Saad Saleh and Boris Koldehofe. **Memristor-based Network Switching Architecture for Energy Efficient Cognitive Computational Models**. In Proceedings of the 18th ACM International Symposium on Nanoscale Architectures (NANOARCH 2023), ACM, 4 pages, 2023. doi: 10.1145/3611315.3633272.

Memristor-based Network Switching Architecture for Energy Efficient Cognitive Computational Models

Saad Saleh
Bernoulli Institute AND Groningen Cognitive Systems and Materials Center (CogniGron),
University of Groningen
Groningen, Groningen, Netherlands
s.saleh@rug.nl

Boris Koldehofe
Institute of Applied Computer Science,
Technische Universität Ilmenau
Ilmenau, Thuringia, Germany
boris.koldehofe@tu-ilmenau.de

ABSTRACT

The Internet makes use of high performance network switches in order to route network traffic from end users to servers. Despite line-rate performance, the current switches consume huge energy and cannot support more expressive learning models, like cognitive functions using neuromorphic computations. The major reason is the use of transistors in the underlying Ternary Content-Addressable Memory (TCAM) which is volatile and supports digital computations only. These shortcomings can be bypassed by developing network memories building on novel components, like Memristors, due to their nonvolatile, nanoscale and analog storage/processing characteristics. In this paper, we propose the use of a novel memristor-based Probabilistic Associative Memory, PA*m*M, which provides both digital (deterministic) and analog (probabilistic) outputs for supporting cognitive computational models in network switches. The traditional digital operations can be supported by a memristor-based energy efficient TCAM, called TCA*m*M$^{\text{CogniGron}}$. Building on PA*m*M and TCA*m*M$^{\text{CogniGron}}$, we propose a novel network switching architecture and analyze its energy efficiency over the experimental dataset of a Nb-doped SrTiO_3 memristive device. The results show that the proposed network switching architecture consumes only 0.01 fJ/bit/cell energy for analog compute operations which is at least 50 times less than the digital operations.

CCS CONCEPTS

• **Networks → In-network processing**; **Middle boxes / network appliances**; • **Hardware → Emerging architectures**; **Memory and dense storage**; **Impact on the environment**.

KEYWORDS

Memristors, TCAM, Switches, Cognitive models, Energy efficiency

ACM Reference Format:
Saad Saleh and Boris Koldehofe. 2023. Memristor-based Network Switching Architecture for Energy Efficient Cognitive Computational Models. In *18th ACM International Symposium on Nanoscale Architectures (NANOARCH '23), December 18–20, 2023, Dresden, Germany*. ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/3611315.3633272

NANOARCH '23, December 18–20, 2023, Dresden, Germany

1 INTRODUCTION

The Internet heavily relies on specialized network devices, called switches, in order to establish the communication links between senders and receivers of network traffic [2, 3]. So far, the design of network switches is dominated by the use of transistor-based Ternary Content-Addressable Memory (TCAM). TCAM enables processing operations, like matching incoming packet header fields against stored rules, in one clock cycle. It stores data in the form of ternary digits i.e. high bit (1), low bit (0), and don't care bit (X). TCAM output is high only if the stored bit is X or the same as the incoming bit. Although TCAM-based switches provide remarkable performance from 12.8-51.2 Tbps [8], they have two major limitations. Firstly, the increasing performance requirements consume a significant amount of energy [8]. Secondly, more expressive functions, like cognitive functions using neuromorphic computations, require the use of a range of inputs/outputs (analog match signals) but TCAM relies on digital inputs/outputs. The limitations of TCAM can be bypassed by studying novel hardware components, like *Memristors*, that can support analog computations.

Memristors are programmable hardware components which offer promising energy efficient characteristics like non-volatility, nanoscale size, and analog storage/processing [8]. The energy efficient properties of memristors motivate the development of alternative designs of memory architectures to counteract or even overcome the energy and performance limitations of current network switches. However, it requires an understanding of integrating memristors in the network switching architectures by developing specialized memory architectures. Building on prior findings [4–6, 9], we present a novel network switching architecture for supporting energy efficient cognitive computations. The proposed architecture makes use of both analog and digital computations built over a specialized memory called memristor-based Probabilistic Associative Memory (PA*m*M) [6]. PA*m*M provides both digital (deterministic) and analog (probabilistic) outputs based on the incoming data. If the input completely matches (or mismatches) to the stored data, PA*m*M provides a deterministic output of 1 (or 0). If the input partially matches to the stored data, PA*m*M provides a probabilistic analog output in between 0 and 1 depending upon the difference of the input with the stored data. However, the use of probabilistic outputs comes at the cost of hardware precision due to the interference from neighboring components and decrease in signal strength. For high precision, the proposed architecture uses an energy efficient memristor-based TCAM called TCA*m*M$^{\text{CogniGron}}$ [4][5]. TCA*m*M$^{\text{CogniGron}}$ uses two memristive states due to its digital operations, while PA*m*M uses the analog memristive states for analog outputs. We analyze the performance

