Optically Disaggregated Data Centres with Minimal Remote Memory Latency: Technologies, Architectures, and Resource Allocation

Georgios Zervas, *Member, IEEE,* Hui Yuan, Arsalan Saljoghei, Qianqiao Chen, and Vaibhawa Mishra

(Invited Paper)

***Abstract*—Disaggregated Rack-Scale Data Centres have been proposed as the only promising avenue to brake the barrier of the fixed CPU-to-memory proportionality caused by main-tray direct-attached conventional/traditional server-centric systems. However, memory disaggregation has stringent network requirements in terms of latency, energy efficiency, bandwidth and bandwidth density. This paper identifies all the requirements and key performance indicators (KPIs) of a network to disaggregate IT resources while summarizes the progress and importance of optical interconnects. Crutially, it proposes a rack and cluster scale architecture which supports the disaggregation of CPU, memory, storage and/or accelerators blocks. Optical circuit switching (OCS) forms the core of this architecture, whereas the end-points (IT resources) are equipped with on-chip programmable hybrid electrical packet/circuit switches. This architecture offers dynamically reconfigurable physical topology to form virtual ones, each embedded with a set of functions. It analyses the latency overhead of disaggregated DDR4 (parallel) and the proposed HMC (serial) memory elements on the conventional and the proposed architecture. A set of resource allocation algorithms are introduced to a) optimally select disaggregated IT resources with lowest possible latency, b) pool them together by means of virtual network interconnect and c) compose virtual disaggregated servers. Simulation findings show up to 34% resource utilization increase over traditional data centres while highlighting the importance of the placement and locality between compute, memory and storage resources. In particular, the network-aware locality based (NALB) resource allocation algorithm achieves as low as 15 nsec, 95 nsec and 315 nsec memory transaction round-trip latency on 63%, 22% and 15% of the allocated VMs accordingly while utilizing 100% of the CPU resources. Furthermore, a formulation to parameterize and evaluate the additional financial costs endured by disaggregation is reported. It is shown that the more diverse the VM requests are the higher the net financial gain is. Finally an experiment was carried out using silicon photonic mid board optics and an optical circuit switch that demonstrates FEC-free 10^{-12} bit error rate performance on up to five tier scale-out network.**

***Index Terms*—memory, accelerator and storage disaggregation, reconfigurable and function embedded architecture, hybrid OCS/EPS, on-board silicon photonic transceivers.**

I. Introduction

DATA centres have been historically based on a server-centric approach with fixed amounts of processor and directly attached memory resources within the boundary of a mainboard tray. Current data centres follow this model, but have to support highly diverse workloads ranging up to 4-orders of magnitude on memory over CPU demand [1]. The mismatch between fixed proportionalities and diverse set of workloads leads to substantially under-utilized resources (often at only 40%) that account for 85% of the total data centre cost [2]. Server-centric data centres use overlay networks, with various protocols and optimization goals, e.g. InfiniBand for low-latency, FibreChannel for Storage Area Networks. To consolidate I/O and switching infrastructure, minimize cost and power as well as increase network flexibility a reconfigurable network functions virtualization (NFV) system has been designed and implemented in [3] as a protocol-independent programmable switch [4] to simplify development.

The vision of disaggregation is to depart from the traditional paradigm of the mainboard-as-a-unit (server-centric model) and to enable the creation of function block-as-a-unit (resource-centric model) having a baseline disaggregated pool of components including a) compute, b) memory c) storage d) network and e) accelerators. The result is a new type of computing system that is network-centric and can offer immense flexibility that can potentially maximize resource utilization while enabling new workflows and applications with few resource boundaries. However, a number of fundamental challenges arise on such communication-centric computer architectures and need addressing:

- latency overheads, compared to current direct-attached model, should be minimized,
- system should support a substantially higher bandwidth and bandwidth density at very low cost and power consumption,
- network architecture and system should offer specific performance and services according to communication type (e.g., compute-to-memory, compute-to-storage) on same substrate for maximum flexibility
- orchestration of compute, memory and network resources to maximize resource utilization and workload performance at minimum latency and cost.

