White Paper

Communications Service Providers
Service Assurance

Intracom Telecom Machine Learning Boosts NFV Energy Efficiency

The Intracom Telecom NFV Resource Intelligence™ (NFV-RI™) and Intel® Xeon® Scalable processors with Enhanced Intel SpeedStep® Technology predict virtual network function traffic loads to optimize power savings

The benefits of network functions virtualization (NFV) are well understood by communications service providers (CoSPs) and are a key enabler as mobile networks expand from network core to edge. In two of the most prominent cellular network functions, namely evolved packet core (EPC) and radio access network (RAN), many CoSPs have reported a significant total cost of ownership (TCO) reduction.

But at the same time, there are important challenges. In contrast to non-virtualized IT workloads, virtual network functions (VNFs) based on the Data Plane Development Kit (DPDK) have very stringent KPIs, like deterministic performance, low latency, and zero packet drops. An implication of this is that the CPU in a server will always operate at the highest frequency for an NFV workload in order to achieve the requisite performance.

Specifically, a property of DPDK-based VNFs is that they keep the servers constantly running at a high power-up state, as if the servers are always operating for peak demand. The reason is that DPDK is relying on polling mechanisms to achieve zero drops, low latency, and high-throughput in packet processing. This polling keeps the VNF CPUs at the highest CPU utilization at all times, even during idle periods, thus consuming the maximum possible power.

Many VNFs use the open source Data Plane Development Kit (DPDK) to improve data throughput in a virtualized environment. DPDK enables fast networking functions and accelerated packet processing. Existing middleware, like the Linux operating system, fails to dynamically adjust CPU frequencies for DPDK-based VNFs, even though they feature power management policies to dynamically manage power for CPUs when they are not busy. The reason is they are unaware of the actual load a DPDK VNF has at any time. DPDK VNFs appear to the OS as being always busy, at 100 percent CPU utilization. Finding ways to leverage this knowledge of the actual CPU utilization allows the implementation of mechanisms to adapt CPU frequencies according to the VNF's load.

Intel® Network Builders ecosystem partner Intracom Telecom has added a frequency feedback loop (FFL) workflow to its Network Functions Virtualization Resource Intelligence™ (NFV-RI™) to predict VNF traffic levels and dynamically adjust the frequencies of each CPU core used by DPDK-based VNFs according to their incoming load, while promoting zero packet drops. In tests conducted by Intracom Telecom, power consumed by the system was reduced by 14% on average, and up to 35%, over a 24-hour period.[1]

Intracom Telecom NFV-RI™ Overview

The NFV-RI is a resource intelligence platform that runs on Intel® architecture-based servers and leverages Intel® Resource Director Technology (Intel® RDT) to provide enhanced performance on multi-tenant platforms where cache and memory bandwidth is contended between different applications. Using Intracom Telecom's NFV-RI platform, CoSPs can apply optimization policies and processes to their VNFs to meet service levels promised in their SLAs.

Table of Contents

NFV-RI employs artificial intelligence (AI)-powered exploration to search for resource allocations that optimize the performance of one or more VNFs, in a fully automated fashion, requiring zero domain expertise from the user. The platform considers CPU placement, last-level cache (LLC) capacity, DRAM bandwidth, and CPU frequency for each of the co-located workloads. It operates in closed-loop form, taking feedback from performance probes provided by the VNFs to correlate the impact of its resource decisions on their performance. Once the automated exploration is complete, the NFV-RI defines and assigns shared resources like CPU cores and LLC into chunks dedicated for private use on multi-tenant platforms where cache and memory bandwidth is contended between different applications.

Intracom Telecom Frequency Feedback Loop (FFL)

To overcome the power waste associated with DPDK-based VNFs constantly running at a high-power state, even when not necessary, Intracom Telecom has added the frequency feedback loop (FFL) workflow to its NFV-RI. The FFL is a new workflow that uses machine learning to predict VNF traffic levels to enable reduced power usage during light or off-peak traffic periods, without compromising performance. The solution can dynamically adjust the frequencies of CPUs processing DPDK-based VNFs according to their incoming load, while helping meet the goal of zero packet drops, in a fully automated, closed-loop fashion.