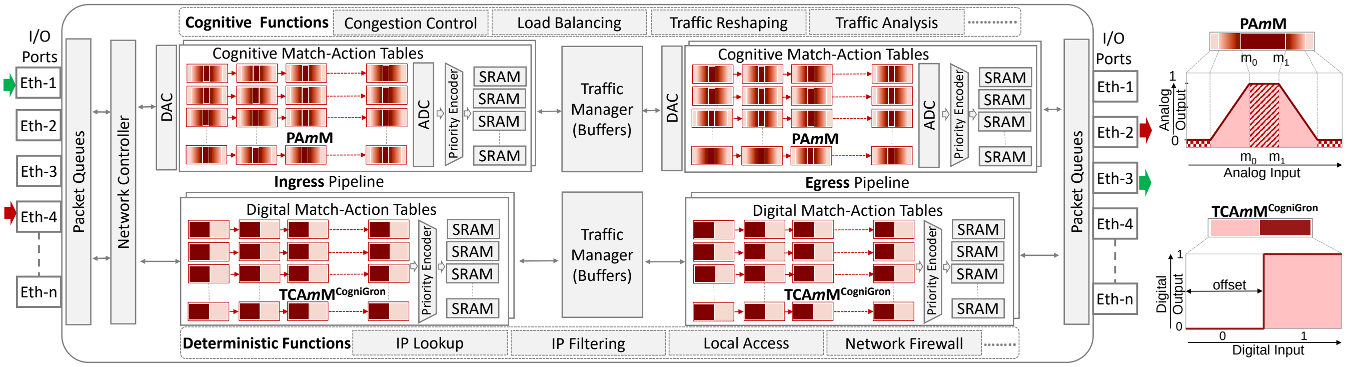


Figure 1: The proposed memristor-based network switching architecture built over PA*m*M and TCA*m*M$^{\text{CogniGron}}$.

of our proposed architecture over the dataset of a physically fabricated Nb-doped SrTiO$_3$ (Nb:STO) memristive device.

Contributions and Research Findings. In this paper, we propose a novel network switching architecture building over memristors for supporting energy efficient cognitive computational models. Our major contributions are three-fold; (1) Development of a novel network switching architecture supporting both digital and analog computations, (2) Analysis of the processing operations and system architecture for the proposed network switching architecture, (3) Analysis of the energy consumption of analog and digital processing for the Nb:STO-based memristive device. The research showed that the analog processing operations consume only 0.01 fJ/bit/cell energy for Nb:STO-based memristive device. The digital computations consume 1-16 fJ/bit/cell energy depending upon the processing operations i.e., match and mismatch operations. Moreover, the nonvolatile properties of memristors (Nb:STO) reduce the standby power consumption to zero.

2 PROPOSED MEMRISTOR-BASED SWITCH

The proposed memristor-based switching architecture is shown in Fig. 1. Its major modules include the PA*m*M and TCA*m*M$^{\text{CogniGron}}$ in the processing stages. The accompanying modules include the network controller, packet queues, traffic managers, and I/O ports.

2.1 PA*m*M

PA*m*M stores the network rules in the form of programmable analog states of memristors i.e., resistance. Its function is to compute the difference between the input and the stored contents. If the input lies within the programmed thresholds, the output is a deterministic high. Otherwise, the output is analog (probabilistic) ranging in between 0 and 1. Fig. 1 shows the abstract working operation of PA*m*M. The rules can be stored by programming the parameters m_0 and m_1. Any input in the programmed range (m_0 to m_1) will yield a deterministic high output 1. The inputs lying outside the programmed range are analog (0-1) depending upon the difference between the applied input and the stored contents.

