Theta > 0$ and $0 \leq \Gamma < 1$. Then, there exists an exp-channel C such that every finite or infinite pulse train with pulse lengths $\Theta_n \leq \Theta$, $n \geq 0$, and duty cycles $\Gamma_n \leq \Gamma$, $n \geq 1$, is mapped to the zero signal by C.

Proof: We reuse the notation of the proof of Lemma 10.

Choose V_{th} such that $\Gamma < V_{\text{th}} < 1$. We show that there exists a $\tau > 0$ for which (10) and (12) hold. After having shown this, we are done since then the exp-channel C with initial value 0, voltage threshold V_{th}, and time constant τ satisfies the property of the lemma's statement.

We start by determining those θ for which $f'(\theta) \geq 1$. A straightforward calculation reveals $f''(\theta) < 0$ for all θ, i.e., f' is decreasing. Furthermore, solving the equation

$$f'(\theta) = \frac{1-\Gamma}{\Gamma} \cdot \frac{V_{\text{th}} e^{-\frac{1-\Gamma}{\Gamma}\frac{\theta}{\tau}}}{1 - V_{\text{th}} e^{-\frac{1-\Gamma}{\Gamma}\frac{\theta}{\tau}}} = 1 \tag{15}$$

gives the unique solution

$$\Theta_f = \tau \cdot \frac{\Gamma}{1-\Gamma} \cdot \ln \frac{V_{\text{th}}}{\Gamma}.$$

Hence, $f'(\theta) \geq 1$ for all $\theta \in [0, \Theta_f]$. Because Θ_f as well as $-\tau \ln(1 - V_{\text{th}})/2$ tend to infinity as $\tau \to \infty$, there exists a $\tau > 0$ that satisfies (10) and (12). ∎

By choosing $\Gamma = \gamma(1+\varepsilon) < 1$ according to Theorem 2 for some $\varepsilon > 0$ sufficiently small and Θ sufficiently large, SPF input pulses with duration $\Delta_0 \leq \tilde{\Delta}_0$ are mapped to a constant zero-output of the exp-channel according to Lemma 11. Let T be the time when pulse Δ_n, $n \geq 1$, of the feed-back loop with duty cycle $\geq \gamma(1+\varepsilon)$ has started. When choosing Θ so large that the feed-back loop in Fig. 5 has already locked to constant 1 at time $T + \Theta$, SPF input pulses with duration

$\Delta_0 > \tilde{\Delta}_0$ lead to a single up-transition (occurring only after $T + \Theta$) at the output. We can in fact establish.

Theorem 3: There is a circuit that solves unbounded SPF.

Proof: Since $\Delta_0 \leq \tilde{\Delta}_0$ leads to a pulse train with duty cycle at most $\gamma < \Gamma$ by Theorem 2, Lemma 11 guarantees a zero-output, provided Θ is chosen larger than $\tilde{\Delta}_1$.

For $\Delta_0 \geq \delta_\infty$, Lemma 7 trivially guarantees a single rising transition at the output. For values Δ_0 satisfying $\tilde{\Delta}_0 < \Delta_0 < \delta_\infty$, there is some time T where a 1-pulse Θ_n starts at the input of the exp-channel that will (along with its subsequent 0) have a duty cycle $\Gamma_n \geq \Gamma > \gamma$. By choosing Θ so large that, by time $T+\Theta$, the last input transition (to 1) has already occurred, Lemma 11 in conjunction with Lemma 12 below not only guarantees that all pulses occurring before T cancel, but also the ones that occur before time $T + \Theta$. After all, even a single, long pulse $\Theta_n = \Theta$ would still be canceled. Therefore, since the input of the exp-channel is already stable at 1 at time $T + \Theta$, only the final rising transition will eventually appear at the output. ■

VII. Impossibility of Bounded SPF

A. Continuity of Channels

In this section, we prove that strictly causal channels are continuous in a certain sense that we will define precisely.

To compare signals, we write $s_1 \leq s_2$ if s_2 is 1, whenever s_1 is and denote with $|s_1 - s_2|$ the signal that is 1, whenever the value of s_1 does not equal that of s_2. Consider signals s_1 and s_2 with $s_1 \leq s_2$ supplied to a channel C. A simple induction on the input transitions of a signal s_1 is already sufficient to show that $f_C(s_1) \leq f_C(s_2)$ as well: due to monotonicity of $\delta_\downarrow, \delta_\uparrow$, the occurrence time of every rising (resp. falling) transition can only decrease (resp. increase) when replacing an input transition at s_1 by its earlier (resp. later) matching transition at s_2, irrespectively whether additional 1-pulses at s_2 exist or not. We therefore obtain the following.

Lemma 12: Let s_1 and s_2 be signals such that $s_1 \leq s_2$ and let C be a channel. Then, C is monotone in the sense that $f_C(s_1) \leq f_C(s_2)$.

We next define a distance for signals, for which channels will turn out to be continuous.

Definition 2: For a signal s and a time T, denote by $\mu_T(s)$ the total duration in $[0, T]$, where s is 1. That is, $\mu_T(s)$ is the measure of the set $\{t \in [0, T] \mid s(t) = 1\}$.

For any two signals s_1 and s_2 and every T, we define their *distance up to time T* by $d_T(s_1, s_2) = \mu_T(|s_1 - s_2|)$.

The detailed proof will start out from an arbitrary finite signal s, which consists of an arbitrary but finite number k of *1-pulses* (a nonzero time where $s = 1$), separated by *0-intervals* (a nonzero time where $s = 0$), within $[0, T]$. We will show that inserting an arbitrary but finite sequence of K additional 1-pulses, having duration $\varepsilon_1, \ldots, \varepsilon_K$ with total duration $\sum_{i=1}^{K} \varepsilon_i \leq \varepsilon$, during arbitrary 0-intervals leads to a signal s' with $\mu_T(s') - \mu_T(s) \leq \varepsilon$ and $\mu_T(f_C(s')) - \mu_T(f_C(s)) = O(\varepsilon)$ for $\varepsilon \to 0$. We will show this (in Theorem 4) by successively inserting ε_i into s and bounding the resulting changes of the measure. Note that doing this iteratively is enabled by Lemma 12. The changes of $\mu_T(f_C(s')) - \mu_T(f_C(s))$ result from: 1) the additional output pulse that is possibly generated by ε_i and 2) the inevitable shifts of *all* the subsequent output transitions.

We start with Lemma 13, which reveals that the worst-case effect on $\mu_T(f_C(s'))$ is caused by an input 1-pulse that is appended at the beginning of the (final) 0-interval where it is inserted. We use the shorthand notation $(x)_+$ for $\max(x, 0)$. The subsequent Lemma 14 will show that $\mu_T(f_C(s'))$ is increased by at most $1 + \delta'_\downarrow(-\delta_{\min})$ times the duration of the inserted input pulse.