FFL has a real-time, closed-loop mechanism that adapts the frequency of the cores the VNF is running on to match their actual traffic load, while helping ensure that frequency is high enough to support zero packet drops. This means that VNFs are operating at maximum CPU frequencies during peak hours, and moderate or minimum frequencies during off-peak or light traffic hours.

FFL Working with Enhanced Intel SpeedStep® Technology

In FFL, Intracom Telecom leverages Enhanced Intel SpeedStep® Technology for Intel processors to dynamically alter the operating frequency of CPUs. This feature allows the system to dynamically adjust processor voltage and core frequency, decreasing average power consumption and heat production. Combined with existing power-saving features, Enhanced Intel SpeedStep Technology can provide balance between power production and consumption. Enhanced Intel SpeedStep Technology uses design strategies that include separation between voltage and frequency changes and clock partitioning and recovery.

Because the feature reduces the latency associated with changing the voltage/frequency pair (referred to as P-state), those transitions can be undertaken more often. More-granular, demand-based switching and optimization of the power/performance balance is enabled.

FFL Machine Learning

At the heart of the FFL is a machine learning (ML) module that predicts how busy VNFs will become in the next short-term period. To achieve this, it uses network interface card (NIC) statistics, CPU event counters, and metrics from the VNFs themselves, from current and previous time windows. Therefore, depending on the current power level and the predicted busyness level, the mechanism proactively scales a VNF's frequencies up in order to prevent packet drops, or scales it down, in order to save power.

Specifically, FFL can handle two kinds of traffic variations using ML techniques:

- Predict imminent overload situations (i.e., VNF capacity saturation while operating at a certain CPU frequency) and scale up frequencies well in advance, at a suitable level.
- Detect underload situations (i.e., operating with more CPU frequency than is sufficient to sustain a certain traffic level) and scale down frequencies at a suitable level, but in a more gradual manner.

In case of abrupt, sudden traffic bursts that are impossible to predict, FFL reacts instantly, within 10-20 msec, setting the CPU frequencies to their maximum level directly.

FFL is able to handle open VNFs, those that expose their busyness in terms of the load on receive (RX) queues via the DPDK Telemetry API, or through custom RESTful element managers. FFL can also work with closed VNFs that do not expose anything. In the latter case, FFL employs ML techniques to infer the busyness level of a VNF indirectly, leveraging platform metrics. Because this approach lacks the observability of the open VNFs, in most cases the FFL is forced to operate more conservatively (i.e., higher frequencies), in an effort to avoid packet losses at the cost of missing energy optimization opportunities.

