

2022-05-25

## Cross-Atlantic Experiments on EU-US Test-beds

Sachin Sharma

*Technological University Dublin, sachin.sharma@tudublin.ie*

Avishek Nag

*University College Dublin*

Byrav Ramamurthy

*University of Nebraska-Lincoln*

Follow this and additional works at: <https://arrow.tudublin.ie/aaconmusart>



Part of the [Digital Communications and Networking Commons](#)

---

### Recommended Citation

S. Sharma, A. Nag and B. Ramamurthy, "Cross-Atlantic Experiments on EU-US Test-beds," in IEEE Networking Letters, doi: 10.1109/LNET.2022.3177712.

This Article is brought to you for free and open access by the Conservatory of Music and Drama at ARROW@TU Dublin. It has been accepted for inclusion in Articles by an authorized administrator of ARROW@TU Dublin. For more information, please contact [arrow.admin@tudublin.ie](mailto:arrow.admin@tudublin.ie), [aisling.coyne@tudublin.ie](mailto:aisling.coyne@tudublin.ie), [gerard.connolly@tudublin.ie](mailto:gerard.connolly@tudublin.ie).



This work is licensed under a [Creative Commons Attribution-NonCommercial-Share Alike 4.0 License](#)  
Funder: NGIAtlantic H2020 project under agreement no. OC3-292

# Cross-Atlantic Experiments on EU-US Test-beds

Sachin Sharma<sup>1</sup>, Avishek Nag<sup>2</sup> and Byrav Ramamurthy<sup>3</sup>

Technological University Dublin<sup>1</sup>, University College Dublin<sup>2</sup>, University of Nebraska-Lincoln<sup>3</sup>

E.mail: Sachin.Sharma@TUDublin.ie

**Abstract**—Today, there are a number of real testbeds world-wide among which Fed4Fire testbeds are prominent in the EU, while POWDER and COSMOS are prominent in the US. This paper aims to validate inter-testbed experiments between the EU and the US by connecting a number of Fed4Fire and US testbeds as part of an NGIAtlantic project. The goal is to compare the hop count, the topology formed, the maximum bandwidth permitted, and the loss and jitter that occurred between different testbeds. Additionally, Software Defined Networking (SDN) experiments between EU and US testbeds are conducted, and an edge-computing use case is developed and tested.

## I. INTRODUCTION

Simulations are valuable in analyzing new protocols and ideas in computer networking as they mimic the testing environment closely. However, when a protocol/system is subjected to unexpected events and dynamic environmental constraints, testbeds offer a higher degree of confidence compared to simulations. Further, testbeds provide realism to the evaluation of a protocol/system. Currently, several testbeds such as GENI [1], Emulab [2], PlanetLab [3], Fed4Fire [4], ORBIT [5], POWDER [6], and COSMOS [7] are created for researchers or companies to test their implemented protocols.

Typically, these testbeds are open to researchers from industry and academia for the purpose of testing prototypes without investing in testing infrastructure. However, these testbeds have limited resources causing long waiting times for their approval and making the experiment size limited. Inter-testbed experiments can solve this problem by requesting resources from multiple testbeds, which can collectively act as a large resource pool. We, therefore, conduct inter-testbed experiments between the US and the EU testbeds. Moreover, such an experiment is necessary to implement edge computing use cases, where physical distance plays a role [8].

| testbed      | Resources              | Networks           | Location           |
|--------------|------------------------|--------------------|--------------------|
| Virtual Wall | Bare-Metal/<br>Virtual | Wired              | Ghent, EU          |
| w-ilab1.t    | Bare-Metal             | Wireless/IoT       | Ghent, EU          |
| w-ilab2.t    | Bare-Metal             | Wireless/IoT       | Ghent, EU          |
| CityLab      | Bare-Metal             | Wireless/IoT       | Antwerp, EU        |
| POWDER       | Bare-Metal/<br>Virtual | Wireless/<br>Wired | Salt Lake City, US |

TABLE I. Testbeds for our inter-testbed study

For our inter-testbed experiments, we chose the IMEC virtual wall [9], w-ilab1.t, w-ilab2.t [10], and CityLab testbeds [11] from the EU and the POWDER testbed [12] from the US. These testbeds are chosen as part of an NGIAtlantic H2020 Open Call project [13] to experiment with Software Defined Networking (SDN) using a mix of wireless and wired testbeds. Table I lists these testbeds according to the resources available

(bare-metal or virtual), the networks that can be created, and their locations. The term ‘bare-metal’ refers to a single-user machine whose resources are not shared. In contrast, a virtual resource can be shared by several users.