To address these challenges this article proposes:

1) the dRedBox (disaggregated Recursive Datacentre-in-a-Box) architecture that offers circuit switching (electrical for very short reach (centimeters) and optical for up to 1 Km reach communication) at its core to:
   a) offer lowest possible latency between end-points,

G. Zervas, H. Yuan, A. Saljoghei, Q. Chen and V. Mishra are with the Department of Electronic and Electrical Engineering, University College London, London, UK e-mail: (see https://www.ee.ucl.ac.uk/staff/academic/uceegze).

X. YYY and Y. XXX are with Anonymous University.

Manuscript received July XX, 2017;

   b) deliver deterministic latency,
   c) guarantee end-to-end bandwidth,
   d) be modular and scalable to support both high data rates (scale-up) or number of end-points (scale-out).
2) use programmable packet/circuit switching on the end-points embedded with processor, accelerator and memory units and associated logic to
   a) eliminate need for network interface cards and in turn substantially reduce latency of PCIe interfaces, footprint, power consumption and cost,
   b) allocate a switching service (packet or circuit) on each port dynamically to best suit data transaction requirements i.e. flow size, number of end-points, bandwidth, latency, etc.
   c) be able to dynamically programme the on-chip hardware and offer different protocols (i.e Ethernet, Infiniband, Fibrechannel), for different transactions (compute-to-memory, compute-to-end-user, etc.)
3) introduce and evaluate network-aware algorithms to allocate and maximize CPU utilization while delivering lowest possible round-trip latency.
4) offer diverse functions at run-time, enable the architecture to also offer topology and function reconfigurability. This means that a Virtual Machine (VM) or application request can be uniquely served by a custom topology and network function chain as reported in [4].
5) perform experiments using SiP MBO and optical switches to demonstrate the ability to deliver scale-out architecture to 3-tier and 5-tier optical network while having foward error correction (FEC) free operation (BER 10^{-12}).

Section II summarizes prior work on the various types of resource disaggregation, lists key requirements for networks and defines key performance indicators. Section III provides a thorough survey on optical interconnect technologies and highlights the more suitable ones for disaggregated systems. Section IV introduces the proposed architecture and ability, its theoretical round-trip latency values and elaborates on its ability to support function and topology reconfiguration. The resource allocation algorithms developed and simulations conducted are reported in Sections V and VI respectively. The experimental work is reported in Section VII prior to the Section IX that concludes the article.

II. Disaggregation and requirements

This section reports on resource disaggregation types reported in literature as well as specifies the set of network requirements and defines key performance indicators.

A. Related work

Resource component, i.e. memory, storage, and network, disaggregation from the compute elements i.e. CPUs, aims to maximize utilization of all resources. It also opens up a window of opportunities towards the formation of applications that are not bounded to resource proportionality that exists in direct-attached systems.

Disaggregation of non-volatile or long-term persistent data storage, i.e. hard disks, has been historically addressed and substantial progress has been made using storage area networks (SANs) and network attached storage (NAS) systems that both provide networked storage solutions. Flash storage disaggregation was recently proposed to deal with Flash over-provisioning [5]. It is most beneficial when the ratio of compute to storage requirements varies/or differs widely between applications . Gao *et. al.* [6] reportred on typical latency and bandwidth values within a traditional server and aimed to evaluate the minimum equivalent requirements for a disaggregated system. A roadmap report [7] describes a set of important metrics and requirements for interconnect technologies beyond bandwidth and latency which include cost per bit, cost per bandwidth-reach, bandwidth density, footprint and energy efficiency.

Memory disaggregation is the most challenging task amongst of all due to the tight dependencies which it poses with processors. First and foremost as described by Rao *et. al.* [8] there are two fundamental roadblocks that haven't allowed memory disaggregation to materialize: a) current network **latency** is too high and b) **bandwidth** between CPU and memory is too high to be supported by existing network architectures and technologies deployed in Data Centres (i.e. multi-tier networks using Layer 2/3 electronic equipment).

Simulation studies have recently indicated that network overheads due to additional network latency and lower access memory bandwidth could degrade the application performance by up to 66% and up to 20% respectively [9]. The additional cost of latency and bandwidth due to disaggregation is explored by Abali *et. al.* [10], which concludes that the penalty will be non-trivial at rack-scale while impractically high at data centre scale. This study also identifies the cost of optical interconnect to be at least 10 times more expensive compared to direct attached memory systems.