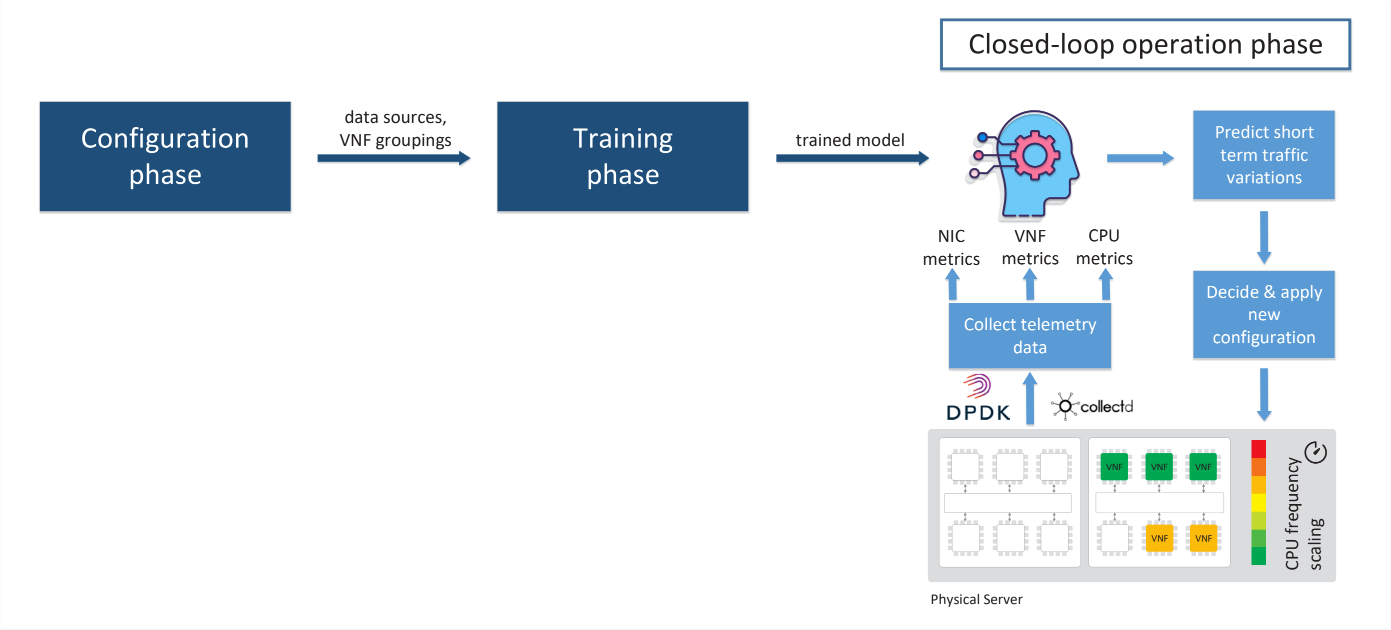

Figure 1. FFL operation at a glance.[2]

Three Phases of FFL

Figure 1 shows the three phases of FFL operation—configuration, training, and closed-loop operation.

In the configuration phase, the user provides all the parameters needed for configuring FFL in its subsequent phases, like the suitable data sources or the mapping of VNFs to FFL instances. Note that with FFL the user can simultaneously run multiple independent instances of the feedback loop on the same platform, with each instance controlling a group of one or more VNFs. The VNFs within a group are typically tightly coupled, like a chain of packet processing functions that accept their own traffic. As such, the decisions for scaling CPU frequencies are made independently for them, according to their current traffic load.

In the training phase, FFL is trained to find ideal frequencies for different traffic levels. The ideal CPU frequency for a certain traffic level is the minimum CPU frequency that yields zero packet drops. In general, the higher the traffic levels, the more CPU frequency is required to keep packet drops at zero. Note that different VNFs may have different ideal frequencies, even for the same traffic, and on the same platform. In this phase the user is initially asked to specify the maximum known capacity of the VNF, that is, the maximum amount of traffic that the VNF is able to sustain in production without errors.

Subsequently, FFL will attempt to identify the ideal frequencies for a certain number (N) traffic steps, uniformly distributed in the range between 0 and Cmax, asking the user to feed VNFs with the corresponding traffic rate in each step. The outcome of this phase is a trained model able to predict how traffic is going to vary in the subsequent short-term period and decide the best CPU frequency for the predicted level.

In the closed-loop operation phase the user launches one or more FFL instances on the machine, specifying which trained model should be used for each. This phase may run indefinitely in production. It dynamically detects changes in traffic load and decides automatically which CPU frequency should be applied to the VNFs.

FFL Requirements

FFL can be applied on packet processing VNFs that use DPDK version 19.11 or higher. VNFs should be running as bare-metal applications on Linux platforms, and they should be directly attached to the platform's NICs. Support of KVM- and Kubernetes-based VNFs accepting traffic via SR-IOV is in the works. FFL interfaces with the Collectd telemetry framework, which should be already installed and running on the platform (version 5.11 or higher).

To be able to dynamically adjust the VNFs' CPU frequencies according to their incoming load, FFL needs to continuously monitor, in real-time, two data sources:

1. A data source indicating how busy the VNF currently is.
2. A data source indicating the current number of dropped packets.