The circuit for PA*m*M contains two memristors (M_0 and M_1), two resistors (R_X), and five transistors, as shown in Fig. 2a. PA*m*M operates in the analog domain. During the write operation, PA*m*M programs memristor M_1 such that the voltage V_A is equal to m_1. In the second step, PA*m*M programs memristor M_0 such that the voltage V_B equals m_0. The relationship of voltages V_A and V_B with the circuit configurations is expressed by Eq. 1 and Eq. 2, respectively. V_{DD} is the read voltage and V_{IN} is the applied input voltage.

$$V_A = \frac{R_X}{M_1 + R_X}(V_{DD} - V_{IN}) + V_{IN} \tag{1}$$

$$V_B = \frac{R_X}{M_0 + R_X}(V_{DD} - V_{IN}) + V_{IN} \tag{2}$$

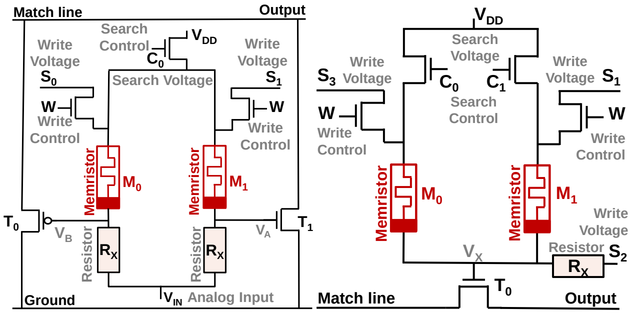
During the search operation, the control line (C_0) is enabled and the write line (W) is disabled. The incoming query is applied as V_{IN} by converting from digital to analog signals through Analog-to-Digital Converters (ADCs). If the developed voltage V_A is greater than the threshold voltage of the transistor T_1, the output drops. The rate of drop of output voltage depends upon the voltage V_A and the transistor's operating curves. Similarly, if the developed voltage V_B is less than the threshold voltage of transistor T_0, the output drops.

The given PA*m*M design, adapted from [6], applies the input to both memristors simultaneously. As Nb:STO operates at low voltages, it does not need any resistors with transistors T_0 and T_1.

2.2 TCA*m*M$^{\text{CogniGron}}$

TCA*m*M$^{\text{CogniGron}}$ stores network rules in form of digital data by using only binary memristive states. The role of TCA*m*M$^{\text{CogniGron}}$ is to output a digital high only if the stored data contains a don't care bit X or the same data as the incoming bit. Fig. 1 shows the abstract working operation of TCA*m*M$^{\text{CogniGron}}$ cell. The programmability of TCA*m*M$^{\text{CogniGron}}$ consists of the specification of the stored bits i.e., low bit 0, high bit 1, or don't care bit X.

The circuit for TCA*m*M$^{\text{CogniGron}}$ contains two memristors (M_0 and M_1), one resistor (R_X), and five transistors to perform the write and search operations, as shown in Fig. 2b [4]. During the write operation, TCA*m*M$^{\text{CogniGron}}$ programs the memristors M_0 and M_1 in high and low resistance states for programming a high bit, and vice versa. During the search operation, one of the two control lines (C_0 or C_1) is enabled based on the search query. The output is high (match) only if V_X is greater than the threshold voltage of the output transistor (T_0). If a don't care bit is stored, both memristors are in low resistance states and the output is also high (match). The



(a) PA*m*M (adapted from [6]). (b) TCA*m*M$^{\text{CogniGron}}$ [4].

Figure 2: The circuits for PA*m*M and TCA*m*M$^{\text{CogniGron}}$.

developed voltage V_X and its relationship with the programmed memristances is determined by Eq. 3 and Eq. 4, respectively. V_{DD} is the applied read voltage.

$$V_X(C_0 = 1) = \frac{R_X}{M_0 + R_X} V_{DD} \tag{3}$$

$$V_X(C_1 = 1) = \frac{R_X}{M_1 + R_X} V_{DD} \tag{4}$$

2.3 Network controller and associated modules

The modules accompanying the PA*m*M and TCA*m*M$^{\text{CogniGron}}$ include the network controller, packet queues, traffic manager, ADCs, Digital-to-Analog Converters (DACs), Priority Encoder, Static Random Access Memory (SRAM), and I/O ports, as described below.