Overall, disaggregating each of the building blocks in such an architectures imposes a different set of requirements and so satisfying all of these under a single infrastructure is a challenging task. It triggers a plethora of requirements across hardware technologies, interconnect architecture, protocols, algorithms and the system software. Critically the network architecture and infrastructure needs to address the diverse requirements of each communication type i.e. CPU to CPU, CPU to memory, CPU to Storage.

Early attempts on network system experiments and orchestration algorithm for disaggregated architectures are reported recently in literature. Yan *et. al.* [11] reported a switch and interface card (SIC) to replace standard network interface card (NIC) and support packet and circuit switched services. However, the approach demonstrated memory to memory transactions instead of compute to memory transactions since the SIC) is interfacing to CPU via PCIe. This affects the transfer latency which is reported to be over 10 microseconds for even small size data i.e. 1KB. Such latency values impose substantial penalty to applications [9]. An integer linear programming (ILP) formulation to address the static planning of disaggregated resources was reported by Pages *et. al.* [12].

B. Requirements

This section highlights the key performance indicators across all the aspects reported above.

Network key performance indicators and definitions:

- **Latency** measured in *time, i.e. nanoseconds*, is defined and measured in a number of ways:
  - flit to flit (or head to head) latency is measured from a first flit sent by the source to the first flit arrived at the destination. This is used to evaluate a range of system communication such as processor to processor.
  - head to tail latency is the latency from the first flit of information sent to the last flit of information received. This is used to measure the time taken to write data from a processor to a memory element.
  - transaction latency is the time between a query (i.e. read data from memory) and the the arrival of results from memory back to the processor.
  - interrupt latency is the time that elapses from when an interrupt is generated to when the source of the interrupt is serviced.
- **Bandwidth** is the maximum data rate transferable by a device/interconnect/network measured in Gb/s and **bandwidth density** is the net data rate deliverable per footprint area and it is measured in *Gb/*cm^2
- **Cost** is the financial value of the device/interconnect/system/network and it is measured in *$*. Moreover, cost per bit is the cost per bit of data transmitted/processed/switched and measured in *$/bit*
- **Power consumption** or otherwise power usage which is measured in Watts is the total power required by a device/system to operate . Moreover **energy efficiency** is the amount of energy required to transmit/process/switch a bit of information and it is measured in *joules/bit*
- **Footprint** is the area occupied by a device/system and it is measured in cm^2
- **Network modularity and scalability** is reflected by the ability of the system to scale-out and/or scale-up in a pay-as-you-go fashion while having a good proportionally of end devices to network system.
- **Protocol flexibility and programmability** refers to the ability of a system not to be bound to a single protocol throughout its lifetime.
- **Architecture programmability/reconfigurability** refers to the ability of the infrastructure such as topology and accompanied functions such as switching (i.e. packet/circuit), forwarding, processing, to be programmed at run-time and be hardware re-purposed to suit the needs of the application and service demanded.

III. Enabling interconnect technologies

Hybrid memory cube (HMC) and high bandwidth memory (HBM) modules are already capable of supporting I/Os with multi Tb/s bandwidth [13]. These devices are one of the key memory technologies for disaggregation. This is due to their serialised I/Os that can potentially lead to very low system latency as it will be described in Section IV. Furthermore, international technology roadmap for semiconductors (ITRS) predicts that semiconductor chips will support I/O capacities beyond 1 Pb/s by 2030 [14]. It can be envisioned that the interconnects in the next generation disaggregated and resource-centric platforms should be able to support such high bandwidth demands while adhering to ultra-low latency requirements not present in conventional Data Centre architectures. As always, cost, performance, energy efficiency and device foot print are very important. Thus, given the high transmission losses, cross-talks and reflections associated with the electrical based interconnects, their use in high performing disaggregated platforms can be limited. In retrospect, the large bandwidth, and low loss profile associated with optical interconnects [15] makes their integration appealing in next generation disaggregated systems.

To meet the growth in demand for bandwidth in today's data center architectures, there has been a continuous progress in the development and the standardization of the next generation optical interconnect interfaces with data rates of up 200-400 Gb/s and transmission distances up to 10 km [31] exploiting wavelength division multiplexing (WDM). The use of WDM in these systems will alleviate the need for space consuming fiber ribbons used in contemporary pluggable interconnects. Like their predecessors, these interconnects will also be based on cost effective intensity modulation and direct detection (IM-DD) schemes. However, to achieve a smooth migration from contemporary systems, they will employ a 4 level pulse amplitude modulation (PAM4) format. PAM4 makes the most use of the pre-existing physical layer architecture used in one off keying (OOK) systems whilst doubling the spectral efficiency.