Busyness Indication Data Source

There are two options for the data source that indicates the busyness level of a VNF:

- The VNF itself may expose its busyness, represented by the load on its RX queues. Specifically, FFL requires metrics expressing: a) the number of times the polling threads encountered zero packets at the receive queues ("empty polls"), and b) the number of times the polling threads found enough packets to fully fill the queues ("full polls"). Typically, open VNFs can be easily extended to provide these metrics, if they do not expose them already. Full and empty poll counts may be exposed either through the DPDK Telemetry API,[3] or through a RESTful Element Manager.
- Obtaining the metrics from a closed VNF requires the FFL to monitor CPU event counts to infer the busyness indirectly.

Dropped Packets Data Source

Packets that are being dropped on the RX queues of a VNF are an indication that the VNF is in an overload situation, usually operating at a lower CPU frequency to sustain the current traffic without errors. In FFL, the dropped packets can be measured via one of the following ways:

- The DPDK Telemetry API
- The extended NIC stats DPDK API (accessible through Collectd's "dpdkstat" plugin)
- A custom, RESTful key-value element manager

Burst Tolerance

Intracom Telecom has evaluated[4] how effective FFL is in handling sudden traffic bursts, in an effort to understand what the main factors are that affect it. Intracom Telecom has used the sample DPDK L2 forwarding application[5] extended with busyness metrics like the ones of vEPC (e.g., full polls and empty polls). The packet size is 64 bytes. The packet rate ranges between ~3.9 Mpps (2 Gbps) and ~31.2 Mpps (16 Gbps).

The key conclusion is that the VNF's tolerance to packet drops on sudden bursts depends not only on the burst size, but also on the starting and ending traffic levels. Per Table 1, the tests show that significant bursts can be processed between 2 to 4 and 2 to 10 Gbps and this is an acceptably large burst. In scenarios when bursts are even larger than that, for example a burst size of 10 Gbps, that yields ~60 drops when starting from 2 Gbps, but yields ~20,400 drops when starting from 6 Gbps. The larger the ending traffic of the burst is, and the smaller the current CPU frequencies are, the faster the RX queues, including network interface card queues, are filling up, resulting in packet drops.

BURST (GBPS)	AVG DROPS
2 -> 4	0
2 -> 6	0
2 -> 8	0
2 -> 10	0
2 -> 12	60
2 -> 14	6,300
2 -> 16	22,000
6 -> 8	-
6 -> 10	-
6 -> 12	-
6 -> 14	3,200
6 -> 16	20,400
10 -> 12	-
10 -> 14	-
10 -> 16	291
14 -> 16	-

Table 1. Number of packet drops during sudden bursts; left column shows bursts in terms of starting data flows and ending data flows in Gbps. Right column shows number of packets dropped during the burst.

FFL Improves Energy Efficiency at Tier-1 CoSP in Greece

Intracom Telecom collaborated with a tier-1 CoSP in Greece in order to optimize the energy consumption in user plane components of the CoSP's mobile network.

Initially, Intracom Telecom's goal was to measure the user plane traffic on these components for many consecutive days in order to verify the variability of traffic load depending on the time of the day, and hence the existence of opportunities for energy optimization. Figure 2 shows typical data traffic patterns for uplink and downlink compiled by averaging the data traffic collected during the measurement period. This pattern of usage variability validates Intracom Telecom's initial expectation.[1]