The objectives of this paper include:

- 1) Determine feasibility of EU-US inter-testbed activities.
- 2) Connecting different experiments deployed on geographically apart testbeds with public IP addresses and investigating hops between testbeds, round-trip time, jitter, loss, and the maximum permissible UDP and TCP bandwidths.
- 3) Identify the topology over which different testbeds are connected.
- 4) Compare the performance of an SDN experiment performed between different testbeds.
- 5) Implement an edge-computing use case and test it to show the benefits of an inter-testbed experiment.

## II. INTER-TESTBED BENCHMARK EXPERIMENT

| Testbed      | Name     | Location                      | CPU                                  | Memory |
|--------------|----------|-------------------------------|--------------------------------------|--------|
| CityLab      | Node 6   | Middelheimlaa,<br>Antwerpen   | AMD GX-412TC 1GHz<br>Quad-core       | 4GB    |
|              | Node 18  | Vekestraat, 2000<br>Antwerpen |                                      |        |
| W-ilab1.t    | nuc03    | Zwijnaarde, Ghent             | Intel i5-4250U                       | 8GB    |
|              | nuc04    |                               | 1.3GHz Quad-core                     |        |
| W-ilab2.t    | nucX3    | Zwijnaarde, Ghent             | Intel i5-4250U                       | 8GB    |
|              | nucX4    |                               | 1.3GHz Quad-core                     |        |
| Virtual Wall | n1011-06 | Zwijnaarde, Ghent             | Intel i5-9400, 2.90GHz, 6<br>Core    | 32GB   |
|              | n1013-01 |                               |                                      |        |
| POWDER       | nuc3     | Salt Lake, Utah               | Intel i5-5300U, 2.3GHz,<br>Quad-Core | 8GB    |
|              | nuc4     |                               |                                      |        |

TABLE II. Hardware Resources Used

Our inter-testbed experiment deploys a mix of wireless/wired nodes using bare-metal machines in CityLab, w-ilab1.t, w-ilab2.t, virtual wall, and POWDER testbeds (Table II). We selected the testbed’s nodes based on their availability at the time of experimentation. We installed Ubuntu 18.04 on each node and collected CPU and memory information. Table II depicts this information along with location information which is available on the testbed’s website [4], [6]. The results in this paper are averaged over 50 readings.

### A. Availability of Public IPv4 and IPv6 addresses

Figure 1 shows whether the selected nodes at each testbed have public IPv6 or IPv4 addresses. This is investigated by running the Linux ‘‘ifconfig’’ command on each node. It shows that all the nodes except the nodes in the POWDER have public IPv6 addresses. Additionally, while public IPv4 addresses are not assigned by default at virtual wall nodes, they can be requested as a separate resource at the virtual wall nodes. The problem is that there are limited public IPv4

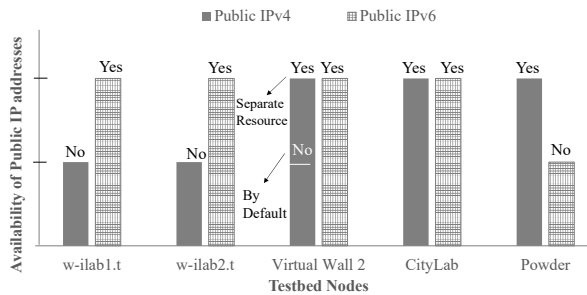


Fig. 1: Availability of Public IP addresses

addresses available at the virtual wall (the first and second virtual walls have 47 and 53 public IPv4 addresses available respectively). Further, there are no public IPv4 addresses available for w-ilab1.t and w-ilab2.t. Moreover, nodes in the CityLab and POWDER have public IPv4 addresses by default.