Although these PAM4 based interconnect interfaces offer a significant bandwidth enhancement over existing interconnect technologies, they still lack the Tb/s scales required in the next generation high performing disaggregated systems. Luckily, In the recent years there has been a continuous effort in the identification of key technologies which could potentially play a role in increasing the net data rates over IM-DD based interconnect interfaces over the 200-400 Gb/s channels envisioned today [32]. Recently, there has been an immense level of progress in achieving and demonstrating higher data rates over data center interconnects operating in a single optical channel. A major shortcoming of the optical based interconnect architectures operating over a single independent optical channel is the limited bandwidth of the electro-optical devices available. This factor enforces the employment of spectral efficient modulation formats or the exploitation of other available dimensions such as the phase or the polarization. Some examples of potential spectrally efficient modulation formats over OOK based schemes are PAM, Duobinary modulation, discrete multi tone (DMT) and carrier-less amplitude phase (CAP) [33]. However, it should be stated that DMT and CAP modulation formats suffer from a poor receiver sensitivity and they require complex DSP subroutines [34]. On the other hand, even though the density and the power consumption of coherent transceivers has been approaching transceivers used in data centers [35], the system complexities associated with these subsystems could possibly prohibit their integration in cost effective optical interconnects in the near future. So far, signaling rates over 200 Gb/s have been demonstrated over a single optical channel using PAM 4 [33], [36]. This data rates were further extended over 300 and 400 Gb/s [37], [38], however, most of this extensions required the exploitation of other available dimensions such as phase and polarization. Despite the impressing throughputs achieved in these recent serial transmission experiments, these trials still fail to meet the Tb/s scales required for next generation disaggregated systems. To remedy this, multiple optical channels operating at such high signaling rates can be multiplexed in a single transceiver [39], [40] to achieve the required Tb/s scales.

It should be noted that due to the bandwidth limitations associated with directly modulated lasers or external modulators, nearly all recent single channel and multi channel demonstrations above 50 Gb/s except for a back to back 71 Gb/s OOK based demonstration [41] failed to achieve an error free performance. Thus, most of these high capacity links heavily relied on forward error correction (FEC) and intensive digital signal processing (DSP) subroutines to alleviate the impact of distortions imposed by the bandwidth limited link. Unfortunately, the latency induced by the power consuming FEC modules at the receiver end can render these high capacity systems incapable of meeting the low latency profiles required by the high performing disaggregated architectures. This factor enforces the need for *FEC free* interconnects which requires the employment of large number of channel operating at low data rates. The use of low signaling rates per channel reduces the impact of the signal distortions imposed by the band limited electro-optical devices and the dispersive channel. A major draw back from such architectures housing a high channel count is the large inventory of independent devices such as drivers, lasers, modulators and detectors which they need to support. This drawback can lead to a high levels of power consumption and space inefficiency [37]. However, the photonic integration of these large scale architectures can allow for the attainment of highly dense and energy efficient transceiver with a small footprint [42].

Table I presents a set of recent trials and products demonstrating the performance of multi channel photonic integrated transceivers for IM-DD based optical interconnects. All of these demonstrations were based on the mature OOK signaling format apart from [16] and [19] which employ the spectrally efficient DMT format for achieving high signaling rates per channel whilst keeping the channel count low. The remaining trials listed, irradiated the need for a FEC coding and intensive DSP by achieving bit error rate (BER) thresholds at or below 10^{-12} , thus, the latency profile from these remaining devices is suitable for the disaggregated systems in question. These trials propose channel multiplexing both at the spatial (SDM) and frequency domain (WDM). Although WDM can significantly increase the space efficiency of the optical link, for high channel counts, the need for multiplexers and de-multiplexers as well as light sources with stable emission frequencies can cause complexities in the system. Thus, for a transceivers with a larger channel count, spatial multiplexing is preferred. A major disadvantage of using spatial based multiplexing is the space inefficiency which could result form interconnecting two N channel transceivers by N independent fiber links. However, by exploiting the breakthroughs in multi core fiber (MCF) architectures [15] it can be possible to accommodate such spatially multiplexed channels in a space efficient manner [42]. As it can be seen in Table I channel counts up to 168 had been demonstrated allowing for data rates up to 1.3 Tb/s to be reached on a single transceiver. Even though, the power consumption for this particular transceiver is rated at 13.4W [30] which is similar to the transceiver operating at 672 Gb/s, the energy consumption per bit is lower then the transceiver operating at 672 Gb/s. This observation is an indicator of power savings possible by exploiting optical integration. Moreover, its also clear that by integration of optics, it is possible to maintain a small footprint while increasing the throughput towards Tb/s scales. This factor is highlighted in the bandwidth densities achieved in demonstrations listed in Table I, where bandwidth densities up to 64 Gb/s/mm^2 were achieved. Table I also refers to the operational wavelength and the type of fiber employed in each trial. The photonic integration of such high throughput transceivers can significantly increase the faceplate density of a typical 1-RU (482.60mm x 44.45mm) Rack. However,

