Lawrence Livermore National Laboratory

FPGA based Network Traffic Analysis using Traffic Dispersion Graphs

2nd September, 2010

Faisal N. Khan

Lawrence Livermore National Laboratory, P. O. Box 808, Livermore, CA 94551

This work performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344

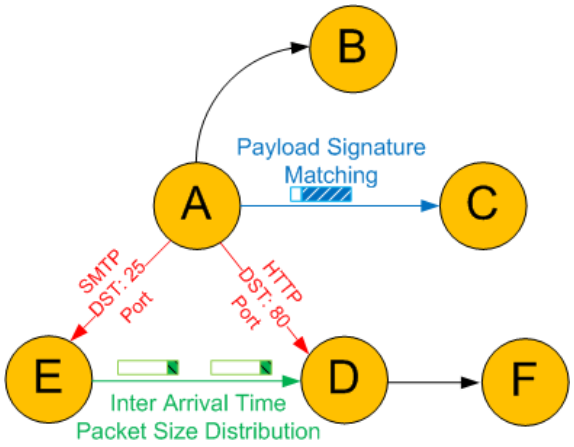
Release # LLNL-PRES-450851

Outline

- Introduction and Motivation
- FPGA Architecture & Algorithm
- Experimental Results
- Conclusions and Future Work

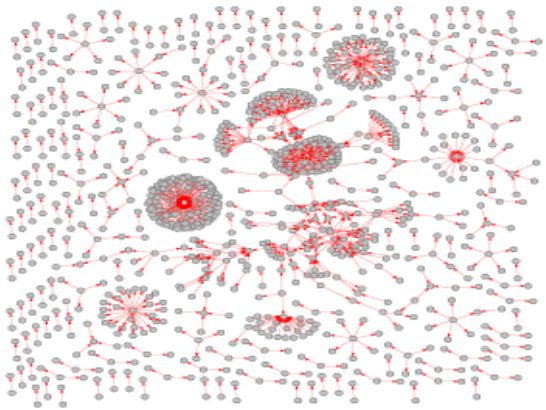
Network Traffic Analysis

- Network traffic analysis & classification is keystone in wide range of applications such as detection of network anomalies and attacks, identification of new applications, and traffic engineering.
  - Traditionally it has been done using
    - TCP destination port
    - Payload signature matching
    - Behavioral characterization of a host and its flows

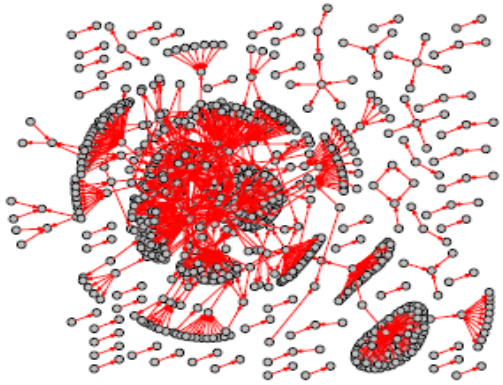


Traffic Dispersion Patterns

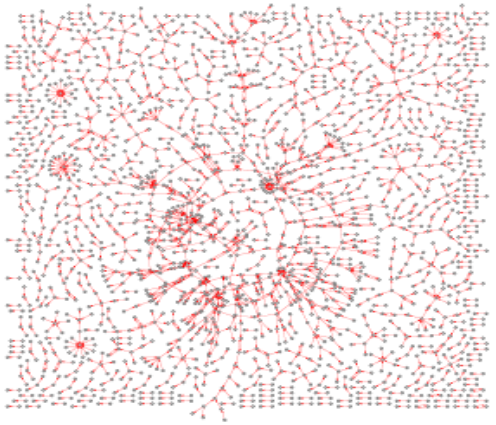
- Traffic Dispersion Patterns/Graphs (TDPs/TDGs) are a network wide extension of behavioral techniques [Iliofotou-07].
- Communication patterns between the hosts exhibit insights into different kinds of applications and anomalies in a network.



