Evaluating Energy Efficiency of Gigabit Ethernet and Infiniband Software Stacks in Data Centres

Vincenzo De Maio, Vlad Nae, Radu Prodan
Institute of Computer Science, University of Innsbruck, Austria
Email: {vincenzo, radu}@dps.uibk.ac.at

***Abstract*—Reducing energy consumption has become a key issue for data centres, not only because of economical benefits but also for environmental and marketing reasons. Many approaches tackle this problem from the point of view of different hardware components, such as CPUs, storage and network interface cards (NIC). To this date, few works focused on the energy consumption of network transfers at the software level comprising their complete stacks with different energy characteristics, and the way the NIC selection impacts the energy consumption of applications. Since data centres often install multiple NICs on each node, investigating and comparing them at the software level has high potential to enhance the energy efficiency of applications on Cloud infrastructures.**

We present a comparative analysis of the energy consumption of the software stack of two of today's most used NICs in data centres, Ethernet and Infiniband. We carefully design for this purpose a set of benchmark experiments to assess the impact of different traffic patterns and interface settings on energy consumption. Using our benchmark results, we derive an energy consumption model for network transfers and evaluate its accuracy for a virtual machine migration scenario. Finally, we propose guidelines for NIC selection from an energy efficiency perspective for different application classes.

I. Introduction

Energy-aware computing has emerged as an important research topic in high-performance and mainstream computing. A large amount of research [1], [2], [3] focused already on reducing energy consumption in data centres and improving their efficiency. Many of these works focus on specific hardware components such as CPUs, storage, memory, but few of them focus on network transfers. In the networking area, existing works investigate energy-saving techniques like sleeping and rate adaptation [4] with focus on routers and switches [5] or on MPI parallel scientific applications [6]. Several works like [7] focused on the energy consumption of network transfers in message passing models, but few investigated it at the *software level*, comprising their complete stacks with different power characteristics and the way they impact the energy consumption of applications. Since data centres often install multiple Network Interface Cards (NICs) on each node, we believe that investigating and comparing them at the software level has high potential to enhance the energy efficiency of applications on Cloud infrastructures.

In this paper, we investigate the main factors influencing the energy consumption of the software stack of the two mostly used networks in data centres: *Ethernet* and *Infiniband*. Our goal is to model their energy consumption at the application software level (not at the hardware level), considering all components involved in the network transfers (CPU, RAM, I/O, and NIC). For this purpose, we design a set of network-intensive benchmarks that emulate a wide spectrum of possible application behaviours. We vary parameters such as transfer size, number of simultaneous transfers, payload size, communication time, and traffic patterns. We focus on homogeneous nodes and on data transfers running over the TCP transfer protocol, because it is the most pervasive one according to [8]. We execute the benchmarks on machines equipped with NICs belonging to the two families and compare their energy consumption. We do not consider energy consumption of routers and switches because our goal is to investigate energy consumption of a data centre's node. We derive network transfer's energy consumption models for each network software stack and evaluate their accuracy in predicting the network energy consumption of non-live virtual machine (VM) migration. We use VM migration as a case study for two reasons: (1) it is widely used for fault tolerance and energy-aware consolidation in Cloud data centres [9], and (2) it is a network-intensive process. Finally, we propose guidelines for NIC's selection depending on the application characteristics.

The paper is organised as follows. We review the related work in Section II and the network hardware and its software stack in Section III. We introduce our benchmarking methodology in Section IV and the experimental setup in Section V. We analyse the benchmarks' results in Section VI, from which we derive an energy consumption model in Section VII. We evaluate the accuracy of our model in a VM migration scenario in Section VIII, We discuss our findings in Section IX and conclude our paper in Section X.

II. Related work

Energy aware networking: Many works exploit network awareness to save energy, with focus on routing equipment and algorithms: [10] investigates energy-aware allocation of resources in Clouds considering network topology, [11] proposes a network power manager which dynamically manages routers to reduce energy consumption, while [12] proposes a model for energy-aware routing. Complementary to these works, we focus on the energy consumption from the perspective of software application, including not only the NICs, but also the other components involved in network transfers.

DOI: https://doi.org/10.1109/UCC.2014.10

Network energy modelling: One of the first studies on network energy consumption focuses on energy consumption of routers, switches and hubs [13] but does not take into account energy consumed by the NICs. Many works like [5], [14] provide models for router power consumption, but do not consider the power consumed by NICs for network transfers. Other works [15] provide models for energy consumption of wireless network interfaces, which are of interest to mobile devices rather than data centres. [16] introduces a energy consumption model for network equipment and transfers for large-scale networks, based on transfer time and bandwidth. We propose here a complementary model for network transfers considering different NICs and more parameters. Works like [4] consider only transfer time when building a model for network transfers. In our work, we consider additional factors.

VM migration: One of the first works about VM migration is [17], but it does not take into account the energy consumption of this process. Other works such as [18], [19] investigate the advantages of using VM migration to achieve energy savings in data centers, but do not consider its own energy consumption. However, it focuses on the total energy consumed and does not highlight which consumption is related just to the network transfer. Moreover, this model makes a simplistic assumption that two nodes involved in a network operation consume the same energy, which may not be true for some NICs, as we will show in this paper.