### B. Topology of our inter-testbed experiment

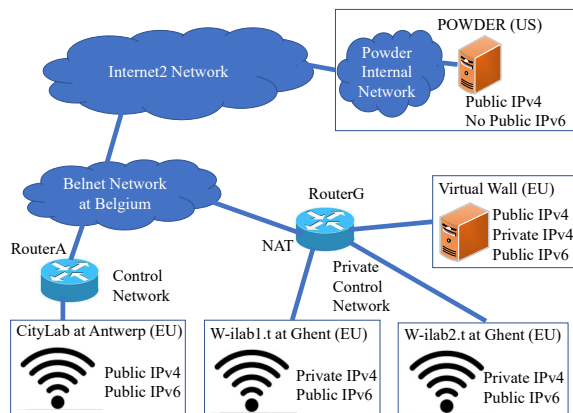


Fig. 2: Topology of our inter-testbed experiment

Figure 2 shows the discovered topology of our inter-testbed experiment using ‘mtr’ and ‘traceroute’ commands. W-ilab1.t, w-ilab2.t, and virtual wall testbeds are connected via a router (routerG) located at IMEC, Ghent. Additionally, routerG is connected to the Belnet network, which connects external educational institutions, research centres, and government centres in Belgium. CityLab, also in Belgium, directly connects to Belnet through a router at Antwerp University (RouterA). Moreover, we found that Belnet is directly connected to an independent US network called Internet2, which is dedicated to research and education and connects the POWDER testbed. We also infer the following from the above experiment:

- 1) Communication between virtual wall, w-ilab1.t, and w-ilab2.t is possible through private IPv4 addresses assigned to them, as they share the same private networks.
- 2) CityLab, w-ilab1.t, w-ilab2.t, and virtual wall are reachable to each other through public IPv6 addresses.
- 3) Communication between CityLab and the POWDER testbed is possible only using public IPv4 addresses. POWDER nodes do not have public IPv6 addresses.

- 4) As the nodes at the POWDER testbed only have public IPv4 addresses and the virtual wall nodes can request public IPv4 addresses, communication between the virtual wall and the POWDER testbed is possible using public IPv4 addresses.
- 5) Communication to the POWDER nodes (IPv4 address) from w-ilab1.t and w-ilab2.t is possible through a NAT (Network Address Translation) enabled at the routerG.
- 6) It is not currently possible to communicate with the w-ilab1.t and w-ilab2.t nodes from the POWDER nodes, as there are no public IPv4 addresses for w-ilab1.t and w-ilab2.t nodes and no public IPv6 addresses for POWDER nodes. This may be possible in the future by redirecting the traffic through the virtual wall.

### C. Number of Hops Travelled

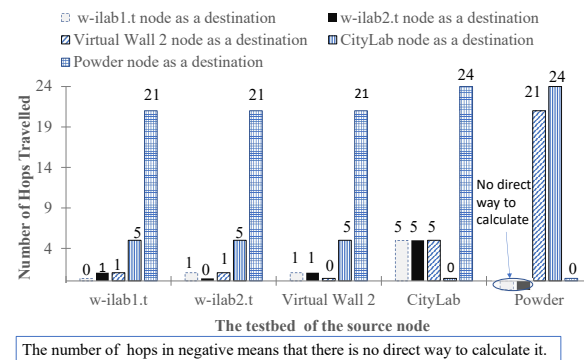


Fig. 3: Hops between different nodes

Figure 3 shows the hops between different nodes when using a traceroute application. The figure shows that nodes in a testbed are directly connected (i.e., 0 hops) to each other. Further, w-ilab1.t, w-ilab2.t, and the virtual wall testbeds are 1 hop away from each other. In addition, the CityLab is 5 hops away from w-ilab1.t, w-ilab2.t, and the virtual wall testbeds. Moreover, the POWDER nodes are 21 hops away from w-ilab1.t, w-ilab2.t, and virtual wall nodes, and 24 hops away from the CityLab nodes. We did not calculate the number of hops from the POWDER nodes as source to the w-ilab1.t and w-ilab2.t nodes as destination, as there is no direct way to calculate this (as mentioned in the previous subsection).

### D. Round Trip Time

Figure 4 shows the average round-trip time (RTT) of ping between two nodes over 50 readings. It also displays the minimum and maximum RTT through an error bar. The shortest overall RTT (0.3 ms) was recorded between nodes of the same testbed. The RTT between w-ilab1.t and w-ilab2.t are less than 1 ms. The RTT between CityLab and any of the other testbeds in Belgium is less than 5 ms while the longest RTT is 140 ms between the EU and the US testbeds.