Lemma 13: Let s be a signal that is eventually constant 0 and let C be a channel with s on its input. Denote by t_n the time of the last (falling) transition in s and by δ_n its delay in the channel algorithm for C. Then, the maximal $\mu_T(f_C(s'))$ among all s' obtained from s by appending one pulse of length $\Delta > 0$ after time t_n is attained by the addition of the pulse at time $t_n + (\delta_n - \delta_{\min})_+$ (which results in a cancellation of the last transition at the output if $\delta_n \geq \delta_{\min}$, and a right-shift of the last transition at the input if $\delta_n \leq \delta_{\min}$).

Proof: We first show the lemma for $T = \infty$ and then extend the result to finite T. Let s'_γ be the addition of the pulse of length Δ to s at time $t_n + \gamma$.

For all $0 \leq \gamma \leq (\delta_n - \delta_{\min})_+$, the time from t_n to the last transition on $f_C(s'_\gamma)$ is

$$f(\gamma) = \gamma + \Delta + \delta_\downarrow\big(\Delta - \delta_\uparrow(\gamma - \delta_n)\big).$$

In the class of all s'_γ with $0 \leq \gamma \leq (\delta_n - \delta_{\min})_+$, the maximum of $\mu_\infty(f_C(s'_\gamma))$ is attained at the maximum of f. This is because the transition at time $t_n + \gamma$ cancels that at time t_n in this case. The derivative of f is

$$f'(\gamma) = 1 - \delta'_\downarrow\big(\Delta - \delta_\uparrow(\gamma - \delta_n)\big) \cdot \delta'_\uparrow(\gamma - \delta_n).$$

The condition $f'(\gamma) = 0$ is equivalent to $\delta'_\downarrow(\Delta - \delta(\gamma - \delta_n)) = 1/\delta'_\uparrow(\gamma - \delta_n)$, which is in turn equivalent to $\Delta = 0$, as $\delta'_\downarrow(-\delta_\uparrow(t)) = 1/\delta'_\uparrow(t)$ by Lemma 2. Hence, $f'(\gamma)$ is never zero. Since $f'(\gamma) \to 1$ as $\gamma \to \infty$, as the concave $\delta_\downarrow, \delta_\uparrow$ satisfy $\lim_{t\to\infty} \delta'_\uparrow(t) = \lim_{t\to\infty} \delta'_\downarrow(t) = 0$, the derivative of f is always positive, hence f is increasing. This shows that $\gamma = \delta_n - \delta_{\min}$ is a strictly better choice than any other γ in this class.

For the class of s'_γ with $\gamma > (\delta_n - \delta_{\min})_+ \geq 0$, the length of the appended pulse at the output is

$$g(\gamma) = \Delta + \delta_\downarrow\big(\Delta - \delta_\uparrow(\gamma - \delta_n)\big) - \delta_\uparrow(\gamma - \delta_n).$$

Since the transitions at t_n and $t_n + \gamma$ do not cancel in this class, the maximum of $\mu_\infty(f_C(s'_\gamma))$ is attained at the maximum of g. But it is easy to see, using the monotonicity of δ, that g is decreasing. The maximum of g is hence attained at $\gamma = (\delta_n - \delta_{\min})_+$.

Consequently, the choice $\gamma = \gamma_0 = (\delta_n - \delta_{\min})_+$ maximizes $\mu_\infty(f_C(s'_\gamma))$ in any case. By Lemma 3, this choice results in a cancellation of the last (falling) transition in $f_C(s)$, hence a right-shift of the latter in $f_C(s')$. This concludes our proof for $T = \infty$.

Let now T be finite. Denote by T_0 the time of the last, falling, output transition in $f_C(s'_{\gamma_0})$. In this case, transitions of $f_C(s)$ and $f_C(s'_{\gamma_0})$ are the same except the last, falling, transition, which is delayed from $t_n + \delta_n$ to T_0. We distinguish the two cases: 1) $T \leq T_0$ and 2) $T > T_0$. In case 1), the last transition of $f_C(s)$ is delayed beyond T in $f_C(s'_{\gamma_0})$. Because all other transitions remain unchanged in all $f_C(s'_\gamma)$, the measure $\mu_T(f_C(s'_{\gamma_0}))$ is maximal among all $\mu_T(f_C(s'_\gamma))$ if $T \leq T_0$. In case 2), we have $\mu_T(f_C(s'_{\gamma_0})) = \mu_\infty(f_C(s'_{\gamma_0}))$. But because

$\mu_T \le \mu_\infty$ and $\mu_\infty(f_C(s'_{\gamma_0}))$ is maximal among all $\mu_\infty(f_C(s'_\gamma))$, so is $\mu_T(f_C(s'_{\gamma_0}))$ among all $\mu_T(f_C(s'_\gamma))$. ∎

Note that, owing to the above considerations, we can restrict our attention to $T = \infty$ in the sequel.

Lemma 14: Let s be a signal that is eventually constant 0 and let C be a channel. Then, adding a pulse of length ε_1 at or after the last transition of s causes $\mu_T(f_C(s')) \le \mu_T(f_C(s)) + (1 + \delta'_\downarrow(-\delta_{\min}))\varepsilon_1$.

Proof: Denote by t_n the last transition in s and by δ_n its delay. By Lemma 13, the worst-case effect on the $\mu_T(f_C(s'))$ is achieved by appending the pulse at time $t_n + (\delta_n - \delta_{\min})_+$; call the resulting signal s'.

We first assume $\delta_n - \delta_{\min} > 0$. Here, the two new transitions in s' are $t_{n+1} = t_n + \delta_n - \delta_{\min}$ and $t_{n+2} = t_n + \delta_n - \delta_{\min} + \varepsilon_1$. Their corresponding delays are $\delta_{n+1} = \delta_{\min}$ and $\delta_{n+2} = \delta_\downarrow(\varepsilon_1 - \delta_{\min})$. By the mean value theorem of calculus and Lemma 2, the duration of the resulting pulse is

$$\begin{aligned}\varepsilon_1 + \delta_{n+2} - \delta_{n+1} &= \varepsilon_1 + \delta_\downarrow(\varepsilon_1 - \delta_{\min}) - \delta_\downarrow(-\delta_{\min}) \\ &= \left(1 + \delta'_\downarrow(\xi)\right) \cdot \varepsilon_1 \qquad (16)\end{aligned}$$

for some $-\delta_{\min} \le \xi \le \varepsilon_1 - \delta_{\min}$. Since $\delta'_\downarrow(.)$ is decreasing and $\delta'_\downarrow(-\delta_{\min}) > 0$, we hence deduce $0 \le \delta_{n+2} - \delta_{n+1} \le \delta'_\downarrow(-\delta_{\min})\varepsilon_1$. Thus, $\mu_T(f_C(s'_1) - f_C(s)) = \varepsilon_1 + \delta_{n+2} - \delta_{n+1} \le (1 + \delta'_\downarrow(-\delta_{\min}))\varepsilon_1$ as claimed in our lemma.