III. Network hardware

We choose in our work the Ethernet and Infiniband NICs because they are to the best of our knowledge the most used interconnection technologies used in data centres. While communications running on Ethernet use the implementation of TCP/IP provided by the operating system, Infiniband software stack relies on kernel-bypass mechanisms and on RDMA-based capabilities. Such capabilities have a different impact on energy consumption. Therefore, comparing these two software stacks may give interesting insights about energy consumption of network transfers. In the next two subsections, we describe both interfaces in detail.

A. Ethernet

Ethernet is the most popular local-area network technology. It defines several protocols which refer to the family covered by the *IEEE 802.3* standard using four data rates:

- 10 Mbps for 10Base-T Ethernet, in IEEE 802.3;
- 100 Mbps, also called Fast Ethernet, in IEEE 802.3u;
- 1000 Mbps, also called Gigabit Ethernet, in standard IEEE 802.3z;
- 10-Gigabit, also called 10 Gbps Ethernet, in standard IEEE 802.3ae.

We focus on Gigabit Ethernet because, along with the newer 10-Gigabit, it is the most used interconnection technology in data centres. The minimum frame size for Gigabit Ethernet (1000Base-T standard) is 520 bytes, while the Maximum Transmission Unit (MTU) is 1500 bytes.

B. Infiniband

Infiniband is a popular switch-based point-to-point interconnection architecture that defines a layered hardware protocol (physical, link, network, transport), and a software layer to manage the initialisation and the communication between devices. Each link can support multiple transport services for reliability and multiple virtual communication channels. The links are bidirectional point-to-point communication channels that can be used in parallel to achieve higher bandwidth. Infiniband offers a bandwidth of 2.5Gbps in its single data rate version used in our work for comparison with Gigabit Ethernet. TCP/IP communications are mapped to the Infiniband transport services through IP over Infiniband (IPoIB) drivers provided by the operating system. An Infiniband NIC can be configured to work in two operational modes.

Datagram: is the default operational mode of IPoIB described in RFC 4391 [20]. It offers an unacknowledged and connectionless service based on the unreliable datagram service of Infiniband that best matches the needs of IP as a best effort protocol. The minimum MTU allowed is 2044 bytes, while the maximum is 4096 bytes.

Connected: mode described in RFC 4755 [21] offers a connection-oriented service with a maximum MTU of 2GB. Using the connected mode can lead to significant benefits by supporting large MTUs, especially for large data transfers.

Setting Infiniband in one of these two modes will result in mapping a TCP communication on a different Infiniband transport service. For this reason, we will measure the energy consumption of an Infiniband network transfer in both modes.

IV. Experimental Methodology

In this section we describe the benchmarking methodology for evaluating the energy consumption of the NIC software stacks. We first outline the impacting factors and then present the benchmarks and the evaluation metrics.

A. Energy-impacting factors

We describe the main factors affecting the energy consumption of network transfers according to our studies.

Time: this parameter must be considered since the longer a network transfer, the more energy it consumes.

Transport protocol: affects energy consumption because it defines the way in which transfers are performed. It defines how application layer's effective data are encapsulated. Such encapsulation inherently affects the NIC's operational mode and the amount of transferred data. While there exist many transport protocols (e.g. TCP, UDP, RSVP, SCTP), we only focus our analysis on TCP (the most pervasive one) due to space limitations.

Per-packet payload size: is the real data transmitted with a single packet, juxtaposed to a header that makes the communication possible. The payload size depends on many factors such as protocol configuration, physical layer MTU, maximum segment size (MSS, representing the largest amount of data that can be sent in a single packet) on TCP, and other application characteristics (e.g. some applications require frequent exchange of small packets). Payload size has an impact on time, since a smaller payload size implies a higher number of packets and thus, more headers to process.

Number of connections: to the NIC, typically shared among multiple applications that simultaneously send and receive data. With an increasing number of connections, one could experience a higher energy consumption due to the overhead introduced by their arbitration.

Traffic patterns: of different types generated by network-centric applications as showed in [22], characterised by the inter-arrival time of packets.

B. Benchmarks

We investigate each factor through six benchmarks. All the benchmarks run on TCP transport protocol.

BASE: benchmark investigates the impact of the network transfer on the energy consumption by transferring a fixed amount of data using sockets without any specific tuning.

PSIZE: benchmarks investigate whether the NIC energy consumption is related to the payload size under two premises: (1) *PSIZE-DATA* determines the impact of the payload size on energy efficiency independent of the data size by repeatedly transferring a fixed amount of data while varying the maximum payload size, and (2) *PSIZE-TIME* performs a maximum payload size evaluation with a fixed transfer time by continuously transferring data until the timeout we set is reached.

n-UPLEX: benchmark evaluates the energy consumption of NICs in full duplex (FD) mode, while handling multiple concurrent connections. We transfer a fixed amount of data using a varying number of FD connections on each machine.

PATTERN: benchmarks evaluate the effects of traffic patterns on energy consumption. We transfer data multiple times, and configure the data transmissions to be a succession of *burst* and *throttle* intervals, representing fixed time intervals in which the NICs are continuously communicating and idle, as depicted in Figure 1. For *PATTERN-B* we keep the throttle size constant and vary the burst size, while *PATTERN-T* we vary the throttle size keeping a constant burst size.