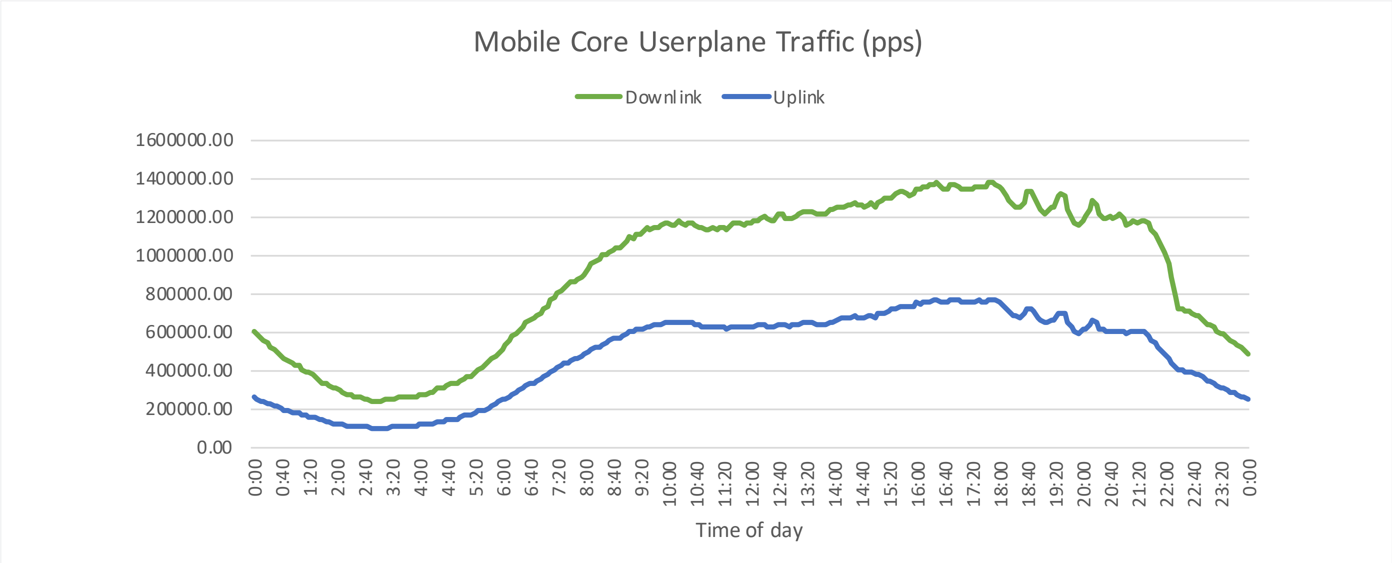

Figure 2. Traffic in packets per second measured for a single EPC node during an average 24-hour period.

The next step was to evaluate FFL's effectiveness in reducing the vEPC's energy consumption during the CoSP's off-peak hours. Intracom Telecom used OMEC,[6] one of the first full-featured, scalable, high performance open source EPCs, to prototype the CoSP's vEPC. Since Intracom Telecom's focus was on the polling-intensive components, it picked the user plane functions for evaluation, namely the service and packet gateway functions (SGW-U, PGW-U). The SGW-U and PGW-U were implemented combined as a single DPDK application, which was deployed alone on a server. A high-level diagram of the OMEC architecture, along with the components used for Intracom Telecom's evaluation, are shown in Figure 3. Downlink and uplink user plane traffic was generated by an external system, following the patterns shown previously in Figure 2.

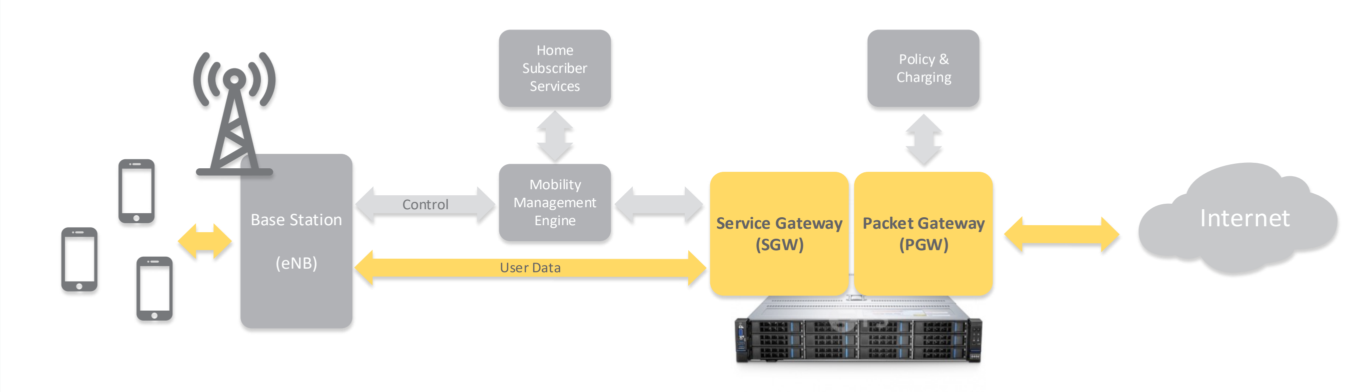

Figure 3. vEPC architecture.