If $\delta_n - \delta_{\min} \le 0$, then t_n is effectively replaced by $t_n + \varepsilon_1$ in s'_1, i.e., right-shifted. If $\delta_\downarrow(t_n - t_{n-1} + \varepsilon_1 - \delta_{n-1}) \le \delta_{\min}$, then the output signal is not changed since the pulse from t_{n-1} to t_n remains canceled. If $\delta_\downarrow(t_n - t_{n-1} + \varepsilon_1 - \delta_{n-1}) > \delta_{\min}$, then the measure is changed by

$$t_n + \varepsilon_1 + \delta_\downarrow(t_n - t_{n-1} + \varepsilon_1 - \delta_{n-1}) - t_{n-1} - \delta_{n-1}.$$

Using Lemma 3 on s gives $t_n \le t_{n-1} + \delta_{n-1} - \delta_{\min}$ and thus the change in measure is at most $\varepsilon_1 - \delta_{\min} + \delta_\downarrow(\varepsilon_1 - \delta_{\min}) \le (1 + \delta'_\downarrow(-\delta_{\min}))\varepsilon_1$ [recall the derivation of (16)]. This concludes our proof. ∎

Note carefully that Lemma 14 can be applied iteratively for appending an arbitrary sequence of new pulses $\varepsilon_1, \ldots, \varepsilon_K$, one after the other: provided $\sum_{i=1}^K \varepsilon_i \le \varepsilon$, it ensures that the overall change in measure incurred by all the newly inserted pulses is at most $(1 + \delta'_\downarrow(-\delta_{\min}))\varepsilon$.

However, in addition to the increase of $\mu_T(f_C(s'))$ as predicted by Lemma 14, which results from appending a new or enlarging the last pulse at the output, we also need to consider the resulting shifts of *all* the subsequent transitions that may already exist in $f_C(s)$. In particular, consider the rising transition at t_{n+1} that ends the 0-interval where a 1-pulse ε_1 is inserted: the inserted pulse changes its previous output transition and, hence, the delay δ_{n+1}.

To bound the resulting effects, we need to distinguish whether the 0-interval at the input completely vanishes when ε_1 is inserted. The former will be considered in Lemma 16. Otherwise, i.e., if the 0-interval does not vanish completely, there are two possibilities: 1) the newly inserted pulse cancels, and hence right-shifts, the falling output transition caused by t_n, i.e., the beginning of the 0-interval, in the output signal. Lemma 13 revealed that this happens when the inserted pulse starts at $t_n + (\delta_n - \delta_{\min})_+$ (possibly causing the left end of the 0-interval t_n at the input to be shifted right as well). Lemma 14 bounds the resulting shift to be at most $(1 + \delta'_\downarrow(-\delta_{\min}))\varepsilon_1$. Note carefully, though, that this change of measure does not include a possible effect on the delay δ_{n+1} of the transition at time t_{n+1} that ends the 0-interval. The latter will be dealt with in Lemma 18. For the remaining case 2), where the inserted pulse starts later than $t_n + (\delta_n - \delta_{\min})_+$, Lemma 15 bounds its effect on δ_{n+2}.

Lemma 15: Let s be a signal that is eventually constant 1, with t_n and t_{n+1} denoting the beginning and the end of the last 0-interval of s, and δ_n and δ_{n+1} the delays caused by channel C. Then, inserting a pulse of length $\varepsilon_1 < t_{n+1} - t_n$ at time $t > t_n + (\delta_n - \delta_{\min})_+$ within this 0-interval (possibly causing the end of the 0-interval t_{n+1} to be shifted left) causes δ_{n+1} to decrease by at most $(1 + \delta'_\uparrow(-\delta_{\min}))\varepsilon_1$, i.e., $\mu_T(f_C(s'))$ to increase by at most $(1 + \delta'_\uparrow(-\delta_{\min}))\varepsilon_1$.

Proof: Let $a = t_{n+1} - t_n - \varepsilon_1 - \delta_n$, and $v \ge 0$ be the difference between the falling transition of the inserted pulse ε_1 and t_{n+1}. If $v = 0$, then t_{n+1} is shifted left by ε_1 at the input. We first show that $a \ge -\delta_{\min}$. This is trivially satisfied for $\delta_n \le \delta_{\min}$ since $\varepsilon_1 < t_{n+1} - t_n$. For $\delta_n > \delta_{\min}$, we have $t > t_n + \delta_n - \delta_{\min}$. Since obviously $t_{n+1} \ge t + \varepsilon$, $a \ge -\delta_{\min}$ also holds in this case.

Let $\eta_1 \ge 0$ be the difference between $\delta_{n+1} = \delta_\uparrow(t_{n+1} - t_n - \delta_n) = \delta_\uparrow(a + \varepsilon_1)$ (before insertion) and $\delta'_{n+1} = \delta_\uparrow(t_{n+1} - (t_{n+1} - v + \delta_d)) = \delta_\uparrow(v - \delta_d)$ (after insertion), with δ_d denoting the delay of the falling transition of the inserted pulse. As the delay δ_u of the rising transition of the inserted pulse is $\delta_u = \delta_\uparrow(t_{n+1} - v - \varepsilon_1 - t_n - \delta_n) = \delta_\uparrow(a - v)$, we get $\delta_d = \delta_\downarrow(t_{n+1} - v - t_{n+1} + v + \varepsilon_1 - \delta_u) = \delta_\downarrow(\varepsilon_1 - \delta_\uparrow(a - v))$. We thus find

$$\eta_1 = \delta_\uparrow(a + \varepsilon_1) - \delta_\uparrow\big(v - \delta_\downarrow\big(\varepsilon_1 - \delta_\uparrow(a - v)\big)\big). \qquad (17)$$

Differentiating with respect to v, we obtain that η'_1 is equal to

$$\begin{aligned}&-\delta'_\uparrow\big(v - \delta_\downarrow\big(\varepsilon_1 - \delta_\uparrow(a - v)\big)\big) \\ &\quad \times \Big(1 - \delta'_\downarrow\big(\varepsilon_1 - \delta_\uparrow(a - v)\big)\delta'_\uparrow(a - v)\Big).\end{aligned}$$

Since $\delta'_\downarrow(\varepsilon_1 - \delta_\uparrow(a - v)) < \delta'_\downarrow(-\delta_\uparrow(a - v)) = 1/\delta'_\uparrow(a - v)$ by Lemma 2, we obtain $\eta'_1 < 0$. Therefore, η_1 attains its maximum for $v = 0$. Using the involution property, this reveals $\eta_1 \le \delta_\uparrow(a + \varepsilon_1) - \delta_\uparrow(-\delta_\downarrow(\varepsilon_1 - \delta_\uparrow(a))) = \delta_\uparrow(a + \varepsilon_1) + \varepsilon_1 - \delta_\uparrow(a)$. Using the concavity of $\delta_\uparrow(.)$, we finally obtain $\eta_1 \le \varepsilon_1(1 + \delta'_\uparrow(a)) \le (1 + \delta'_\uparrow(-\delta_{\min}))\varepsilon_1$ since $a \ge -\delta_{\min}$. ∎

The next lemma handles the case where the inserted 1-pulse fills up the 0-interval completely, i.e., $t_{n+1} - t_n = \varepsilon_1$ in the notation of Lemma 15. It is actually the dual of this lemma, in the sense that, rather than inserting a 1-pulse in a 0-interval, it deals with deleting a 0-interval in a 1-pulse.