HTTP



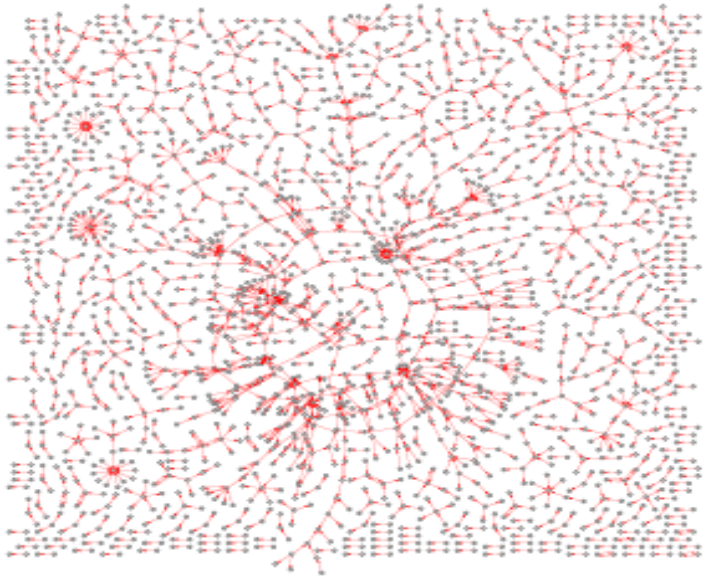
SMTP



WinMX (Peer-to-Peer)

Applications

- Isolating anomalous hosts and sub-networks
- Detecting the spread of worm and viruses.
- Characterizing unknown activity/application.
- Flagging suspicious activity without examining the content.



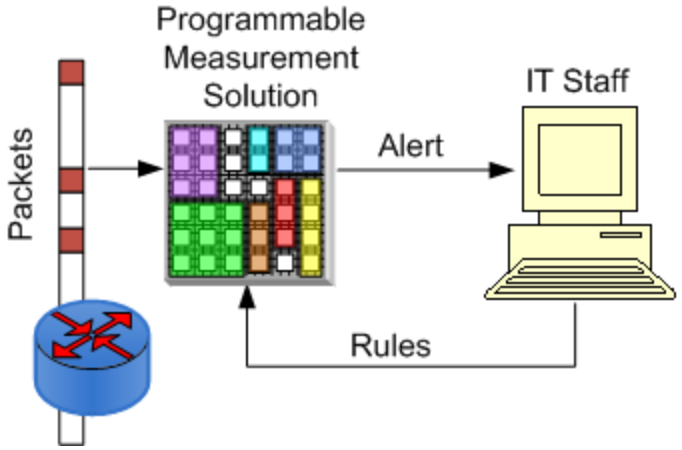
WinMX (Peer-to-Peer)

Motivation

- TDPs are traditionally processed offline (non-streaming)
  - Full view of the graph: Complete flexibility
  - Slow and resource intensive
- Real-time threats get missed out.
- The question of processing TDPs in real-time has mostly been overlooked.

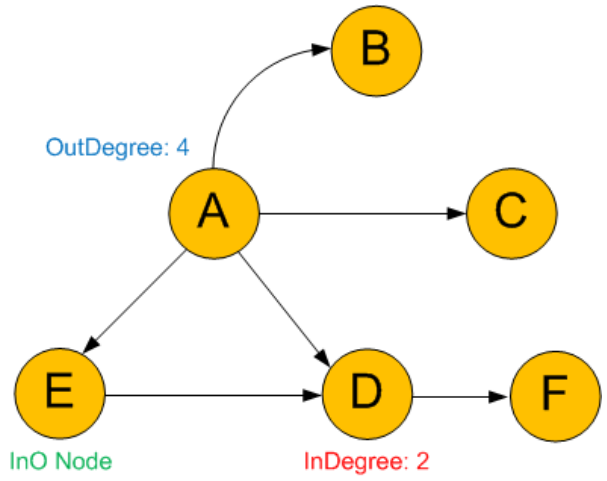
Goals

- An autonomous passive system that can detect anomalous behaviors in streaming traffic in real-time.
- Isolates or alarms the IT staff about the threats.
- Programmable to varying network monitoring needs.
- Scalable to millions of edges and new traffic features.
- Has a defined accuracy budget.