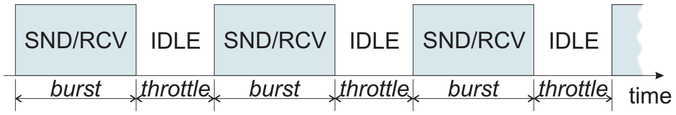


Fig. 1: PATTERN benchmark (burst/throttle intervals).

For the PSIZE benchmarks, we need to successively set the transferred data size and a transmission timeout, and to strictly control the packet size. This can be achieved by altering the MSS and by disabling any buffering algorithms. For the n-UPLEX benchmark, we need to configure the type of (FD/HD) connections and the number of simultaneous connections. Finally, the PATTERN benchmark requires the possibility to shape the communication patterns through variable burst and throttle intervals. In the next section, we are going to see how we implemented our benchmarks.

C. Nimble NEtwork Traffic Shaper

To configure the metrics of our study based on transfer data size and timeout, payload size, FD/half-duplex (HD) connections, connection concurrency, and transmission patterns, we analyzed three of the most popular open-source network diagnosis and benchmarking tools: `ttcp` (http://www.pcausa.com/Utilities/pcattcp.htm), `netperf` (http://www.netperf.org/netperf/) and `iperf` (http://iperf.sourceforge.net/). Table I presents a comparison of the flexibility of these tools focused on the provided configuration options for the metrics relevant to our study. Since none of the analysed tools covers all configuration parameters needed, we designed the *Nimble NEtwork Traffic Shaper (NNETS)*, a versatile network traffic shaping tool implemented in Python 2.7 using the standard socket API, publicly available under GNU GPL v3 license[1]. In addition to the custom design required for accommodating all studied configurations, the tool allows a proper instrumentation of network and energy metrics. We implemented it with a clear separation between data processing and networking operations in order to instrument only the relevant regions of code, excluding data staging and pre-/post- processing operations and ensuring that the measured energy consumption is strictly related to the network transfer.

D. Metrics

To evaluate software stacks' energy efficiency we employ the following five metrics:

Machine energy consumption: in Kilojoules (kJ) for running each experiment;

Network energy consumption: in Kilojoules (kJ), computed as the difference between the machine's energy consumption during benchmarks' execution and its idle consumption. This metric includes the energy consumed by all hardware components involved in a network transfer, which we purposely include to have a more realistic metric related to the software application and not to the hardware;

Average power: in Watts (W), defined as the ratio between network energy consumption and its execution time;

Energy per byte: in Nanojoules (nJ), defined as the ratio between the network energy consumption and the number of bytes transferred, which indicates how energy consumption varies in relation to the size of data transfer;

Energy per packet: in Millijoules (mJ), defined as the ratio between the network energy consumption and the amount of packets transferred.

V. Experimental Setup

We employ two machines, both equipped with Infiniband and Gigabit Ethernet NICs, as specified in Table II. We set the MTU on all machines to 16382 bytes for the Infiniband NICs in connected mode, to 2044 bytes in datagram mode, and to 1500 bytes for the Gigabit Ethernet NICs. The machines are connected through two dedicated server-grade network switches to exclude the impact of external network traffic. For each NIC and connectivity mode, we run the benchmarks in three configurations (send, receive and n-uplex), namely:

[1] To be published at: http://code.google.com/p/nnets/

Tool	*Transfer data size*	*Transfer timeout*	*MSS setting*	*Disable buffering*[1]	*FD/HD connections*	*Concurrent connections*	*Variable burst*	*Variable throttle*
`ttcp` (v1.12)	✓	✗	✗	✓	✗	✗	✗	✗
`netperf` (v2.4)	✓	✓	✗	✓	✗	✗	✗	✗
`iperf` (v2.05)	✓	✓	✓	✓	✓	✓	✗	✗

[1] e.g. an option for setting the TCP_NODELAY

TABLE I: Comparison of networking benchmarking/diagnosis tools.

Host	*CPU*	*Kernel*	*Gigabit NIC*	*Infiniband NIC*	*Gigabit switch*	*Infiniband switch*	dom0 kernel	Xen version
`k12`	4× Opteron 880	Linux 2.6.9-67	Broadcom BCM5704	SDR Mellanox MT23108	Cisco Catalyst 3750	Mellanox MT47396 Infiniscale-III	3.0.4	4.2
`k13`								

TABLE II: Experimental hardware.

- ETH-SND/RCV, ETH for Gigabit Ethernet in send, receive and n-uplex;
- IBC-SND/RCV, IBC for Infiniband connected in send, receive and n-uplex;
- IBD-SND/RCV, IBD for Infiniband datagram in send, receive and n-uplex.