Lemma 16: Let s be a signal of channel C that is eventually constant 0, with t_{n+2} denoting the last falling input transition and δ_{n+2} its delay, with the last preceding 0-interval $\varepsilon_1 = t_{n+1} - t_n$ starting at $t_n > t_{n-1} + (\delta_{n-1} - \delta_{\min})_+$. If the signal s' is obtained from s by entirely dropping the last preceding 0-interval, i.e., s' contains a single 1-pulse between t_{n-1} and t_{n+1}, then the resulting delay δ'_{n+2} satisfies $\delta'_{n+2} \le (1 + \delta'_\downarrow(-\delta_{\min}))\varepsilon_1$, i.e., $\mu_T(f_C(s'))$ increases by at most $(1 + \delta'_\downarrow(-\delta_{\min}))\varepsilon_1$.

Proof: It is obvious that the situation is exactly dual to Lemma 15, its proof hence applies literally after changing every rising transition to a falling one and vice versa. ∎

The only remaining issue is to bound the propagation of the shift of a single output transition to later output transitions, which will be done in Lemma 18 below. It relies on the following technical lemma.

Lemma 17: Let s be a signal containing a rising transition at t_3, and let t_2 and t_1 be the times of the two previous transitions, respectively. With $\Theta_2 = t_3 - t_2 > 0$ and $\Theta_1 = t_2 - t_1 > 0$ denoting the previous 0-interval and 1-pulse, starting at t_2 and t_1, respectively, and δ_1 denoting the channel delay of the transition at t_1, the other channel delays are $\delta_2 = \delta_\downarrow(\Theta_1 - \delta_1)$ and $\delta_3 = \delta_\uparrow(\Theta_2 - \delta_2)$, and

$$\delta'_\uparrow(\Theta_2 - \delta_2)\delta'_\downarrow(\Theta_1 - \delta_1) < 1 \tag{18}$$

irrespectively of whether there are cancellations or not.

Proof: Due to concavity of $\delta_\uparrow(.)$, we observe

$$\begin{aligned}\delta_3 &= \delta_\uparrow(\Theta_2 - \delta_2) = \delta_\uparrow\big(\Theta_2 - \delta_\downarrow(\Theta_1 - \delta_1)\big)\\ &\le \delta_\uparrow\big(-\delta_\downarrow(\Theta_1 - \delta_1)\big) + \Theta_2\delta'_\uparrow\big(-\delta_\downarrow(\Theta_1 - \delta_1)\big)\\ &= -\Theta_1 + \delta_1 + \frac{\Theta_2}{\delta'_\downarrow(\Theta_1 - \delta_1)}\end{aligned} \tag{19}$$

in the final step, we used the involution property and Lemma 2. Invoking the latter once again, we find the following expression for $\delta'_\downarrow(-\delta_3)$:

$$\begin{aligned}\frac{1}{\delta'_\uparrow\big(\Theta_2 - \delta_\downarrow(\Theta_1 - \delta_1)\big)} &= \delta'_\downarrow\big(-\delta_\uparrow\big(\Theta_2 - \delta_\downarrow(\Theta_1 - \delta_1)\big)\big)\\ &\ge \delta'_\downarrow\left(\Theta_1 - \delta_1 - \frac{\Theta_2}{\delta'_\downarrow(\Theta_1 - \delta_1)}\right)\\ &> \delta'_\downarrow(\Theta_1 - \delta_1)\end{aligned}$$

which implies (18) and completes our proof. ∎

Finally, Lemma 18 considers a 1-pulse and a consecutive 0-interval and focuses on the falling and rising transition of the latter. It bounds the variation of their delays when the delay of the rising transition of the 1-pulse is decreased by ε_1, under the assumption that the 1-pulse did not cancel at the output initially. Note that Lemma 18 also holds (analogously) for the dual situation, i.e., a signal with the opposite transitions.

Lemma 18: Let s be an input signal of channel C containing a rising transition at t_3, with delay δ_3, and let t_2 and t_1 resp. δ_2 and δ_1 be the times of the previous transitions resp. their delays. Assume that the preceding 1-pulse is not canceled at the output, i.e., $\delta_2 > \delta_{\min}$. With $\Theta_2 = t_3 - t_2 > 0$ and $\Theta_1 = t_2 - t_1 > 0$ starting at t_2 and t_1, respectively, a shifted signal σ' with decreased $\delta'_1 = \delta_1 - \varepsilon_1$, $\varepsilon_1 > 0$, satisfies $\delta_2 \le \delta'_2 < \delta_2 + \delta'_\downarrow(-\delta_{\min})\varepsilon_1$ and $\delta_3 \ge \delta'_3 > \delta_3 - \delta'_\uparrow(\Theta_2 - \delta'_2)\delta'_\downarrow(\Theta_1 - \delta_1)\varepsilon_1$. If $\delta_\downarrow(.)$ is two-times continuously differentiable and $\varepsilon_1 \le -\delta'_\downarrow(-\delta_{\min})/(2\delta''_\downarrow(-\delta_{\min}))$, then $\delta'_3 > \delta_3 - 2\varepsilon_1$.

Proof: Using the concavity of $\delta_\downarrow(.)$ in $\delta_2 = \delta_\downarrow(\Theta_1 - \delta_1)$, we obtain $\delta'_2 = \delta_\downarrow(\Theta_1 - \delta'_1) = \delta_\downarrow(\Theta_1 - \delta_1 + \varepsilon_1) \le \delta_\downarrow(\Theta_1 - \delta_1) + \varepsilon_1\delta'_\downarrow(\Theta_1 - \delta_1)$. Since $\delta_2 > \delta_{\min}$, Lemma 2 implies $\Theta_1 - \delta_1 > -\delta_{\min}$ and hence $\delta'_\downarrow(\Theta_1 - \delta_1) < \delta'_\downarrow(-\delta_{\min})$. This confirms $\delta'_2 < \delta_2 + \delta'_\downarrow(-\delta_{\min})\varepsilon_1$.