Network Controller. The controller extracts the required header fields from the incoming packets. The header fields are forwarded to either PA*m*M or TCA*m*M$^{\text{CogniGron}}$-based packet processing stages. The network functions requiring cognitive models through analog processing are forwarded to the PA*m*M-based packet processing. Some examples of these network functions include congestion control, load balancing, etc. However, functions requiring high precision and deterministic computations are forwarded to the digital packet processing stages. Some examples of high precision network functions include IP lookup, traffic analysis [7, 10], etc.

Packet Queues. As the traffic transmission and processing rate varies, it leads to delays, jitters, and irregular traffic patterns at network switches. To cater to the varying rates, the input and output side contain the packet queues for the temporary storage of packets.

Traffic Manager. If the packet arrival rate is greater than the packet processing and transmission rate, the network is quickly congested. To avoid this issue, the network switches use Active Queue Management (AQM) techniques to selectively drop the packets based on the network congestion. Traffic manager implements the AQM techniques for minimizing network congestion.

ADC and DAC. The operation of PA*m*M in the analog domain requires the use of DAC for converting the digital signals to the analog domain. Moreover, the final processed output is converted back to the digital domain by using ADC.

Priority Encoder. The PA*m*M and TCA*m*M$^{\text{CogniGron}}$-based packet processing matches the incoming query against multiple parallel programmed rules. In case of multiple successful matches, the architecture uses the priority encoder for selecting the highest priority match among multiple parallel occurring matches. The rules are programmed in descending order of priority so that the priority encoder can select the match at the highest location.

Table 1: Minimum energy consumption for digital and analog processing.

Technology	Architecture	Energy Consumption
Analog processing	PA*m*M	0.01 fJ/bit/cell
Digital processing	TCA*m*M$^{\text{CogniGron}}$	1-16 fJ/bit/cell

SRAM. The actions for corresponding successful matches are stored in the SRAM-based cells. The SRAM cells can use either the transistors or memristors in their cell design.

I/O Ports. The input and output links are connected to the respective I/O ports based on the ethernet connections. For increased capacity, optical I/O ports can also be used.

3 PERFORMANCE ANALYSIS

We analyze the performance of the proposed switching architecture over an experimental dataset of Nb:STO [1]. The STO exhibits memristive behavior at the Schottky interface of the metal electrode and substrate. The performance analysis of PA*m*M and TCA*m*M$^{\text{CogniGron}}$-based packet processing stages is shown below;