For the energy measurements, we use Voltech PM1000+ power analysers (with 0.2% accuracy) connected to the machines' AC side and capable of reading the power twice per second. For each benchmark, we select the input parameters to produce an execution time of at least 50 seconds, which allows us to have at least 100 readings in each execution. Table III summarises the experimental parameters. The *data* and *time* columns denote the termination condition of each benchmark experiment. When the data size is set, the experiment terminates after transferring the indicated amount of data (i.e. the session and transport overheads), while when the timeout is set, the experiment is terminated after the indicated time. The *payload* indicates the size of the useful data in each packet, computed as a percentage of MTU minus 40 bytes (the size of IP and TCP headers), but for simplicity we denote it as "a percentage of MTU". The *connections* column indicates the number of concurrent connections through which the transfer is made. Finally, the *burst* and *throttle* represent the concrete time intervals of continuous activity and inactivity of the NICs. For the PSIZE benchmarks, we vary the maximum payload between 30% and 100% of the NICs' MTU. We also set the `TCP_NODELAY` flag to prevent packets smaller than MTU from being buffered. For PSIZE-DATA we set the data size to 75GB, while for PSIZE-TIME we set a timeout of 5 minutes. For the n-UPLEX benchmark, we transmit a fixed amount of data of 150GB (sending 75GB and receiving 75GB) over n FD connections. For both PATTERN benchmarks, we set the data size to only 11GB, as the studied traffic patterns considerably increase the transfer times. In the PATTERN-B benchmark, we keep the throttle size constant to 10 $\mathrm{ms}$and vary the burst size to 2, 4, 6, 8, and 10 ms. Conversely, for the PATTERN-T benchmark, we vary the throttle to 2, 4, 6, 8, and 10 ms with a constant burst size of 10 ms. We run each experiment for ten times, which ensures an average coefficient of variation of 0.053, and present the average of the results.

VI. Experimental Results

In this section we present the results of our experiments.

A. BASE

We observe in Table IV a considerable difference in energy consumption for running the BASE benchmark. The immediate finding is that transferring the same quantity of data over Infiniband in connected mode is more efficient than the other alternatives in terms of energy and time. We can also observe that Infiniband's energy consumption significantly differs between sending and receiving operations: 30% less energy for sending than receiving in connected mode, and 10% less energy for receiving compared to sending. It is also noteworthy that, even in this simple benchmark, the network energy consumption is between 1.58 and 6.33 kJ, which can potentially be up to 20% of energy consumption in a node with lower idle power consumption. The other metrics provide supplementary insight into these NICs' energy efficiency. Although it might appear that the Infiniband in connected mode is more energy efficient with the lowest average power in operation, this only holds true when the two communicating parties require large amounts of on-hand data to be transferred. When the communication is message centric and the volume of effective data is low, resulting in a high number of packets being transmitted, the Gigabit Ethernet NIC is the more energy-efficient choice, closely followed by Infiniband in datagram mode. In conclusion, these preliminary findings hint that an energy efficient network communication depends on the nature of the traffic generated by the application. For data intensive traffic in applications such as data warehousing and content streaming or delivery, the more energy-efficient network is Infiniband configured in connected mode. On the other hand, for finer-grained traffic and real-time message exchanges such as low traffic databases and online games, Gigabit Ethernet is more efficient. The following experiments give further assessment of the energy consumption of the two networks with respect to traffic characteristics.

B. PSIZE

We begin with the PSIZE benchmark, focused on the influence of the payload size on networks' energy efficiency.

Benchmark	*Size* [GB]	*Time* [min]	*Payload* [% MTU]	*Connections*	*Burst* [ms]	*Throttle* [ms]
PSIZE-DATA	75	–	**30 – 100**	1 HD	–	–
PSIZE-TIME	–	5	**30 – 100**	1 HD	–	–
n-UPLEX	150	–	100	**1 – 8 FD**	–	–
PATTERN-B	11	–	100	1 HD	**1 – 10**	10
PATTERN-T	11	–	100	1 HD	10	**1 – 10**

TABLE III: Benchmark summary with focus metric in bold.

Configuration	*Machine energy [kJ]*	*Network energy [kJ]*	*Execution time [s]*	*Average power [W]*	*Energy per packet sent/received [mJ]*	*Energy per byte sent/received [nJ]*
ETH-SND	291.8	6.01	674	8.92	0.11	73
ETH-RCV	291.3	5.94	673	8.80	0.10	71.9
IBC-SND	131.6	1.58	307	5.14	0.54	20.0
IBC-RCV	133.0	2.26	308	7.36	0.78	28.8
IBD-SND	182.1	6.33	414	15.3	0.16	78.3
IBD-RCV	175.6	5.72	401	14.3	0.14	71.0

TABLE IV: BASE benchmark results (I).

PSIZE-DATA.: The results in Figure 2(a) show that the energy consumption of the software stacks of the studied NICs is inversely proportional to payload, the most efficient operational point being reached for the maximum payload. Also noteworthy is the significantly better scalability in terms of energy when employing Infiniband NIC in connected mode: 36% energy consumption increase for a 50% decrease in payload, versus 84% for Gigabit Ethernet and 79% increase for Infiniband in datagram mode. Analysing the other metrics presented in Figure 2, we can identify in detail the energy-to-payload relation. Figure 2(b) suggests that, while for Infiniband in connected mode the energy consumption per transferred packet is proportional to its payload, it is relatively constant in the case of Infiniband in datagram mode and Gigabit Ethernet. Conversely, Figure 2(c) reveals a stronger inverse correlation between the payload and the energy consumption per transferred effective byte. The Infiniband in datagram mode and the Gigabit Ethernet NICs are affected in terms of energy efficiency by a payload decrease, the energy consumption per effective byte nearly tripling at a 30% of MTU payload. This behaviour is less severe for Infiniband in connected mode, the energy per byte doubling for a payload of 30% of MTU.