As for $\delta'_3 = \delta_\uparrow(\Theta_2 - \delta'_2)$, the mean value theorem of calculus yields, for some $\xi \in [-(\delta'_2 - \delta_2), 0]$

$$\begin{aligned}\delta'_3 &= \delta_\uparrow\big(\Theta_2 - \delta_2 - \big(\delta'_2 - \delta_2\big)\big)\\ &= \delta_\uparrow(\Theta_2 - \delta_2) - \big(\delta'_2 - \delta_2\big)\delta'_\uparrow(\Theta_2 - \delta_2 + \xi)\\ &> \delta_\uparrow(\Theta_2 - \delta_2) - \varepsilon_1\delta'_\downarrow(\Theta_1 - \delta_1)\delta'_\uparrow\big(\Theta_2 - \delta'_2\big)\end{aligned} \tag{20}$$

for the last step, we used the upper bound from the previous paragraph and the monotonicity of $\delta'_\uparrow(.)$. Recalling $\delta_3 = \delta_\uparrow(\Theta_2 - \delta_2)$ confirms the lower bound on δ'_3 stated in our lemma.

If $\delta_\downarrow(.)$ is two-times continuously differentiable, we find by convexity of $\delta'_\downarrow(.)$ that $\delta'_\downarrow(x + \varepsilon_1) \ge \delta'_\downarrow(x) + \varepsilon_1\delta''_\downarrow(x)$; note that $\delta''_\downarrow(x)$ is negative and increasing. Consequently, $\delta'_\downarrow(x + \varepsilon_1)/\delta'_\downarrow(x) \ge 1 + \varepsilon_1\delta''_\downarrow(x)/\delta'_\downarrow(x) \ge 1 + \varepsilon_1\delta''_\downarrow(-\delta_{\min})/\delta'_\downarrow(-\delta_{\min})$ for $x \ge -\delta_{\min}$. Using this with $x = \Theta_1 - \delta_1 > -\delta_{\min}$ reveals

$$\delta'_\downarrow(\Theta_1 - \delta_1)\delta'_\uparrow\big(\Theta_2 - \delta'_2\big) \le \frac{\delta'_\downarrow\big(\Theta_1 - \delta'_1\big)}{1 + \varepsilon_1\frac{\delta''_\downarrow(-\delta_{\min})}{\delta'_\downarrow(-\delta_{\min})}} \cdot \delta'_\uparrow\big(\Theta_2 - \delta'_2\big) < 2$$

due to (18) and $\varepsilon_1 \le -\delta'_\downarrow(-\delta_{\min})/(2\delta''_\downarrow(-\delta_{\min}))$. This completes our proof. ∎

Combining the previous lemmas finally leads to the proof of continuity.

Theorem 4: Let C be a channel and let $T \ge 0$. Then, the mapping $s \mapsto f_C(s)$ is continuous with respect to the distance d_T.

Proof: Let s be a signal. We show that, if $\mu_T(|s - s'|) \to 0$, then $\mu_T(|f_C(s) - f_C(s')|) \to 0$. Because

$$|s - s'| = \big(\max\big(s, s'\big) - s\big) + \big(s - \min(s, s')\big)$$

where $\max(s, s')(t) = \max(s(t), s'(t))$ and $\min(s, s')(t) = \min(s(t), s'(t))$ for all t, the condition $\mu_t(|s - s'|) \to 0$ is equivalent to the conjunction of both $\mu_T(|\max(s, s') - s|) \to 0$ and $\mu_T(|s - \min(s, s')|) \to 0$. Because $\max(f_C(s), f_C(s')) \le f_C(\max(s, s'))$ and $\min(f_C(s), f_C(s')) \ge f_C(\min(s, s'))$ by Lemma 12

$$|f_C(s) - f_C(s')| \le f_C\big(\max\big(s, s'\big)\big) - f_C\big(\min\big(s, s'\big)\big)$$

which shows that we can suppose without loss of generality $s' \ge s$ for all n.

Let an arbitrary sequence of 1-pulses $\varepsilon_1, \ldots, \varepsilon_K$, ordered by insertion times, with total measure $\sum_{i=1}^{K} \varepsilon_i \le \varepsilon$ be given, and let s' be obtained from the signal s by adding these pulses. Since T is finite, s consists of an arbitrary but finite number k of *1-pulses* (a nonzero time where $s = 1$), separated by *0-intervals* (a nonzero time where $s = 0$), within $[0, T]$. Thanks to monotonicity of our measure, recall Lemma 12, we can iteratively apply the appropriate lemmas for every ε_i, $1 \le i \le K$, as follows: First, Lemma 14 shows that the increase in measure incurred directly from the inserted ε_i is $O(\varepsilon_i)$. Furthermore, Lemma 15 (resp. Lemma 16) bound the measure incurred by the shift of the first output transition following the newly inserted 1-pulse resp. deleted 0-interval caused by inserting ε_i to $O(\varepsilon_i)$. For a 1-pulse that does not yield a newly inserted output pulse, but rather a right-shift of the falling transition at the beginning of the 0-interval, the resulting change in measure is bounded to $O(\varepsilon_i)$ by Lemma 14 already. Finally, consecutively applying Lemma 18 to the subsequent rising transition (or, if the subsequent 1-pulse cancels, its dual to the previous falling transition) shows that the shift of the every of the at most $2k$ subsequent transition is at most $O(\varepsilon_i)$.

Observe that inserting some 1-pulse ε_i does not affect a possibly newly generated 1-pulse caused by $\varepsilon_1, \ldots, \varepsilon_{i-1}$. As there are only finitely many affected later transitions, the resulting change of $\mu_T(f_C(s_n) - f_C(s)) = O(\varepsilon_i)$ overall. Summing up all the contributions leads to $\mu_T(f_C(s_n) - f_C(s)) = O(\varepsilon)$ and thus confirms $\mu_T(f_C(s_n) - f_C(s)) \to 0$ as $\varepsilon \to 0$. ∎

B. Impossibility in Forward Circuits

A circuit is called a *forward circuit* if its graph is acyclic. Forward circuits are exactly those circuits that do not contain feed-back loops. Equipped with the continuity of involution channels and the fact that the composition of continuous functions is continuous, it is not too difficult to prove that the inherently discontinuous SPF problem cannot be solved with forward circuits.

Theorem 5: No forward circuit solves bounded SPF.

Proof: Suppose that there exists a forward circuit that solves bounded SPF with stabilization time bound K. Denote by s_Δ its output signal when feeding it a Δ-pulse at time 0 as the input. Because s_Δ in forward circuits is a finite composition of continuous functions by Theorem 4, the measure $\mu_T(s_\Delta)$ depends continuously on Δ.

By the nontriviality condition F3) of the SPF problem, there exists some Δ_0 such that s_{Δ_0} is not zero. Set $T = 2\Delta_0 + K$.

Let $\varepsilon > 0$ be smaller than both Δ_0 and $\mu_T(s_{\Delta_0})$. We show a contradiction by finding a Δ such that s_Δ either contains a pulse of length less than ε [contradiction to the no short pulses condition F4)] or contains a transition after time $\Delta + K$ [contradicting the bounded stabilization time condition F5)].