One of Intracom Telecom's primary goals was to evaluate the tradeoff between the observability of the VNFs busyness, and the exploitation of energy-saving opportunities, comparing the energy consumption and the number of dropped packets in the case when the serving and packet gateways were handled by FFL as an open VNF, versus when it was handled as a closed VNF.

To evaluate the open VNF, Intracom Telecom extended OMEC's SPGW source code to expose the load on its RX queues via the following metrics: a) number of times the polling threads encountered zero packets at the queues (empty polls), and b) number of times the polling threads found enough packets to fully fill the queues (full polls). These metrics were exposed via the DPDK Telemetry API.

To evaluate the closed VNF, the FFL VNF could be used "out of the box," without further code modifications. The results are shown in Figure 4.

In the default case, the average power consumption was 264 Watts. In the open VNF the average power consumption was 227 Watts, a 14 percent savings. There were zero packet drops measured. In the closed VNF the average power consumption was also 227 Watts, a 14 percent saving. There were zero packet drops measured. The maximum power savings measured, either for the open or closed VNF was 35 percent (from 266 to 196 Watts). The two operation modes proved to be equally efficient in the long run, yielding the same average savings and always zero packet drops. Nevertheless, as shown in Figure 4, they exhibit different behavior on traffic variations.

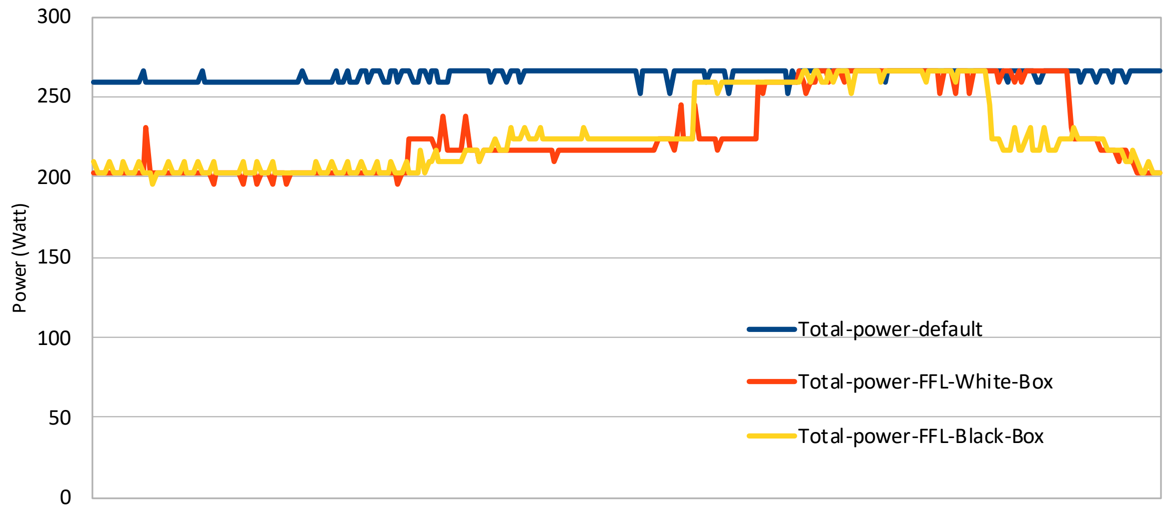

Figure 4. SPGW power consumption use cases for the traffic pattern of the vEPC shown in Figure 3. Blue line represents the default case (OS mechanisms used); red line represents the case when power is managed by FFL and it exposes its busyness; and yellow line represents the case when power is managed by FFL but does not expose anything.