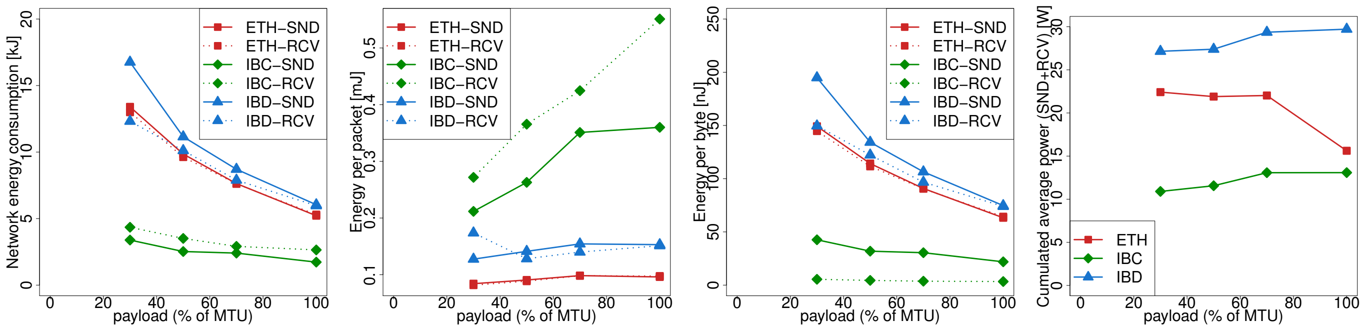
PSIZE-TIME.: We present the resulting average power consumption in Figure 2(d), each point representing the cumulated power for send and receive operations. The main finding of this experiment is that the energy consumption of both Infiniband and Gigabit Ethernet NICs is not exclusively correlated with running time. We observe that while Infiniband (regardless of its operational mode) consumes in average less power with lower payloads, Gigabit Ethernet is more power efficient at higher payloads. Further investigation revealed that Gigabit Ethernet's high power efficiency for larger payloads is likely due to driver optimisations, as we noticed a 32% decrease in CPU utilisation between the transfers with payloads set at 30%, respectively 100% of MTU. The CPU utilisation was constant for all Infiniband transfers in both modes.

To conclude, energy consumption of the networks is inversely proportional to the maximum payload size. Second, Gigabit Ethernet and Infiniband in datagram mode are better suited for lightweight, mixed traffic (with varying payload sizes), while Infiniband connected is by far the most energy efficient for non-fragmented traffic. Finally, network energy consumption is not exclusively time-related, thus one cannot optimise for time and expect proportional savings.

C. *n-UPLEX*

We observe in Figure 3(a) a considerable increase in the energy consumption of Gigabit Ethernet and Infiniband in datagram mode with more concurrent connections. The trend has a piecewise linear shape and is relatively similar for the power traces shown in Figure 3(b). In contrast, Infiniband in connected mode shows a decreasing energy consumption with the increase in concurrent connections. Moreover, although Infiniband in connected mode consumes the least energy for transferring the fixed data amount for multiple connections, it is clearly exhibiting the highest average power consumption. This raises a question regarding the NICs' performance in terms of transfer bandwidth in this contention scenario. We present in Table V a comparison between the variation of the achieved bandwidth, consumed energy, and CPU utilisation between the two extreme cases studied: (1) the network contention case with eight concurrent FD connections and (2) the single FD connection. The results reveal a significant increase of 72% in bandwidth for the Infiniband connected, with a 19.1% average power increase. This variation of its power state with performance (in terms of bandwidth), is the reason of its energy efficiency. At the other end, Gigabit Ethernet exhibits the highest increase in energy consumption of almost 50% with only a marginal 2.5% increase in bandwidth. The considerable average power consumption increase in all cases stems from both (1) NICs requiring more power to handle the increased load and (2) increasing CPU overheads for managing multiple simultaneous connections. This observation is supported by the non-proportional energy consumption versus the CPU utilisation increase shown in Table V. Finally, the increase of CPU utilisation for Infiniband in connected mode is 130.15% higher than the other two configurations due to the increased bandwidth requiring faster data preprocessing.

In summary, in a connection concurrency environment significant power consumption penalties occur, the Infiniband in



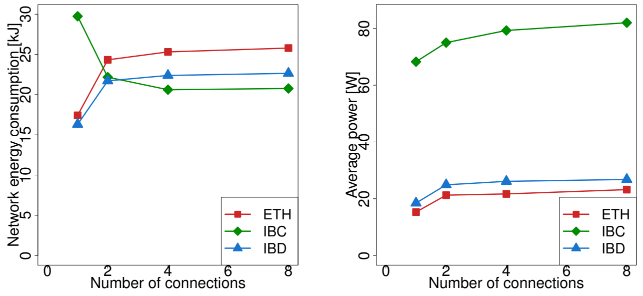
(a) PSIZE-DATA: energy consumption.

(b) PSIZE-DATA: energy per packet.

(c) PSIZE-DATA: energy per byte.

(d) PSIZE-TIME: Cumulated send and receive average power.