Since $\mu_T(s_\Delta) \to 0$ as $\Delta \to 0$ by the no generation condition F2) of SPF, there exists a $\Delta_1 < \Delta_0$ such that $\mu_T(s_{\Delta_1}) = \varepsilon$ by the intermediate value property of continuity. By the bounded stabilization time condition F5), there are no transitions in s_{Δ_1} after time $\Delta_1 + K$. Hence, s_{Δ_1} is 0 after this time because otherwise it is 1 for the remaining duration $T - (\Delta_1 + K) > \Delta_0 > \varepsilon$, which would mean that $\mu_T(s_{\Delta_1}) > \varepsilon$. Consequently, there exists a pulse in s_{Δ_1} before time $\Delta_1 + K$. But any such pulse is of length at most ε because $\mu_{\Delta_1+K}(s_{\Delta_1}) \le \mu_T(s_{\Delta_1}) = \varepsilon$. This is a contradiction to the no short pulses condition F4). ■

C. Simulation With Unrolled Circuits

We next show how to simulate (part of) an execution of an arbitrary circuit $\mathcal{C}$ by a forward circuit $\mathcal{C}'$ generated from $\mathcal{C}$ by unrolling of feedback channels. Intuitively, the deeper the unrolling, the longer the time $\mathcal{C}'$ behaves as $\mathcal{C}$.

Definition 3: Let $\mathcal{C}$ be a circuit, V a vertex of $\mathcal{C}$, and $k \ge 0$. We define the *k-unrolling of $\mathcal{C}$ from V*, denoted by $\mathcal{C}_k(V)$, to be a directed acyclic graph with a single sink, constructed as follows.

The unrolling $\mathcal{C}_k(I)$ from input port I is just a copy of that input port. The unrolling $\mathcal{C}_k(O)$ from output port O with incoming channel C and predecessor V comprises a copy of the output port $O^{(k)}$ and the unrolled circuit $\mathcal{C}_k(V)$ with its sink connected to $O^{(k)}$ by channel C.

The 0-unrolling $\mathcal{C}_0(B)$ from gate B is a Boolean gate X_v without inputs and the constant output value v equal to B's initial value. For $k > 0$, the k-unrolling $\mathcal{C}_k(B)$ from gate B comprises a copy of that gate $B^{(k)}$ with the same initial value and Boolean function. Additionally, for every incoming channel C from V to B in $\mathcal{C}$, it contains the circuit $\mathcal{C}_{k-1}(V)$ with its sink connected to $B^{(k)}$ with a copy of channel C with the same channel functions. All copies of the same input port are considered identical.

To each component Γ in $\mathcal{C}_k(V)$, we assign a value $z(\Gamma) \in \mathbb{N}_0 \cup \{\infty\}$ as follows: $z(\Gamma) = \infty$ if Γ has no predecessor (in particular, is an input port) and $\Gamma \notin \{X_0, X_1\}$, $z(X_0) = z(X_1) = 0$, $z(V) = z(C) = z(U)$ if V is an output port with incoming channel C from U to V, $z(B) = \min\{z(c) \mid c \in C^B\}$ if B is a gate with set of incoming channels C^B, and $z(C) = 1 + z(U)$ if C is a channel from vertex U to a gate. Fig. 6 shows an example of a circuit and an unrolled circuit with its z values.

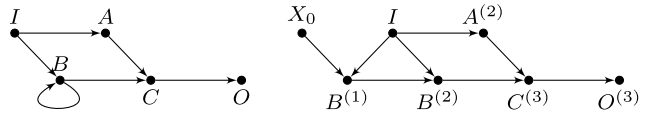

Fig. 6. Circuit $\mathcal{C}$ (left) and $\mathcal{C}_3(O)$ (right) under the assumption that the gate B has initial value 0. It is $z(X_0) = 0$, $z(I) = z(A^{(2)}) = \infty$, $z(B^{(1)}) = 1$, $z(B^{(2)}) = 2$, $z(C^{(3)}) = 3$, and $z(O^{(3)}) = 3$: The z-values of all channels from a vertex V are $z(V) + 1$, except for the channel C' from $C^{(3)}$ to $O^{(3)}$, with $z(C') = z(C^{(3)}) = 3$.

Noting that for every component Γ in $\mathcal{C}_k(V)$, $z(\Gamma)$ is the number of channels on the shortest path from an X_v node to Γ (excluding channels to output ports), or $z(\Gamma) = \infty$ if no such path exists, we get the following.

Lemma 19: The z-value assigned to the sink vertex $V^{(k)}$ of a k-unrolling $\mathcal{C}_k(V)$ of $\mathcal{C}$ from V satisfies $z(V^{(k)}) \ge k$.

We also adapt the execution construction algorithm (Algorithm 2 in Section V) to assign to each generated transition e a *causal depth* $d(e)$. All input transitions and initial transitions of gates (line 11) have causal depth 0. All transitions added to channels in lines 3 and 8 have causal depth 1. All transitions added to gates in line 21 at time t get causal depth $d(e)$ equal to the maximum over all causal depths $d(e')$ of fixed transitions of the gate's incoming channels up to time t. All transitions e added to channels in line 32 get causal depth $d(e)$ equal to $d(e) = d(e') + 1$ where e' is the event at time t of the departing gate B_1 of channel C. When transitions are moved or copied in lines 16 and 40, they retain their causal depths. We immediately get the following.

Lemma 20: For all $k \ge 1$, 1) the simulation algorithm never assigns a causal depth larger than $k+1$ to a transition generated in iteration k and 2) at the end of iteration k the sequence of causal depths of transitions in s_Γ is nondecreasing for all components Γ.

We are now in the position to prove the main result of a circuit simulated by an unrolled circuit. The proof of the theorem has been postponed to the Appendix.

Theorem 6: Let $\mathcal{C}$ be a circuit with input port I and output port O that solves bounded SPF. Let $\mathcal{C}_k(O)$ be an unrolling of $\mathcal{C}$, Γ a component in $\mathcal{C}$, and Γ' a copy of Γ in $\mathcal{C}_k(O)$. For all input signals i, if a transition e (fixed or pending) is generated for Γ by the execution construction algorithm run on circuit $\mathcal{C}$ with input signal i and $d(e) \le z(\Gamma')$, then e is also generated for Γ' by the algorithm run on circuit $\mathcal{C}_k(O)$ with input signal i; and vice versa.

D. Impossibility Result

We can now turn to the proof that bounded SPF is not solvable, even with nonforward circuits. For that, we first note that the number N_k of transitions within circuit $\mathcal{C}$ in the time interval $(k \cdot \delta^{\mathcal{C}}_{\min}, (k+1) \cdot \delta^{\mathcal{C}}_{\min}]$ is upper-bounded by $D^+ N_{k-1} + I_k$, where D^+ is the largest out-degree of vertices in the circuit and I_k is the number of input transitions in the same time interval. This follows from the same argument as Lemma 5. As a consequence, all transitions up to a bounded time have bounded causal depth.