TABLE I
RECENT DEMONSTRATIONS OF LARGE CHANNEL COUNT INTEGRATED TRANSCEIVERS. (BOLD TEXT: (PRE)-COMMERCIAL; BTB: BACK TO BACK)

Channel x Bitrate	**Bandwidth Density (Gb/s/mmˆ2)**	**Energy Efficiency (pJ/bit)**	**Net data rate (Gb/s)**	**Multiplexing**	**Reach(km)/BER**	**Optical Link**	**Ref.**
4 x 66-77 Gb/s	9.1 - 10.6	*	264-308	WDM	1 / 1.5x10^{-2}	1550nm/SMF	[16]
8 x 25 Gb/s	**0.2**	**22.5**	**200**	**Spatial**	**2 / 1x10^{-12}**	**1310nm/SMF**	**[17]**
10 x 25 Gb/s	6.25	*	250	WDM	*	1550nm/SMF	[18]
10 x 83-88 Gb/s	*	*	830-880	WDM	4 / 3.8x10^{-3}	1550nm/SMF	[19]
12 x 25Gb/s	*	2	300	Spatial	BTB / 1x^{-12}	1550nm/SMF	[20]
12 x 12.5Gb/s	*	*	150	Spatial	BTB / 1x^{-12}	MMF	[21]
12 x 28.4Gb/s	**0.6**	**17**	**340**	**Spatial**	*	**850nm/MMF**	**[22]**
12 x 28Gb/s	**1**	*****	**336**	**Spatial**	**0.1 / 1x10^{-15}**	**850nm/MMF**	**[23]**
12 x 25Gb/s	*****	*****	**300**	**Spatial**	**0.07 / ***	**MMF**	**[24]**
12 x 25Gb/s	**0.37**	**10**	**240**	**Spatial**	**BTB / 1x10^{-12}**	**850nm/MMF**	**[25]**
12 x 10Gb/s	**0.187**	*****	**120**	**Spatial**	**0.1 / 1x10^{-12}**	**850nm/MMF**	**[26]**
24 x 28Gb/s	1	13.4	672	Spatial	* / 1x10^{-12}	*	[27]
24 x 15Gb/s	1	*	360	Spatial	0.1 / *	850nm/MMF	[28]
24 x 20Gb/s	15.9	7.3	480	Spatial	* / 1x10^{-12}	850nm/MMF	[29]
168 x 8Gb/s	64	10	1344	Spatial	0.3 / 1x10^{-12}	1000nm/MMF	[30]

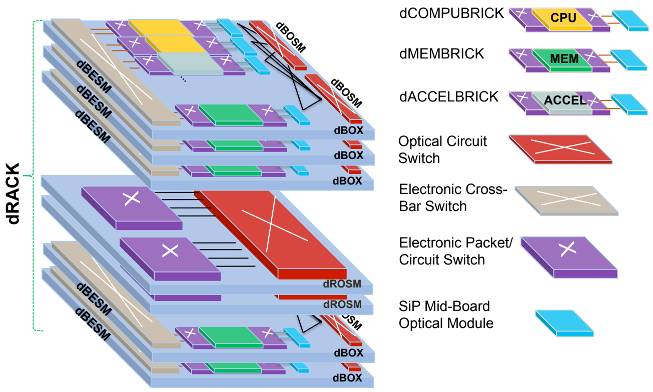


Fig. 1. dRedBox rack-scale architecture interconnected with hybrid optical and electrical switching.