Summarizing Greek Tier-1 CoSP Results

The top portion of Figure 5 below shows a representative downlink traffic pattern and the server power consumption for a virtualized EPC prototype Intracom Telecom evaluated, reproducing real traffic patterns measured in the Greek Tier-1 CoSP's mobile network during an average 24-hour period. According to the top right plot, by default, the power consumption on the server running the VNFs is constantly high, even though traffic varies according to the time of the day. For light, off-peak traffic periods (for example, between 22:00 and 08:00) this suggests overprovisioning, because VNFs could be perfectly operated at lower frequencies, consuming less power, without experiencing packet drops or increased latencies.

As illustrated in the bottom portion of Figure 5, Intracom Telecom's FFL was able to adjust the power consumption of the vEPC server in a way that consistently followed the traffic pattern.

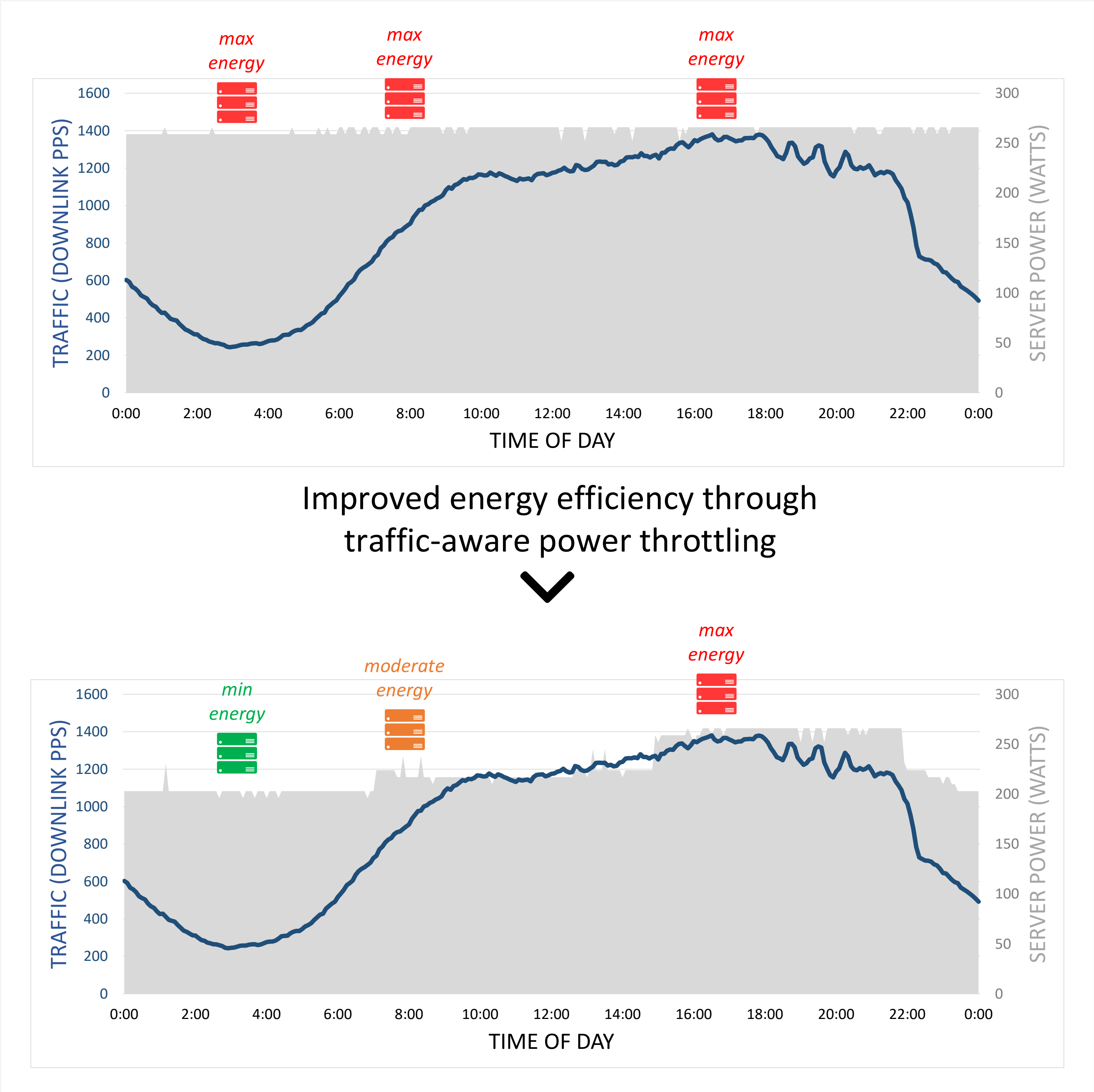

Figure 5. Demonstrates the reduced power consumption of the server using FFL (bottom green graph) given the same traffic pattern.

Conclusion

NFV provides CoSPs great benefits, but requires stringent KPIs. This results in VNFs keeping servers at a high-power state constantly, as if they are always operating for peak demand. Because of this, CPU frequency overprovisioning often occurs at light, off-peak traffic periods, resulting in unnecessary energy use.

Intracom Telecom's NFV-RI FFL workflow leverages machine learning to predict how busy VNFs will become. Using Enhanced Intel SpeedStep Technology for Intel processors, the FFL workflow can dynamically adjust the CPU frequencies of DPDK-based VNFs according to their incoming load, while always ensuring zero packet drops at the maximum rate of 15 Gbps. With the NFV-RI FFL workflow, CoSP infrastructure can operate at low power during off-peak or light-traffic periods and VNF service level objectives (SLOs) can remain assured and unaffected with the potential to reduce power consumed by the system by 14% on average, and up to 35%, over a 24-hour period.