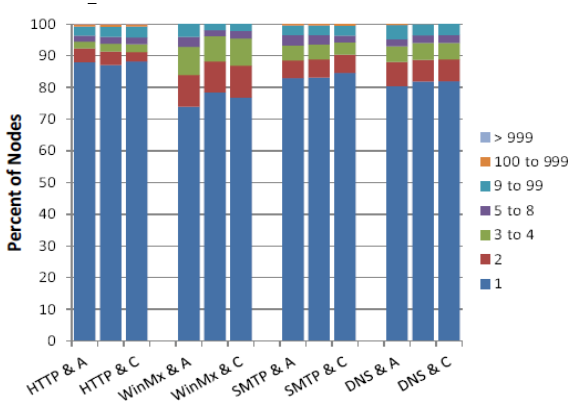
Quantizing Traffic Dispersion Patterns

- Dispersion patterns can be classified as
  - Offline: requires knowledge of complete graph. For e.g. depth of communication
  - Online: can be computed in streaming traffic. For e.g. connectivity of an IP address
- InDegree/OutDegree: The number of incoming/outgoing connections made to/from an IP address/Node.
- InO: The number of hosts that have both incoming and outgoing connections.

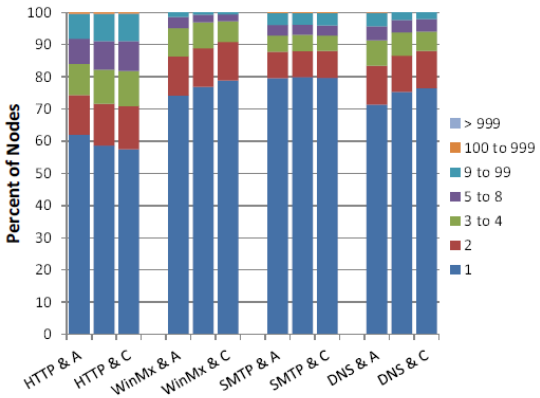


Defining Rules

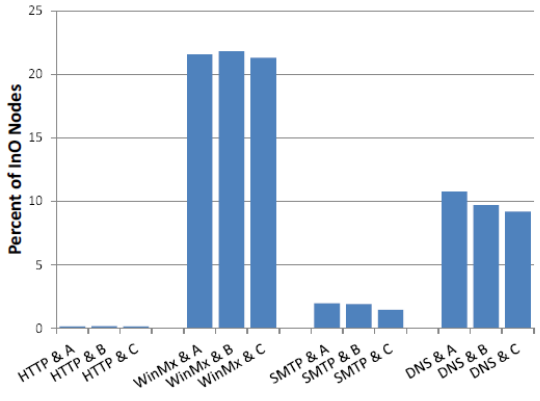
- Simulated backbone CAIDA traces having known P2P activity.