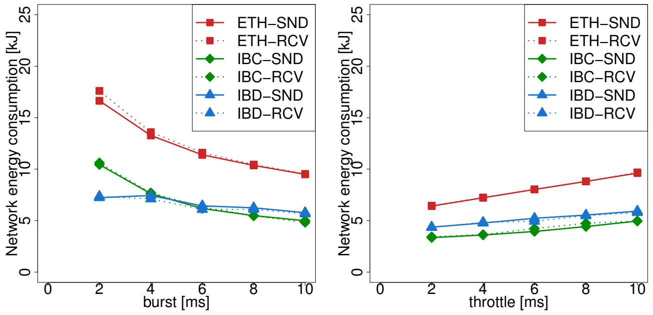
Fig. 2: PSIZE benchmark results.



(a) Network energy.

(b) Average power.

Fig. 3: n-UPLEX benchmark results.



(a) PATTERN-B.

(b) PATTERN-T.

Fig. 4: PATTERN benchmark results.

Metric	*Variation [%] (8 vs 1 connections)*		
	ETH	*IBD*	*IBC*
Bandwidth	+2.49	+4.39	+72.03
Energy	+45.80	+37.33	−31.03
Power	+49.43	+43.37	+19.11
CPU	+38.62	+38.23	+130.15

TABLE V: Variation of relevant metrics with number of concurrent connections.

connected mode being the best choice in terms of energy efficiency. The increased power consumption is due to a higher NICs' power state and to processing overheads.

D. PATTERN

These two experiments study the energy consumption of the NIC software stacks for different communication patterns.

PATTERN-B.: Figure 4(a) shows that Gigabit Ethernet is the least energy efficient for all studied burst intervals. For short burst intervals ($2 - 4$ms), Infiniband datagram is surprisingly more efficient consuming up to 44% less energy than in connected mode. For longer burst intervals, connected mode becomes better consuming 17% less energy.

PATTERN-T.: Figure 4(b) shows a stable, monotonously increasing energy consumption with increasing throttle intervals. It is noteworthy that the energy consumption increases at different rates for the different NICs and operational modes: Gigabit Ethernet's consumption increases by 110J per ms of throttle, while Infiniband by 49J in datagram mode, and by 55J in connected mode. Although Infiniband connected is more energy efficient for the studied configurations, a basic extrapolation shows that for traffic patterns with throttle intervals higher than 50ms the datagram mode becomes the more energy efficient choice.

In conclusion, Infiniband in datagram mode shows the least variation in energy consumption with different transmission patterns (good choice for mixed/undetermined transmission patterns), while Infiniband in connected mode exhibits a very good energy efficiency in a few particular cases (good choice for long transmission bursts).

VII. Network Energy Consumption Model

We model in this section the factors analysed in Section IV-A that affect the network energy consumption. We believe that such a model would help scientists in more accurately predicting the energy consumption of network-intensive applications, as required for example by resource managers and schedulers. We decided to use regression analysis, that has been successfully used in previous energy prediction and modelling works [23]. We employ the NLLS regression algorithm.

For extracting model parameters, we employ the data gathered from ten experimental runs. We assess the accuracy of our models using two metrics: (1) mean absolute error (MAE) and (2) root mean squared error (RMSE) that is also an absolute deviation metric, but more sensitive to large deviations. The difference between the two metrics is a measure of the variance in the individual deviations for all samples. We will also present a normalised value of RMSE (NRMSE) for metric-independent comparisons.

We model the energy consumption of a network transfer as:

$$E = \sum_{x \in \{\texttt{send},\texttt{receive}\}} \left(E_x(\texttt{DATA}_x, p_x, b_x, t_x) + O\left(c_x\right) \right), \quad (1)$$

where $\texttt{DATA}_x$ is the number of bytes transferred, p_x the payload per packet, c_x the number of additional connections (for FD transfers), and b_x and t_x the size of burst and throttle intervals in ms. We calculate E_x as:

$$E_x = \alpha_x \cdot \frac{\texttt{DATA}_\texttt{x}}{p_x} + \frac{\beta_x}{b_x} + \gamma_x \cdot t_x + \mathcal{K}_x, \quad (2)$$

where $x \in \{\texttt{send}, \texttt{receive}\}$, a_x can be interpreted as the cost for sending, respectively receiving a packet, β_x and γ_x are the model parameters, and $\mathcal{K}_x$ is a hardware and driver-related constant for setting up a sending, respectively receiving connection. Regarding the overhead of multiple connections, since Gigabit and Infiniband datagram use the NICs in a different way compared to Infiniband connected, their arbitration of multiple connections will be different too. For this reason, we employ Equation 3 for both Gigabit and Infiniband datagram and Equation 4 for Infiniband connected:

$$O_{\texttt{datagram}}(c_x) = \log(\epsilon \cdot c_x + \zeta); \quad (3)$$

$$O_{\texttt{connected}}(c_x) = \epsilon \cdot c_x^{\zeta}, \quad (4)$$

where ϵ and ζ are the model parameters and $x \in \{\texttt{send}, \texttt{receive}\}$. Table VI shows the model parameters along with the error, calculated over all the samples. The error is always below 9.4% which demonstrates a good accuracy.