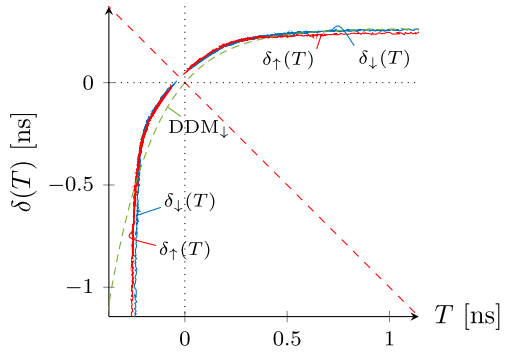

Fig. 7. Measured $\delta_\downarrow$ (blue) and $\delta_\uparrow$ (red) for UMC-90 inverter chain for $V_{DD} = 0.6\ V$, which support the involution hypothesis. By contrast, there is no perfect fit for the exponential DDM delay function (dashed green). From [13].

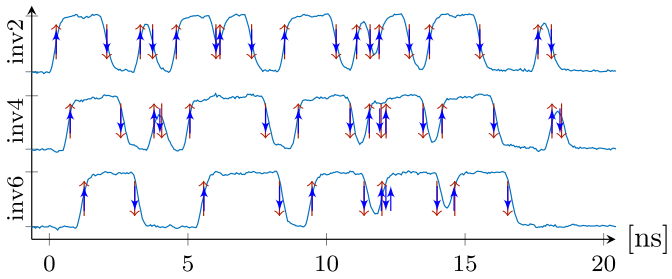

Fig. 8. Measured waveform (solid) for the UMC-90 inverter chain, with the predictions according to the involution model (red long up/down-arrows) and the DDM (blue short up/down arrows). From [13].

We next note that the impossibility of bounded SPF also implies the impossibility of bounded SPF when restricting pulse lengths to be at most some $\Delta_0 > 0$. Now let ζ be an upper bound on the causal depth of transitions up to the SPF stabilization time bound $\Delta_0 + K$. Then, by Theorem 6 and Lemma 19, the ζ-unrolled circuit $\mathcal{C}_\zeta(O)$ has the same output transitions as the original circuit $\mathcal{C}$ up to time $\Delta_0 + K$, and hence, by definition of bounded SPF, the same transitions for all times. But since $\mathcal{C}_\zeta(O)$ is a forward circuit, it cannot solve bounded SPF by Theorem 5, i.e., neither can $\mathcal{C}$.

The above arguments show the following theorem.

Theorem 7: No circuit solves bounded SPF.

We have hence proved that the bounded SPF problem is not solvable in the involution model, matching physical reality.

VIII. Experimental Accuracy Evaluation

To support our claim that the involution model is not only realistic but also faithful, we provide an overview of the major results of our experimental accuracy evaluation for a real circuit described in [13], which used both simulations and measurements. Primary target is a chain of inverters, which allows to track the reshaping of pulse trains along the inverter stages. For our simulations, we considered two different VLSI technologies available to us (UMC-65 nm and UMC-90 nm) as well as different supply voltages (from nominal down to close to the subthreshold regime, which causes the delays to increase). Moreover, the inverter chains were operated at their speed limits, we also conducted dedicated measurements to validate the accuracy of our SPICE simulations: comparing the measurement results with corresponding simulations (using the post-layout netlists extracted from the ASIC design) indeed showed a very good match. For all our measurements, we used a custom UMC-90 ASIC [21] containing an inverter chain monitored by low-intrusive high-speed on-chip analog amplifiers attached to a high-speed real-time oscilloscope.

More explicitly, our experiments were used to validate the following two features of our involution model.

1) *Involution Property:* By using input pulses of decreasing width, we empirically determine $\delta_\uparrow$ and $\delta_\downarrow$ for a single inverter. The resulting graphs for a supply voltage of $V_{DD} = 0.6\ V$ are depicted in Fig. 7. The involution property $-\delta_\downarrow(-\delta_\uparrow(T)) = T$ was used to extrapolate the functions' values for small T which do not allow direct delay measurements. The graphs support our claim that real delay functions can be approximated well by involutions. By contrast, the exponential delay function of DDM channels [6] cannot be fit to the experimental data over the whole range of T. Note that this misfit is even more pronounced in simulations for UMC-65.
2) *Good Accuracy:* Using representative examples of pulse trains, we show that the involution model with the empirically determined $\delta_\uparrow$ and $\delta_\downarrow$ provides very good accuracy. We provide an explicit simulation algorithm for this purpose, which also has been implemented in VHDL and thus allows timing simulations in our model using standard digital circuit simulators like ModelSim. Fig. 8 shows an example analog pulse train together with the digital predictions from both the involution model and the DDM, both showing good accuracy even in presence of short pulses decaying when propagated. Simulation speeds were greatly reduced compared to SPICE: for the 20 ns trace in Fig. 8, VHDL simulation time of the involution model was less than 1 s compared to 33.42 s SPICE simulation time (ngSPICE revision 26, 0.01 ps resolution) run on a MacBook Pro (3.1 GHz Intel Core i7, 2 cores, and 16-GB memory). We further measured accuracy of the involution model and DDM model, by the sum of times where the binary signals differed from the discretized SPICE signal. Summing over the signals inv4 and inv6 in Fig. 8 (inv2 was used to calibrate the DDM and involution models) we obtained 2.174 ns (involution model) and 2.471 ns (DDM model) out of 40 ns time signal differences. Therefore, with all due care, we can indeed claim that our involution model may have good accuracy overall, and is hence indeed a promising candidate for a faithful glitch propagation model.

IX. Conclusion

We showed that binary circuit models based on involution channels are a promising candidate for faithfully modeling glitch propagation in circuits, in the sense that they allow to design circuits solving the SPF problem precisely when this is possible with physical circuits. Involution channels differ from all existing single-history channels, which do not share this property, in that they are also continuous with respect to dropping/inserting input pulses.

Although our results prove that involution channels are superior to all alternative channel models known so far, there are several very important questions which are still open: first,

we did not at all address the question of quantitatively comparing the modeling accuracy of alternative models: although [13] supports our hypothesis that the modeling accuracy of properly chosen instances of our involution channels surpasses the one of alternative channel models, we cannot rule out the possibility that a nonfaithful model like DDM works better in some situations. Second, addressing the SPF problem is only a first step toward a digital model for metastability generation and propagation. Finally, the delay of multi-input gates is likely to depend on the history of all of its input signals; which we cannot express in our current model. Needless to say, addressing these questions requires major efforts and is hence a subject of future research.

REFERENCES

[1] *CCS Timing Library Characterization Guidelines, Version 3.4*, Synopsis Inc., Mountain View, CA, USA, Oct. 2016.