### E. UDP Stress Test

Figure 5 illustrates the maximum permissible UDP bandwidth (the limit after which traffic will be dropped due to

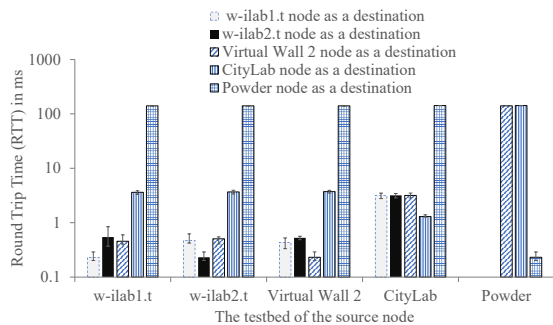


Fig. 4: Ping Round Trip Time (RTT) in ms

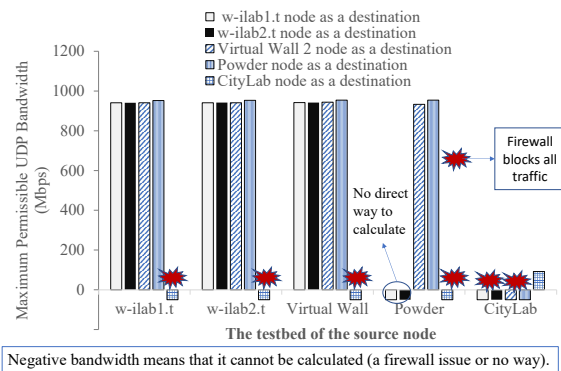


Fig. 5: Bottleneck Bandwidth in Mbps

insufficient bandwidth) between the testbed nodes determined using Iperf. As UDP does not wait for an acknowledgement from a sender, the distance travelled has no effect on the maximum permissible bandwidth. For w-ilab1.t, w-ilab2.t, virtual wall, and POWDER, the maximum bandwidth is approximately 940 Mbps. The maximum bandwidth between CityLab nodes is approximately 92 Mbps. Furthermore, since the iperf experiment between CityLab and any other testbed did not work, it seems that there is a firewall at CityLab that blocks external UDP traffic. Therefore, we could not calculate the maximum bandwidth between CityLab and any other testbed. Figure 3 shows that ping between CityLab and any other testbeds works fine because it seems that the Citylab firewall just passes only ICMP traffic (ping traffic). Additionally, we did not calculate the maximum bandwidth available from POWDER to the w-ilabs as there is no direct way to communicate from POWDER to w-ilabs without using tunneling through the virtual wall (as explained in the previous section).

Figure 6 shows the percentage of loss when the traffic between the nodes reach the amount of the bottleneck bandwidth achieved in Figure 5. When traffic is sent within a testbed, there is no loss. Additionally, we did not observe any packet loss when traffic is sent from one w-ilabs node to another w-ilabs node or a virtual wall node. There is however a loss of traffic when the virtual wall or any other testbed is used.

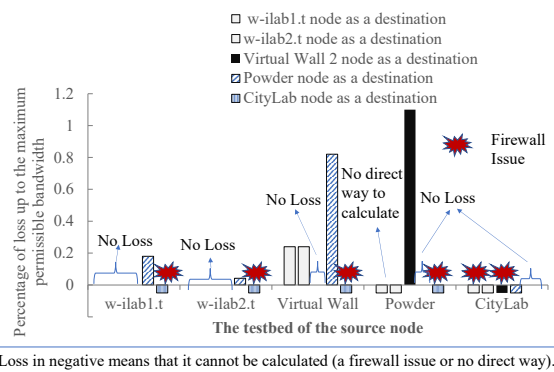


Fig. 6: Loss % when traffic equals the bottleneck bandwidth

When traffic is sent from the virtual wall to any of the w-ilab.t testbeds, the packet loss is less than 0.4%. There could be an issue with RouterG that is causing the packet loss. Since users cannot access this router, we cannot determine the cause of this loss. Packet loss increased to approximately 1% when traffic is sent between the virtual wall and the POWDER. Packet loss could also have occurred somewhere in the Belnet or Internet2 or at the router (RouterG) placed at the virtual wall. This minimal loss makes w-ilab1.t, w-ilab2.t, virtual wall, and POWDER suitable for inter-testbed experiments.