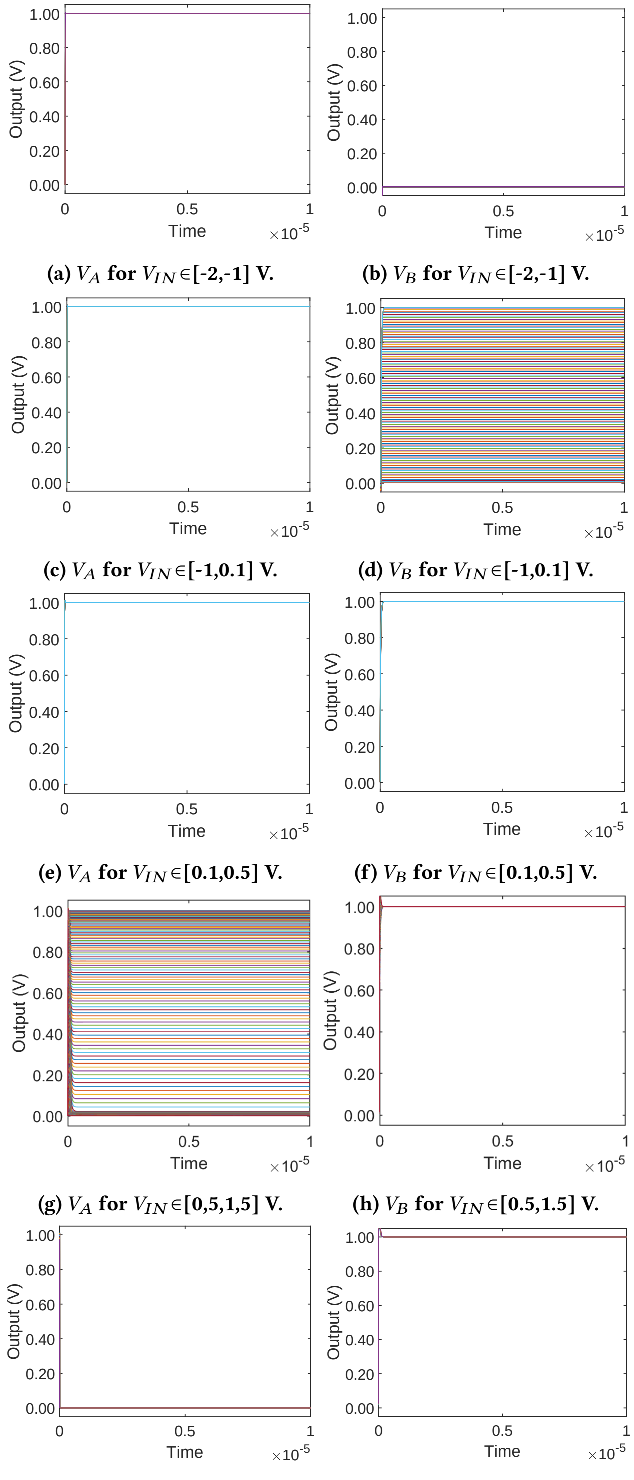
3.1 PA*m*M

The energy consumption of PA*m*M was analyzed by varying the applied input and the programmable resistance of the memristor. The analysis showed that the maximum energy consumption of Nb:STO-based PA*m*M is around 0.16 nJ/bit/cell. However, programming the Nb:STO to high resistance states can provide an operating range with much lower energy consumption. The lowest energy consumption of Nb:STO-based PA*m*M is around 0.01 fJ/bit/cell which is at least 50 times better than the state-of-the-art digital computations (Tab. 1) [4]. Moreover, PA*m*M has zero power consumption in the standby mode due to the non-volatility of Nb:STO.

The analog processing operation of the PA*m*M for various analog inputs is shown in Fig. 3. In the given case, the programmed rule is stored in the range of [0.1, 0.5] V. If the input lies in the range of [-2, -1] V, voltage V_B drops the output voltage to zero. For an input within the range of [-1, 0.1] V, voltage V_B drops the outputs to an analog voltage (0-1). Similarly, the output is either 0 or in between 0-1 for input ranges of [1.5, 2.5] V and [0.5, 1.5] V, respectively. The match line will be high only if both V_A and V_B are high.

3.2 TCA*m*M$^{\text{CogniGron}}$

The energy consumption of TCA*m*M$^{\text{CogniGron}}$ depends on the programmed memristor state. The minimum energy consumption of TCA*m*M$^{\text{CogniGron}}$ for the Nb:STO memristor is around 1 fJ/bit/cell for the highest resistance state. The major reason for the high energy consumption, as compared to PA*m*M, is the use of binary memristive states. Moreover, the high threshold voltage for the output transistor in TCA*m*M$^{\text{CogniGron}}$ also increases the energy consumption. The mismatch and match operations use different resistance states and the power consumption varies from 1-16 fJ/bit/cell based upon the operation. Similar to PA*m*M, TCA*m*M$^{\text{CogniGron}}$ consumes



(a) V_A **for** V_{IN}∈**[-2,-1] V.**

(b) V_B **for** V_{IN}∈**[-2,-1] V.**

(c) V_A **for** V_{IN}∈**[-1,0.1] V.**

(d) V_B **for** V_{IN}∈**[-1,0.1] V.**

(e) V_A **for** V_{IN}∈**[0.1,0.5] V.**

(f) V_B **for** V_{IN}∈**[0.1,0.5] V.**

(g) V_A **for** V_{IN}∈**[0,5,1,5] V.**

(h) V_B **for** V_{IN}∈**[0.5,1.5] V.**

(i) V_A **for** V_{IN}∈**[1.5,2.5] V.**

(j) V_B **for** V_{IN}∈**[1.5,2.5] V.**

Figure 3: The input-output response of PA*m*M-based analog processing by variation in input voltage V_{IN}.

zero power in the standby mode due to the use of nonvolatile Nb:STO. Detailed analysis of TCA*m*M$^{\text{CogniGron}}$ is present in [4].