the high losses associated with electrical cabling is enforcing the integration of these optical transceivers closer to the key building blocks of next generation disaggregated systems as on board transceivers as oppose to using these devices as front panel pluggable transceivers. The integrated optics introduced in Table I can be employed as on board transceivers, but, there are already a number of (pre)-commercial vendors currently offering such on board integrated technologies as highlighted in bold text in Table I. However, the data rates of these low latency on board integrated technologies are limited to 300 Gb/s but it can be foreseen that bandwidths at Tb/s scales will be available commercially in coming years.

In this work, in order for to meet the criteria that was defined earlier in terms of latency, bandwidth, power consumption, bandwidth density and footprint we will exploit a commercial Silicon-photonic (SiP) (second row of Table I) optical on board transceiver operating in single-mode for the optical interconnects used in dRedBox architecture. This particular transceiver is composed of eight spatially multiplexed optical channels as presented in Table I ([27]). The use of such transceivers operating in single-mode as opposed to multi-mode allows for a scalable network operating based on optical circuit switching (OCS), since the use of single mode fiber allows for optical switches with higher port counts [27]. It also offers the longest reach or having the highest power budget while delivering $1x10^{12}$ FEC free BER. This factor suits the dRedBox architecture, as it uses OCS to deliver scale-out optically transparent architectures.

IV. Architecture

This section introduces the dRedBox architecture at rack and cluster scale. It explains all elements across the network system and describes its ability to be function and topology reconfigurable in order to suit the diverse networking demands between all disaggregated computing building blocks.

A. dRedBox architecture

Fig. 1 presents the rack-scale disaggregated architecture proposed by the dRedBox project. As it can be seen, this particular architecture consists of dRacks (disaggregated Racks) housing multiple interconnected dBoxes. Each dBox hosts a) an arbitrary combinations of pluggable compute/memory/accelerator dBricks, b) an electronic cross-point circuit switch for intra dBox connectivity and c) a set of miniaturized optical switches for intra and inter dBox networking. Each dBox is a rack mounted 2U unit that will support up to 16 bricks. Each dBrick will either support general-purpose processing (dCompubrick) or random-access memory (dMembrick) or application-specific accelerator (dAccelbrick). All dBricks are interconnected to all other dBricks on the same dBox by means of the electronic L1 crosspoint circuit switch and the optical circuit switch. The optical circuit switch operates based on beam-steering technology [43].

Communication between various dBricks on different dBoxes is carried strictly via optical circuit switching. Crucially, each dBrick apart from its main information technology purpose (compute/memory/acceleration) uses a reconfigurable system on chip module to perform networking functions beyond just interfacing, as traditional network interface cards do. This aspect of dBricks will be further described in Section VII. Each individual dBrick can embed and support forwarding, switching, and aggregation at either packet or circuit level [3]. These dBricks can further provide protocol independent programmable ports to support protocols and functions that can best suit the type of communication required (i.e. compute-to-memory, compute-to-end user, etc).

To minimize footprint and power consumption while maximizing bandwidth-density as it was described in the previous section, each of the bricks use an on-board SiP transceiver. The employment of SiP transceivers and beam-steering based optical switches allow for a scalable, transparent and ultra-low latency brick-to-brick multi-hop communication platform which will be emphasized on more in Section VII.

B. dRedBox topology and function reconfiguration

Expanding the disaggregated system from a single dRack to a Data Centre, a scale-out network architecture needs to be considered. Such system also needs to address the diverse network requirements for the various types of communications i.e. CPU-to-memory, CPU-to-user, CPU-to-accelerator, CPU-to-storage. Some require extremely low latency communication (i.e. CPU-to-memory) where others need packet switching services following standard protocols i.e. Ethernet for CPU-to-user, or Infiniband for CPU-to-storage. This subsection focuses on addressing these challenges by proposing the combination of reconfigurable optical network topology and on-chip protocol programmability. This can support hybrid electronic packet switching and optical circuit switching supporting any arbitrary chain of programmable network functions and network graph (topology).