VIII. VM Migration

After assessing the accuracy of the model in predicting network energy consumption of our own benchmarks, we evaluate its accuracy in predicting network energy consumption of VM migration on different NICs. We use the same hardware configuration as in Table III, and a dom0 GNU/Linux kernel version 3.0.4 for running Xen on one CPU with 512MB of RAM. We migrate a paravirtualized VM running a 2.6.32 Linux kernel on one CPU, and set its memory size to 4GB to ensure a long-enough migration time for an accurate energy measurement. We issue the migration by using Xen's xm command line interface. We measure the network energy consumption by instrumenting the machines during the migration time and subtracting their static energy consumption. We employ Equation 2 by setting the $\texttt{DATA}_\texttt{send}$ and $\texttt{DATA}_\texttt{receive}$ parameters for the SND and RCV configurations to the memory size in bytes. We set $b_\texttt{send}$ and $b_\texttt{receive}$ to the migration time, $t_\texttt{send}$ and $t_\texttt{receive}$ to 0, and finally $c_\texttt{send}$ and $c_\texttt{receive}$ to 1. We extracted new α and $\mathcal{K}$ parameters because of the different kernel version. In Table VII we show the estimation error for four runs (average coefficient of variation of 0.012). For brevity, we show just the MAE value and the NRMSE, since MAE and RMSE are equals. We use the range of values of each execution as range for NRMSE. As we can see, our model has a maximum MAE of 1.5J, which corresponds to a 16.2% NRMSE. In most of the cases, anyway, the NRMSE is below 9.4%, with a MAE lower than 0.9J, which compared to the total energy consumption for migration (between 248 and 349J) is a quantity which does not significantly affect the accuracy of our prediction.

IX. Discussion

Our results show that there is no "best" NIC in terms of energy efficiency. Furthermore, even setting the same NIC to different operational modes produces distinct results. Comparing for example the results from the PSIZE (Section VI-B) and PATTERN (Section VI-D) benchmarks, we find that Infiniband connected outperforms Infiniband datagram for continuous data transfers at maximum payload (72% less energy consumption), while for different communication patterns Infiniband in datagram mode is 44% more efficient than in connected mode. Therefore, choosing the optimal NIC for energy efficient communication depends on the application requirements and its communication characteristics. For exchanging large data quantities, Infiniband connected will save over 50% of energy compared to Gigabit Ethernet or Infiniband datagram. However, if the application needs to frequently send or receive small packets, Infiniband operating in datagram mode can be a better choice. When the number of exchanged messages is more relevant to the application than the quantity of data transferred, Gigabit Ethernet presents the lowest energy consumption per transferred packet.

One could think on dynamically exploiting NIC capabilities by selecting at runtime the most energy-efficient interface for the given application's communication characteristics. Multiple applications contending for NICs can also contribute to the complexity of this task. As a first step towards this challenging goal, we use our work to define the general guidelines for making a correct decision from an energy efficiency perspective based applications communication characteristics, as summarised in Table VIII.

X. Conclusions and future work

We performed in this paper a comparative analysis of the energy efficiency of today's mostly used NIC families in data centres, Gigabit Ethernet and Infiniband. First, we introduced NNETS, a versatile network benchmarking tool offering eight configuration parameters, some not covered by existing tools (e.g. variable traffic patterns, full duplex connections). Second, we designed a set of benchmarks and evaluated the energy efficiency of the NICs' software stacks in different configurations covering a wide spectrum of possible application behaviours. Third, we introduced energy models capable of providing accurate estimations based on the NIC type of adapter and transfer characteristics including payload size,

	$\alpha_{\texttt{send}}[\mu\text{J}]$	$\alpha_{\texttt{receive}}[\mu\text{J}]$	$\beta_{\texttt{send}}$	$\beta_{\texttt{receive}}$	$\gamma_{\texttt{send}}$	$\gamma_{\texttt{receive}}$	$\mathcal{K}_{\texttt{send}}[\text{kJ}]$	$\mathcal{K}_{\texttt{receive}}[\text{kJ}]$	ϵ	ζ	*MAE* [kJ]	*RMSE*	*NRMSE*
ETH	73.5	71.3	19.71	21.57	0.59	0.58	0.35	0.35	733.14	-685.56	0.44	0.9	0.03
IBC	137.1	181.4	13.93	14.23	0.23	0.19	0.58	0.80	12.59	-0.21	0.82	2.62	0.09
IBD	97.9	69.0	4.13	3.96	0.22	0.16	2.37	2.16	99.52	-82.13	0.83	0.98	0.05

TABLE VI: Model parameters and error.

Configuration	$\alpha_{\texttt{send}}[\mu J]$	$\alpha_{\texttt{receive}}[\mu J]$	$\mathcal{K}_{\texttt{send}}[kJ]$	$\mathcal{K}_{\texttt{receive}}[kJ]$	*MAE [J]*	*NRMSE*
ETH	65.3	62.9	0.044	0.044	0.4192	0.1344
IBC	309.5	309.2	0.166	0.167	1.5015	0.1628
IBD	132.8	108.2	0.063	0.137	1.4076	0.0939

TABLE VII: Non-live migration model parameters and errors.

Application characteristic	*Preferred NIC*
Big data, continuous traffic	Infiniband connected
Continuous message passing	Gigabit Ethernet
Multiple parallel connections	Infiniband connected
Low communication/computation	Infiniband datagram
High communication/computation	Infiniband connected

TABLE VIII: Guidelines for NIC selection depending on communication characteristics.

connection concurrency and traffic patterns with an average error of 6.1%. Fourth, we tested the accuracy of our model in predicting energy consumption of a non-live VM migration process, obtaining an average error of 9.8%. Fifth, we proposed a set of guidelines for choosing the most energy efficient NIC.