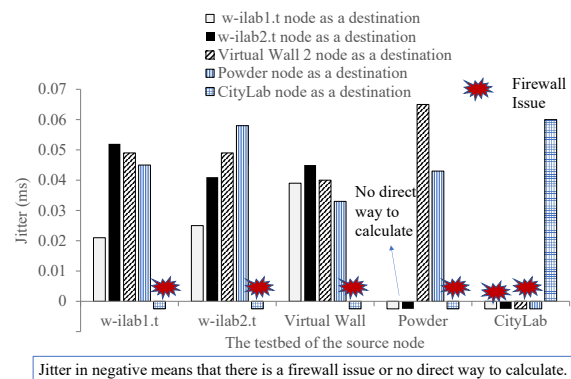


Fig. 7: Average value of Jitter in ms

The average jitter is shown in Figure 7. It shows the jitter is under 0.1 ms, which is minimal for an inter-testbed experiment where nodes are located in two continents.

#### F. TCP Stress Test

Figure 8 shows the maximum TCP bandwidth in Mbps. Figure 8 shows the effect of distance on the maximum TCP bandwidth, since TCP waits for an acknowledgement from the previous segment before sending the next segment, and also re-transmits any unacknowledged segments. The maximum TCP bandwidth is lower than the maximum UDP bandwidth (see Figure 5), as UDP does not wait for an acknowledgement. For traffic sent between w-ilabs.t or virtual wall, the maximum TCP bandwidth is approximately 921 Mbps. It is worth noting, however, that the maximum bandwidth between POWDER and any of the other testbeds is approximately 150 Mbps, which is

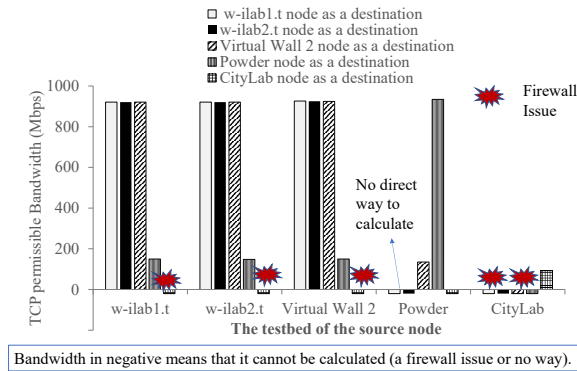


Fig. 8: TCP Bottleneck bandwidth - Mbps

considerably lower than the UDP bandwidth (941 Mbps). This may be because POWDER is located far away from the rest of the testbeds and acknowledgements from the receiver have to travel a long path. Further, the maximum TCP bandwidth between nodes in POWDER is approximately 934 Mbps.

### III. SDN INTER-TESTBED EXPERIMENT

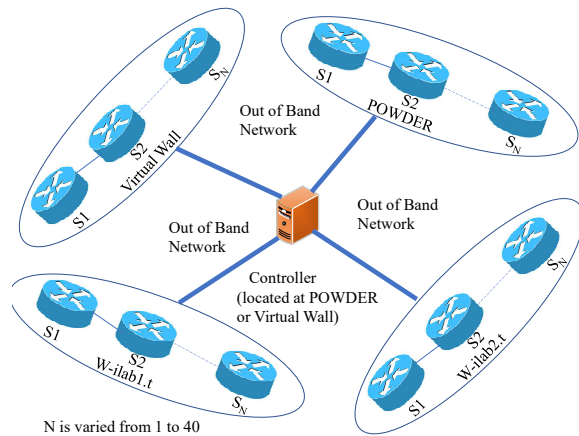


Fig. 9: SDN/OpenFlow Networks created between testbeds

We use only w-ilab1.t, w-ilab2.t, virtual wall, and POWDER testbeds for our SDN experiment, since only these testbeds are found to be appropriate for inter-testbed experiments. For this experiment, OpenFlow is used as an SDN protocol, and a single controller controls all the switches present in each testbed (see Figure 9). Open vSwitch [14] and the POX controller [15] are used as the OpenFlow switch and the controller node. The controller is placed at the POWDER node or the virtual wall node and communicates with the OpenFlow networks created on each testbed through an out-of-band network shown in Figure 9. Figure 9 also shows the deployed OpenFlow switch topology. Mininet is used to create this OpenFlow topology in each testbed. To compute the performance of this SDN network located at several testbeds, the number of OpenFlow virtual switches is increased.