InDegree Distributions



OutDegree Distributions

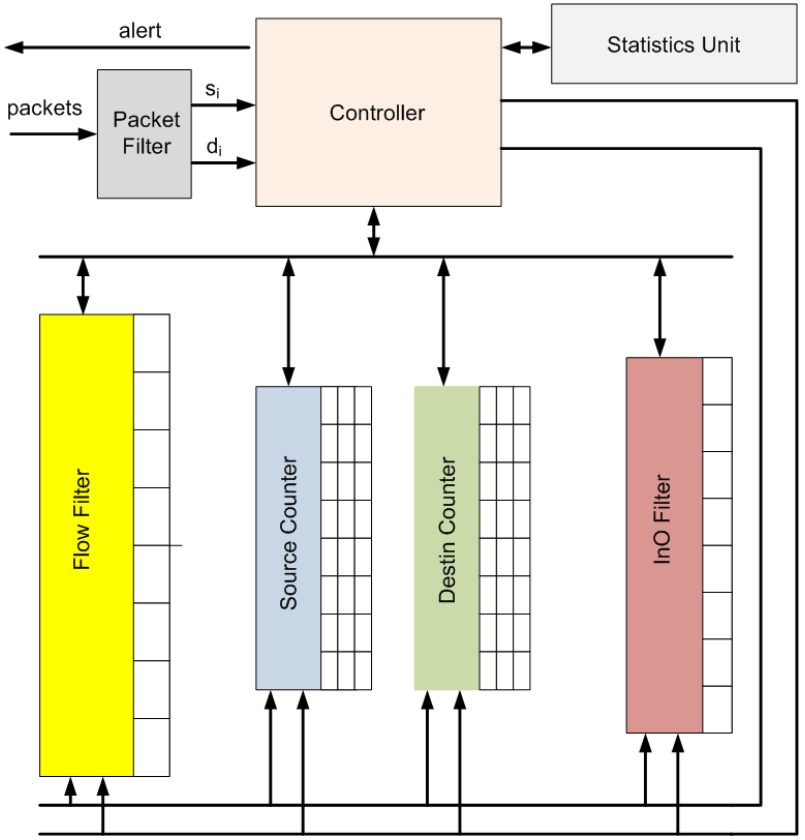


InO Distributions

- P2P Detection Rule: A high InO with a low InDegree and OutDegree
- Port-Scan Rule: High InDegree and/or High OutDegree

The Measurement Solution

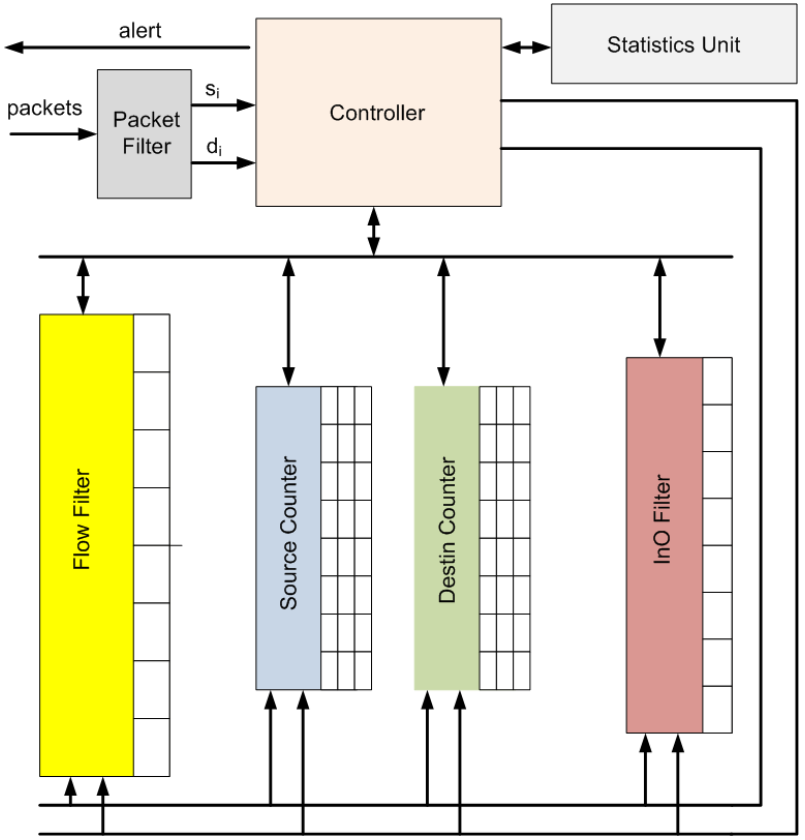
- Mapped on FPGAs that are an ideal platform for the high programmability and throughput requirements of the application.
- Configurable packet filter.
- Bloom Filters to check presence of an incoming IP address.
- Bloom Counters to keep the InDegree and OutDegreees.
- Count values and programmable rules in Statistics Unit.