We plan to extend this work by studying the impact of transport protocol on the energy consumption and by analyse the impact of NICs and our selection guidelines on real-world network-intensive parallel and distributed applications.

Acknowledgment

Austrian Science Fund (FWF) project TRP 237-N23 funded this work.

References

[1] D. Abts, M. R. Marty, P. M. Wells, P. Klausler, and H. Liu, "Energy proportional datacenter networks," *SIGARCH Comput. Archit. News*, vol. 38, no. 3, pp. 338–347, 2010.

[2] L. Huang, Q. Jia, X. Wang, S. Yang, and B. Li, "Pcube: Improving power efficiency in data center networks," in *CLOUD '11*. IEEE, 2011, pp. 65–72.

[3] D. Kliazovich, P. Bouvry, and S. U. Khan, "Dens: Data center energy-efficient network-aware scheduling," in *GREENCOM-CPSCOM '10*. IEEE, 2010, pp. 69–75.

[4] S. Nedevschi, L. Popa, G. Iannaccone, S. Ratnasamy, and D. Wetherall, "Reducing network energy consumption via sleeping and rate-adaptation," in *NSDI '08*. USENIX, 2008, pp. 323–336.

[5] H. sheng Wang, L. shiuan Peh, and S. Malik, "A power model for routers: Modeling alpha 21364 and infiniband routers," *IEEE MICRO*, vol. 23, no. 1, pp. 26–35, 2003.

[6] P. Alonso, R. Badia, J. Labarta, M. Barreda, M. Dolz, R. Mayo, E. Quintana-Orti, and R. Reyes, "Tools for power-energy modelling and analysis of parallel scientific applications," in *ICPP '12*, 2012, pp. 420–429.

[7] S. Jana, O. Hernandez, S. Poole, and B. Chapman, "Power consumption due to data movement in distributed programming models," in *Euro-Par 2014*. Springer International Publishing, 2014, vol. 8632, pp. 366–378.

[8] M. Alizadeh, A. Greenberg, D. A. Maltz, J. Padhye, P. Patel, B. Prabhakar, S. Sengupta, and M. Sridharan, "Data center tcp (dctcp)," *SIGCOMM Comput. Commun. Rev.*, vol. 41, no. 4, pp. –, Aug. 2010.

[9] F. Farahnakian, T. Pahikkala, P. Liljeberg, and J. Plosila, "Energy aware consolidation algorithm based on k-nearest neighbor regression for cloud data centers," in *IEEE/ACM UCC*, 2013, pp. 256–259.

[10] M. Alicherry and T. V. Lakshman, "Network aware resource allocation in distributed clouds," in *INFOCOM '12*. IEEE, 2012, pp. 963–971.

[11] B. Heller, S. Seetharaman, P. Mahadevan, Y. Yiakoumis, P. Sharma, S. Banerjee, and N. McKeown, "Elastictree: Saving energy in data center networks." in *NSDI '10*. USENIX, 2010, pp. 249–264.

[12] Y. Shang, D. Li, and M. Xu, "Energy-aware routing in data center network," in *SIGCOMM '10 workshop on Green networking*. ACM, 2010, pp. 1–8.

[13] M. Gupta and S. Singh, "Greening of the internet," in *SIGCOMM '03*. ACM, 2003, pp. 19–26.

[14] J. Chabarek, J. Sommers, P. Barford, C. Estan, D. Tsiang, and S. Wright, "Power awareness in network design and routing," in *INFOCOM '08*. IEEE, 2008, pp. 457–465.

[15] Q. Wang, M. Hempstead, and W. Yang, "A realistic power consumption model for wireless sensor network devices," in *SECON '06*. IEEE, 2006, pp. 286–295.

[16] A.-C. Orgerie, L. Lefevre, I. Guerin-Lassous, and D. Pacheco, "Ecofen: An end-to-end energy cost model and simulator for evaluating power consumption in large-scale networks," in *WoWMoM*, 2011, pp. 1–6.

[17] C. Clark, K. Fraser, S. Hand, J. G. Hansen, E. Jul, C. Limpach, I. Pratt, and A. Warfield, "Live migration of virtual machines," ser. NSDI'05. USENIX, pp. 273–286.

[18] S. Srikantaiah, A. Kansal, and F. Zhao, "Energy aware consolidation for cloud computing," in *HotPower'08*. USENIX, 2008, pp. 10–10.

[19] C.-H. Hsu, S.-C. Chen, C.-C. Lee, H.-Y. Chang, K.-C. Lai, K.-C. Li, and C. Rong, "Energy-aware task consolidation technique for cloud computing," ser. CloudCom '11. IEEE, 2011, pp. 115–121.

[20] J. Chu and V. Kashyap, "Transmission of IP over InfiniBand (IPoIB)," RFC 4391, IETF, 2006.

[21] V. Kashyap, "IP over InfiniBand: Connected Mode," RFC 4755, IETF, 2006.

[22] T. Benson, A. Akella, and D. A. Maltz, "Network traffic characteristics of data centers in the wild," in *IMC '10*. ACM, 2010, pp. 267–280.

[23] W. Li, H. Yang, Z. Luan, and D. Qian, "Energy prediction for mapreduce workloads," in *(DASC), 2011 IEEE Ninth International Conference on*, 2011, pp. 443–448.