Figures 10 and 11 show OpenFlow session establishment times when the controllers are at the virtual wall and at

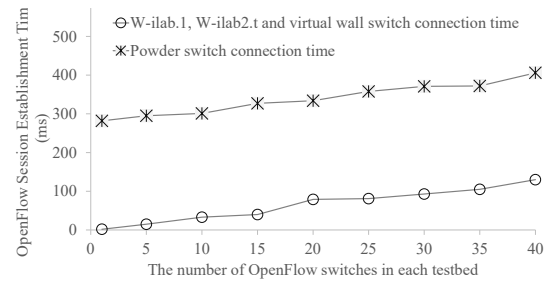


Fig. 10: OpenFlow Session Establishment time in ms of all switches when the controller is placed at the Virtual Wall

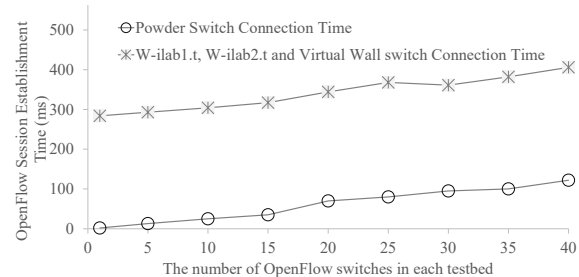


Fig. 11: OpenFlow Session Establishment time in ms of all switches when the controller is placed at a POWDER node

the POWDER, respectively. When the number of switches increases, there is an approximately linear increase in the OpenFlow session establishment time. We observed no significant difference in results when an OpenFlow network is created at w-ilab1.t, w-ilab2.t, or virtual wall testbeds as they are all located in the same building. Thus, the average of all the results from these testbeds is presented in Figures 10 and 11. The results show that when the controller is located far away, the OpenFlow session establishment time is significantly long.

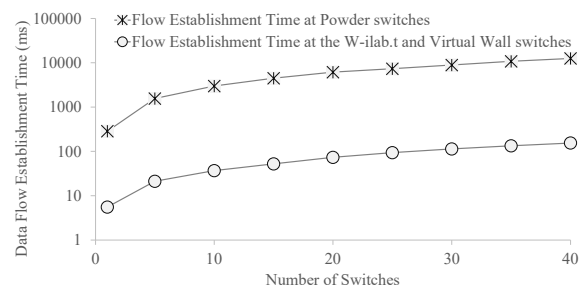


Fig. 12: Data Flow Entry Establishment time of all switches when the controller is placed at the virtual wall (EU) node

Figure 12 and 13 show the data flow establishment time, which is defined as the instant when the controller is able to establish forwarding entries in all the switches along the path to the destination, which is at the end of the network. Figure 12 shows the time when the controller is placed at the virtual wall node and Figure 13 shows the time when the controller is placed at a POWDER node. These results show that when the number of switches along a path to the destination increases, the data flow establishment time increases approximately linearly. This is because the controller

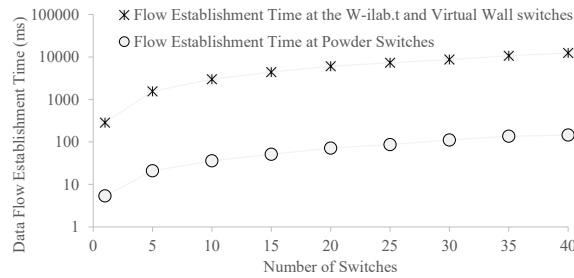


Fig. 13: Data Flow Entry Establishment time of all switches when the controller is at the POWDER (US) node.

has to establish more forwarding entries, as the number of switches along the path increases. As flow establishment time is approximately the same for OpenFlow switches in w-ilab1.t, w-ilab2.t, and virtual wall, the results are combined, and the average is shown in Figure 12 and 13. Our learning from this experiment includes:

- 1) OpenFlow sessions and flow establishment times are significantly longer when the controller is in the EU and the OpenFlow switches are in the US, and vice versa.
- 2) If the controller and OpenFlow switches are located at the same location (e.g., in the EU or US), these times are significantly shorter.
- 3) Experiments on EU and US testbeds can be useful to create edge-computing type scenarios in which some functionality can be moved closer to users to allow for faster decisions and other functionality can reside at a remote testbed as illustrated in Section IV.

#### IV. RESOURCE INTENSIVE EDGE COMPUTING USE CASE

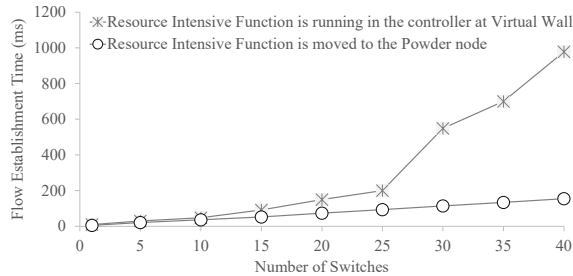


Fig. 14: Flow establishment with the edge computing use case.