FPGA Mapped Architecture

The Measurement Algorithm

- Incoming packet checked for uniqueness in the Flow-filter.
- The source-IP and destination-IP are used to update the degree-counts.
- Statistics unit is updated.
- The IPs are reverse checked for deducing InO behavior.
- A found InO IP is checked in InO-Filter and InO count updated

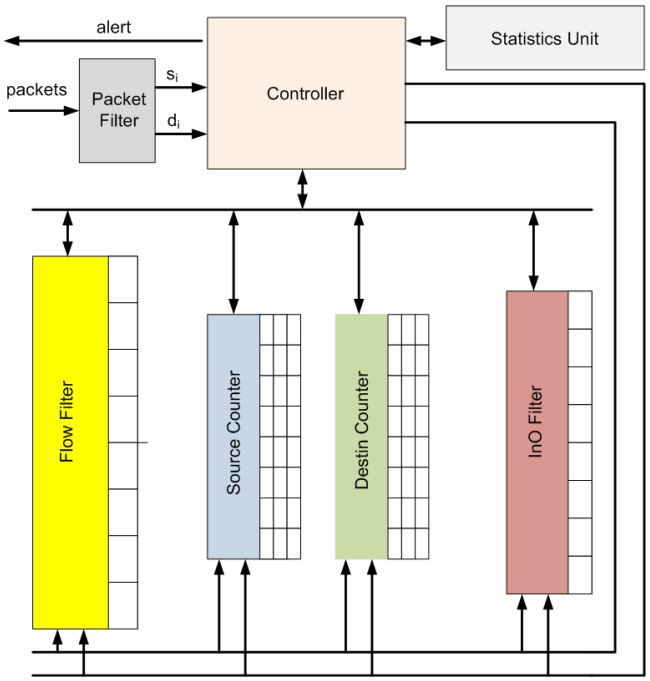


FPGA Mapped Architecture

Design Space Exploration

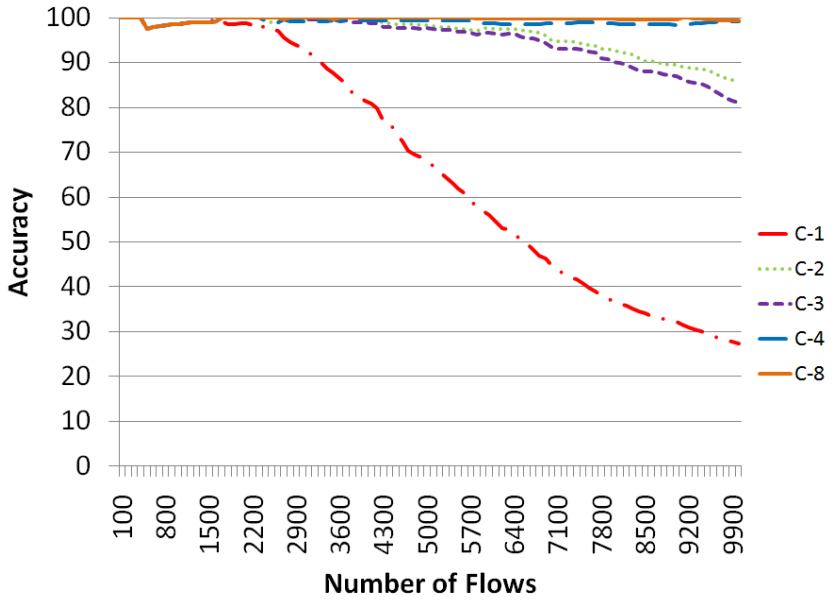
Config	Flow x8	InO x8	Degree x2x8x8	Total (Kb)	Node Count Accuracy	InO Count Accuracy
C-1	16	16	2	512	71.78	27.35
C-2	8	8	4	640	88.31	85.75
C-3	16	8	4	704	92.14	81.07
C-4	8	8	8	1152	90.13	99.18
C-8	32	1	8	1312	96.95	99.34
C-9	32	16	16	2432	99.98	99.67

Architectural Exploration (values in Kb)