4 CONCLUSION AND FUTURE WORK

In this paper, we proposed a novel network switching architecture for supporting digital and analog computations. The proposed architecture makes use of a specialized memory, PA*m*M, for providing deterministic and probabilistic outputs. Considering the requirements of high precision network functions, the architecture also uses a digital memory, TCA*m*M$^{\text{CogniGron}}$, for supporting traditional digital operations. The performance analysis over an experimental dataset of Nb:STO memristor showed that PA*m*M-based analog processing consumes only 0.01 fJ/bit/cell energy. The traditional digital operations require 1-16 fJ/bit/cell energy based on the match and mismatch operations. In the future, we will focus on the understanding of precision and accuracy of line-rate network functions. Moreover, we will study the implementation of learning systems inside memristor-based network switches.

ACKNOWLEDGMENTS

The authors would like to acknowledge the financial support of the CogniGron research center and the Ubbo Emmius Funds (University of Groningen). Special thanks to Anouk S. Goossens and Tamalika Banerjee for the provision of the experimental dataset of Nb:STO.

REFERENCES

[1] Anouk S. Goossens and Tamalika Banerjee. 2023. Tunability of Voltage Pulse Mediated Memristive Functionality by Varying Doping Concentration in $SrTiO_3$. *Applied Physics Letters* 122, 3 (2023), 034101. https://doi.org/10.1063/5.0124135

[2] Azeem Iqbal, Uzzam Javed, Saad Saleh, Jongwon Kim, Jalal S Alowibdi, and Muhammad U Ilyas. 2016. Analytical Modeling of End-to-End Delay in OpenFlow Based Networks. *IEEE Access* 5 (2016), 6859–6871. https://doi.org/10.1109/access.2016.2636247

[3] Uzzam Javed, Azeem Iqbal, Saad Saleh, Syed Ali Haider, and Muhammad U Ilyas. 2017. A Stochastic Model for Transit Latency in OpenFlow SDNs. *Elsevier Computer Networks* 113 (2017), 218–229. https://doi.org/10.1016/j.comnet.2016.12.015

[4] Saad Saleh, Anouk S. Goossens, Tamalika Banerjee, and Boris Koldehofe. 2022. TCA*m*M$^{\text{CogniGron}}$: Energy Efficient Memristor-Based TCAM for Match-Action Processing. In *Proceedings of the International Conference on Rebooting Computing.* IEEE, 89–99. https://doi.org/10.1109/ICRC57508.2022.00013

[5] Saad Saleh, Anouk S. Goossens, Tamalika Banerjee, and Boris Koldehofe. 2022. Towards Energy Efficient Memristor-based TCAM for Match-Action Processing. In *Proceedings of the International Green and Sustainable Computing Conference.* IEEE, 1–4. https://doi.org/10.1109/IGSC55832.2022.9969354

[6] Saad Saleh, Anouk S. Goossens, Tamalika Banerjee, and Boris Koldehofe. 2023. PA*m*M: Memristor-based Probabilistic Associative Memory for Neuromorphic Network Functions. In *Proceedings of the Non-Volatile Memory Technology Symposium (NVMTS).* IEEE, 1–5. In Press.

[7] Saad Saleh, Muhammad U Ilyas, Khawar Khurshid, Alex X Liu, and Hayder Radha. 2015. IM Session Identification by Outlier Detection in Cross-correlation Functions. In *Proceedings of the Annual Conference on Information Sciences and Systems.* IEEE, 1–5. https://doi.org/10.1109/ciss.2015.7086851

[8] Saad Saleh and Boris Koldehofe. 2022. On Memristors for Enabling Energy Efficient and Enhanced Cognitive Network Functions. *IEEE Access* 10 (2022), 129279–129312. https://doi.org/10.1109/access.2022.3226447

[9] Saad Saleh and Boris Koldehofe. 2023. The Future is Analog: Energy-Efficient Cognitive Network Functions over Memristor-Based Analog Computations. In *Proceedings of the Workshop on Hot Topics in Networks.* ACM, 1–9. https://doi.org/10.1145/3626111.3628192

[10] Saad Saleh, Mamoon Raja, Muhammad Shahnawaz, Muhammad U Ilyas, Khawar Khurshid, M Zubair Shafiq, Alex X Liu, Hayder Radha, and Shirish S Karande. 2014. Breaching IM Session Privacy Using Causality. In *Proceedings of the Global Communications Conference.* IEEE, 686–691. https://doi.org/10.1109/glocom.2014.7036887