An edge-computing use case is implemented on the EU-US testbeds. This use case illustrates how an edge node with limited resources slows down controller functions in presence of resource-intensive computing functions and how offloading the computing functions to the cloud can help. During this experiment, a controller is placed at the virtual wall to control the OpenFlow switches in w-ilab1.t and w-ilab2.t. We consider two use cases: (1) the controller at the virtual wall performs both forwarding decisions and resource-intensive computing, processing a lot of data; (2) the controller at the Virtual Wall offloads the resource-intensive tasks to the US controller at the POWDER node.

In this experiment, all controller tasks are performed by a single CPU of a virtual wall node. Figure 14 shows that

in the first case, the resource-intensive functions are running concurrently with the controller’s normal forwarding function, causing tasks such as establishing Flow Entries to be delayed. But, when the resource-intensive functions are moved to the POWDER node in the US, the controller can perform normal actions (e.g., establishing forwarding entries) quickly.

#### V. CONCLUSIONS

This paper reports our cross-Atlantic inter-testbed activity between the EU and the US to perform benchmark experiments, Software Defined Networking (SDN) and Edge Computing resource intensive experiments. The conclusions from these experiments include: (1) the testbeds that can be connected together based on address space compatibility and firewall issues, (2) the maximum permissible bandwidth between different testbed nodes, (3) the physical location of the controller (EU or US) significantly affects OpenFlow sessions and flow establishment times, and (4) a background resource intensive function running on the same controller also affects flow establishment times. This paper is an exemplary instance of integrating diverse testbeds i.e., in terms of capabilities (pure wireless, wired or standard TCP/IP, platforms, and functionalities, and being managed by SDN. Our results have laid a strong foundation for more advanced cross-Atlantic experiments enabled by SDN and an edge computing usecase.

#### ACKNOWLEDGEMENT

This research has received funding from EU H2020 NGIAtlantic project under agreement No. OC3-292.

#### REFERENCES

- [1] “Geni test-bed.” [Online]. Available: <https://www.geni.net/>
- [2] “Emulab test-bed.” [Online]. Available: <https://www.emulab.net/>
- [3] “Planetlab.” [Online]. Available: <https://planetlab.cs.princeton.edu>
- [4] “Fed4fire test-bed.” [Online]. Available: <https://www.fed4fire.eu/>
- [5] “Orbit test-bed.” [Online]. Available: <https://www.orbit-lab.org/>
- [6] “Powder test-bed.” [Online]. Available: <https://powderwireless.net/>
- [7] “Cosmos testbed.” [Online]. Available: <https://www.cosmos-lab.org/>
- [8] C. Both and et al, “Futebol control framework: Enabling experimentation in convergent optical, wireless, and cloud infrastructures,” *IEEE Communications Magazine*, vol. 57, no. 10, pp. 56–62, 2019.
- [9] T. Wauters and et al, “Federation of Internet experimentation facilities: architecture and implementation,” in *European Conference on Networks and Communications (EuCNC 2014)*, Bologna, Italy, Jun. 2014.
- [10] S. Bouckaert and et al, “Federating wired and wireless test facilities through emulab and omf: The ilab.t use case,” in *Testbeds and Research Infrastructure. Development of Networks and Communities*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 305–320.
- [11] J. Struye and et al, “The citylab testbed — large-scale multi-technology wireless experimentation in a city environment: Neural network-based interference prediction in a smart city,” in *IEEE INFOCOM 2018 - IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, 2018, pp. 529–534.
- [12] J. Breen and et al, “Powder: Platform for open wireless data-driven experimental research.” New York, NY, USA: Association for Computing Machinery, 2020.
- [13] S. Sharma, S. Urumkar, G. Fontanesi, B. Ramamurthy, and A. Nag, “Future wireless networking experiments escaping simulations,” *Future Internet*, vol. 14, no. 4, 2022. [Online]. Available: <https://www.mdpi.com/1999-5903/14/4/120>
- [14] “Open vswitch.” [Online]. Available: <https://www.openvswitch.org/>
- [15] “Pox controller.” [Online]. Available: <https://github.com/noxrepo/pox>