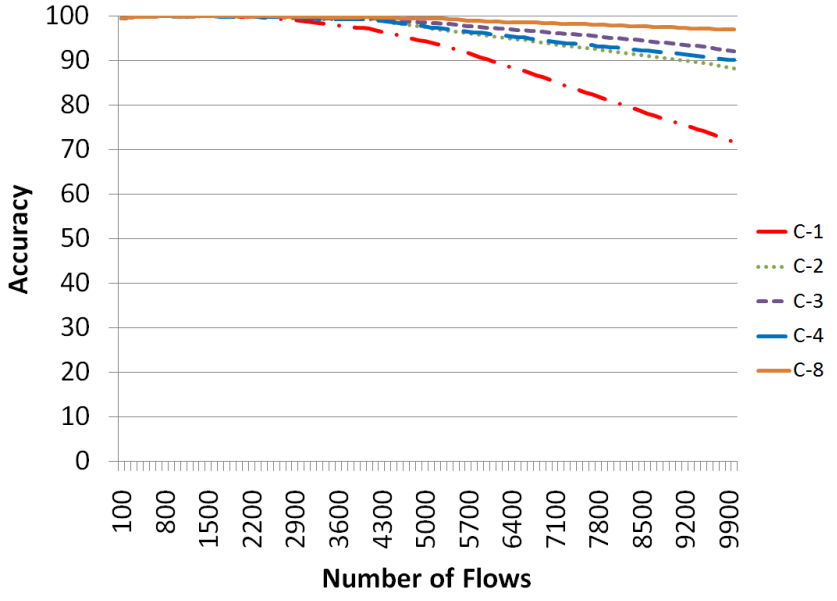


FPGA Mapped Architecture

Design Accuracy



Node Count Accuracy



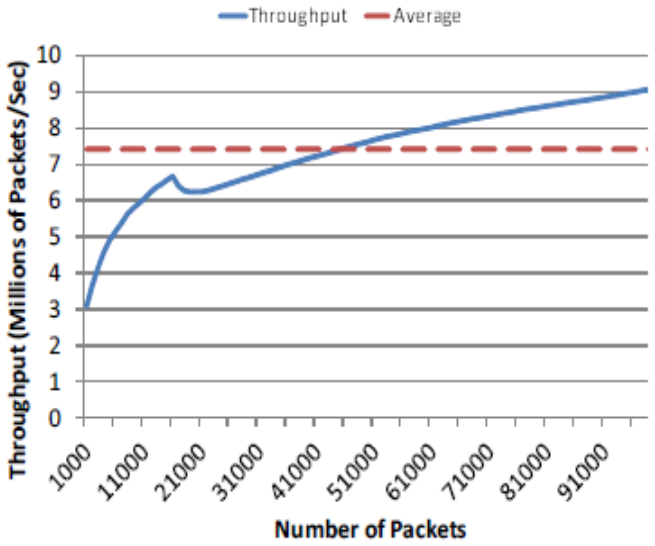
InO Count Accuracy

- The work utilized datasets from CAIDA, WIDE and ISI/USC for empirical evaluations.
- The work discusses analytical and empirical accuracy studies.
- Results show very high amounts of accuracies for up to 10k flows.
- Suitable for detecting threats like Port-Scan.

Implementation Results & System Throughput

- Prototyped C-8 on Virtex-5, XC5VLX50t device.
- Xilinx ISE 11.1
- Ample room in the device for future logic enhancements.
- Average throughput of 7.5 million packets/second.

Metric	Value	Device Utilization
IO Pins	69	14.4%
Number of occupied slices	754	10%
Number of Block-RAMs	44	73%
Clock (MHz)	107	-



Port-Scan Detection Accuracy

- Very high detection accuracy of both scanners and victim identification.

Observed

Actual: Scanners	P	N
P’	4	0
N’	0	33972

Scanner Identification

F1 Score: 1.0

Observed

Actual: Victims	P	N
P’	22	0
N’	2	2303

Scanned Victim Identification

F1 Score: 0.932

- Used USC-LANDER [USCISI-06] and CAIDA traces [CAIDA-03].