Learn More

Intracom Telecom

Intracom Telecom NFV-RI

Intel® Xeon® Scalable processors

Enhanced Intel SpeedStep Technology Chapter 14

Notices & Disclaimers

[1] Data provided by Intracom Telecom, October 2020.

[2] Figures provided courtesy of Intracom Telecom.

[3] https://doc.dpdk.org/guides-18.11/howto/telemetry.html

[4] Testing conducted by Intracom Telecom in July 2020: SUT server utilized dual 24-core Intel® Xeon® Platinum 8168 processors (microcode: 0x2006906) supporting Non Uniform Memory Access (NUMA). Intel® Hyper-Threading Technology was turned on, and Intel® Turbo Boost Technology was turned off. BIOS version was SE5C620.86B.0X.01.0076.101320171718. Total RAM for node 0 was 196 GB and for node 1 was 198 GB. System storage totaled 960 GB and was provided by an Intel® SSD SC2BB960G7. 25 GbE network connectivity was provided by an Intel® Ethernet Controller XXV710. The system also featured a Platform Controller Hub integrated 10 Gigabit Ethernet Controller with integrated SATA controller. Operating system was Ubuntu 18.04.2 LTS with kernel version 5.4.0-45-generic. Workload 1 was l2fwd traffic forwarding app built into DPDK v19.11. Workload 2 was OMEC's Next Generation Infrastructure Core RTC evolved packet core. Compiler was gcc version 7.5.0 and libraries included collectd v5.9.2. Client server utilized dual Intel Xeon Platinum 8168 processors (microcode: 0x200005e) supporting Non Uniform Memory Access (NUMA). Intel Hyper-Threading Technology was turned on, and Intel Turbo Boost Technology was turned off. Total system RAM was 384 GB. System storage totaled 960 GB and was provided by an Intel® SSD Data Center 3520 SC2BB960G7. 25 GbE network connectivity was provided by an Intel Ethernet Controller XXV710.

[5] https://doc.dpdk.org/guides/sample_app_ug/l2_forward_real_virtual.html

[6] https://www.opennetworking.org/omec/

1220/DO/H09/PDF ♻ Please Recycle 345365-001US