[2] *Effective Current Source Model (ECSM) Timing and Power Specification, Version 2.1.2*, Cadence Design Syst., San Jose, CA, USA, Jan. 2015.

[3] "CCS timing, v2.0," Mountain View, CA, USA, Synopsis Inc., White Paper, 2006.

[4] L. W. Nagel and D. Pederson, "SPICE (simulation program with integrated circuit emphasis)," EECS Dept., Univ. California at Berkeley, Berkeley, CA, USA, Rep. UCB/ERL M382, 1973.

[5] S. H. Unger, "Asynchronous sequential switching circuits with unrestricted input changes," *IEEE Trans. Comput.*, vol. C-20, no. 12, pp. 1437–1444, Dec. 1971.

[6] M. J. Bellido-Díaz, J. Juan-Chico, A. J. Acosta, M. Valencia, and J. L. Huertas, "Logical modelling of delay degradation effect in static CMOS gates," *IEE Proc. Circuits Devices Syst.*, vol. 147, no. 2, pp. 107–117, Apr. 2000.

[7] M. J. Bellido-Díaz, J. Juan-Chico, and M. Valencia, *Logic-Timing Simulation and the Degradation Delay Model*. London, U.K.: Imperial College Press, 2006.

[8] F. N. Najm, "A survey of power estimation techniques in VLSI circuits," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 2, no. 4, pp. 446–455, Dec. 1994.

[9] M. Favalli and L. Benini, "Analysis of glitch power dissipation in CMOS ICs," in *Proc. Int. Symp. Low Power Design (ISLPED)*, 1995, pp. 123–128.

[10] M. Függer, T. Nowak, and U. Schmid, "Unfaithful glitch propagation in existing binary circuit models," *IEEE Trans. Comput.*, vol. 65, no. 3, pp. 964–978, Mar. 2016.

[11] L. R. Marino, "The effect of asynchronous inputs on sequential network reliability," *IEEE Trans. Comput.*, vol. C-26, no. 11, pp. 1082–1090, Nov. 1977.

[12] F. U. Rosenberger, C. E. Molnar, T. J. Chaney, and T.-P. Fang, "Q-modules: Internally clocked delay-insensitive modules," *IEEE Trans. Comput.*, vol. 37, no. 9, pp. 1005–1018, Sep. 1988.

[13] R. Najvirt, U. Schmid, M. Hofbauer, M. Függer, T. Nowak, and K. Schweiger, "Experimental validation of a faithful binary circuit model," in *Proc. 25th ed. Great Lakes Symp. VLSI (GLSVLSI)*, 2015, pp. 355–360. [Online]. Available: http://doi.acm.org/10.1145/2742060.2742081

[14] M. A. Horowitz, "Timing models for MOS circuits," Ph.D. dissertation, Integr. Circuits Lab., Stanford Univ., Stanford, CA, USA, 1984.

[15] T.-M. Lin and C. A. Mead, "Signal delay in general RC networks," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 3, no. 4, pp. 331–349, Oct. 1984.

[16] L. T. Pillage and R. A. Rohrer, "Asymptotic waveform evaluation for timing analysis," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 9, no. 4, pp. 352–366, Apr. 1990.

[17] A.-C. Deng and Y.-C. Shiau, "Generic linear RC delay modeling for digital CMOS circuits," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 9, no. 4, pp. 367–376, Apr. 1990.

[18] J. A. Brzozowski and J. C. Ebergen, "On the delay-sensitivity of gate networks," *IEEE Trans. Comput.*, vol. 41, no. 11, pp. 1349–1360, Nov. 1992.

[19] J. Juan-Chico, M. J. Bellido, P. Ruiz-de-Clavijo, A. J. Acosta, and M. Valencia, "Degradation delay model extension to CMOS gates," in *Integrated Circuit Design* (LNCS 1918). Heidelberg, Germany: Springer, 2000, pp. 149–158.

[20] A. Millán, J. Juan, M. J. Bellido, P. Ruiz-de Clavijo, and D. Guerrero, "Characterization of normal propagation delay for delay degradation model (DDM)," in *Integrated Circuit Design* (LNCS 2451). Heidelberg, Germany: Springer, 2002, pp. 477–486.

[21] M. Hofbauer *et al.*, "Pulse shape measurements by on-chip sense amplifiers of single event transients propagating through a 90 nm bulk CMOS inverter chain," *IEEE Trans. Nucl. Sci.*, vol. 59, no. 6, pp. 2778–2784, Dec. 2012.

Matthias Függer received the M.Sc. and Ph.D. degrees in computer engineering from TU Wien, Vienna, Austria, in 2006 and 2010, respectively.

He was an Assistant Professor with TU Wien, and as a Post-Doctoral Researcher with LIX, École polytechnique, Palaiseau, France, and Max-Planck-Institut für Informatik, Saarbrücken, Germany. He is currently with the Centre National de la Recherche Scientifique Researcher, Laboratoire Spécification et Vérification, ENS Paris-Saclay, Univeristé Paris-Saclay and Inria, Cachan, France, and co-leading the Group HicDiesMeus on Highly Constrained Discrete Agents for Modeling Natural Systems. His current research interests include distributed computing for highly restricted agents, circuit design, and synthetic microbiology.

Robert Najvirt received the M.Sc. degree in computer engineering from TU Wien, Vienna, Austria, in 2011, where he is currently pursuing the Ph.D. degree with a research focuses on problems in digital circuit design involving metastability and asynchronous circuits.

He has been an Assistant Professor and a Researcher with the Embedded Computing Systems Group, TU Wien. He is currently an Industrial ASIC Design Engineer.

Thomas Nowak received the M.Sc. degree in computer engineering from TU Wien, Vienna, Austria, in 2010, and the Ph.D. degree in computer science from the École polytechnique, Palaiseau, France, in 2014.

He was a Post-Doctoral Fellow and a Research Assistant with the École normale supérieure, Paris, France. He is an Associate Professor of computer science with the Université Paris-Sud, Orsay, France, and École polytechnique. He is co-leading the HicDiesMeus Group on Highly Constrained Discrete Agents for Modeling Natural Systems. His current research interests include discrete event systems and distributed algorithms in dynamic fault-tolerant networks.

Ulrich Schmid received the degree in computer science and mathematics from TU Wien, Vienna, Austria.

He is a Full Professor and the Head of the Embedded Computing Systems Group, Institut für Technische Informatik, TU Vienna, where he spent several years in industrial electronics and embedded systems design. He has authored and coauthored numerous papers in the field of theoretical and technical computer science. His current research interests include the mathematical and logic-based analysis of fault-tolerant distributed algorithms and real-time systems, with special emphasis on their application in systems-on-chips, and networked embedded systems.

Prof. Schmid was a recipient of several awards and prices, like the Austrian START-price in 1996.