Peer-to-Peer Detection Accuracy

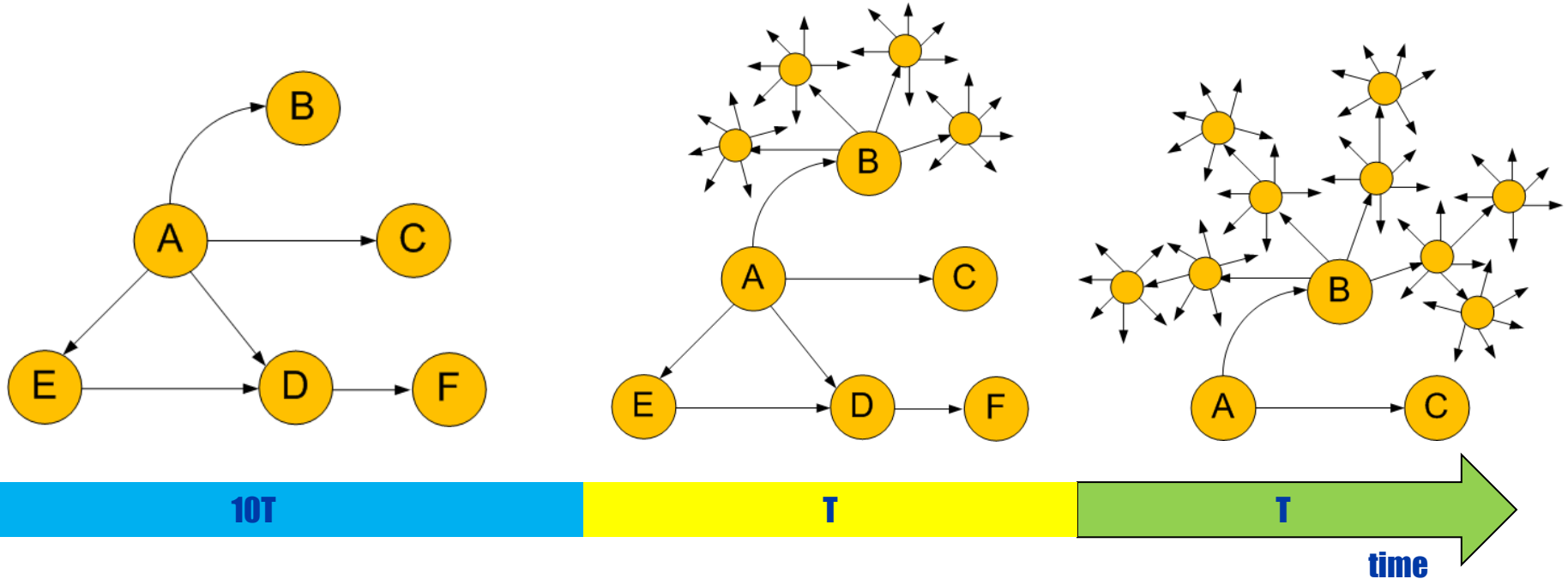
- Programmed Rule: 10% of InO Nodes and Low InDegree and OutDegrees.
- Port based filtering.
- Alarm raised in less than 1400 flows at the correct port (port-6677)
- 1400 flows represent a very high system accuracy design point.
- Useful for classifying P2P activity on a new port.

Conclusion

- A real-time FPGA based solution for processing streaming TDPs.
- Configurable with high processing speeds.
- Real-time detection of Port-Scan and Peer-to-Peer activities.
- Reducing noise in capturing interesting traffic.
- Easily extensible to include temporal rules and other characterization measures.
- Open Source under GNU General Public License: https://computation.llnl.gov/casc/dcca-pub/dcca/Data-centric_architecture.html

Future Directions

- Including temporal relations in rule sets
- Isolating sudden changes from gradual shifts
- For e.g. A new Worm, Spammer, etc



References

- **[Iliofotou-07]** M. Iliofotou, P. Pappu, M. Faloutsos, M. Mitzenmacher, S. Singh, and G. Varghese, “Network monitoring using traffic dispersion graphs (TDGS),” in *Proceedings of the 7th ACM SIGCOMM Internet Measurement Conference, 2007, pp. 315–320.*
- **[USCISI-06]**Internet Addresses Survey dataset, PREDICT ID: USC-LANDER/attack-tcpsyn-20061106. Traces taken 2006-11-06. Provided by the USC/LANDER project <http://www.isi.edu/ant/lander>.
- **[CAIDA-03]** CAIDA OC48 Trace Project 20030115-100000-0-anon.pcap https://imdc.datcat.org/data/1-3N3T-4=20030115-100000-0-anon.pcap+(60+min)