The dRedBox data centre architecture is based on the 3-tier topology as it is shown in Fig. 2. This particular topology considers optical switching modules called dBOSM (disaggregated Box Optical Switch Module), dROSM (disaggregated Rack Optical Swtich Module) and dDOSM (disaggregated Data Centre Optical Switch Module). The dBOSM optical

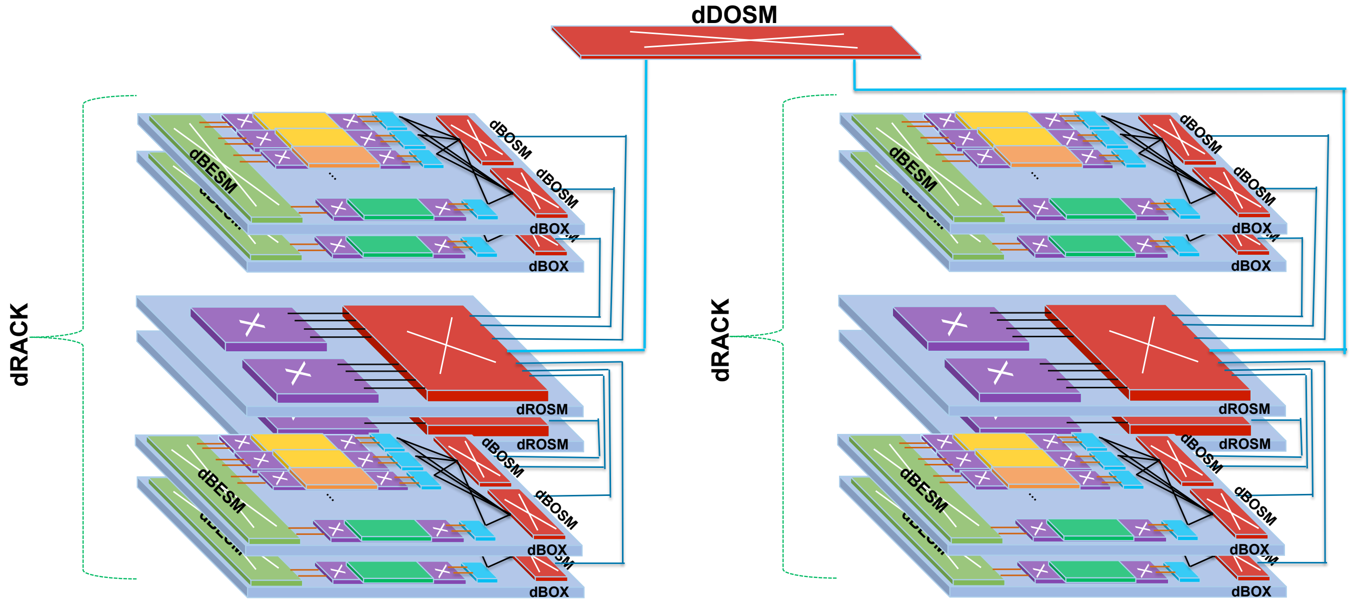


Fig. 2. Disaggregated cluster-scale Data Center network architecture.

switches at tier-1 accommodate for up to 16-dBricks. To deliver high level of modularity and reflect the subscription ratio requirements, these switches have a small port-count (i.e. 48 or 96) with highest density). On the other hand, the dROSM and dDOSM switches at tier 2 and tier-3 require a large port-count (i.e. 384x384). Tier-2 switch in addition to offering transparent connectivity between tier-1 and tier-3, it also provides access to pluggable electronic programmable packet switches. Packet switching services can be supported by use of programmable on-chip packet/circuit switching supported on dBrick or attached as pluggable modules on dBOSM.

The first and main reason for using a pure circuit switched network (either electrical for intra dBox communication or optical for intra/inter dBox communication) is to flatten the topology and once configured deliver lowest possible CPU to memory latency. This imitates the CPU bus structure which delivers the lowest level of deterministic latency and guaranteed bandwidth between processor and memory elements while offering dynamic reconfiguration to connect any number of compute dBricks to any number of memory dBricks. The proposed architecture explores the support of both DDR4 (parallel-based memory) and optically-attached HMC (serial-based memory) dBricks. However, it proposes the use of optically-attached HMC as the ideal candidate due to a) serial communication and inherent compatibility with high-speed interconnect and network operation b) ability to deliver very low latencies as described below and c) elimination of additional chip (e.g. ASIC, FPGA, MPSoC) at memory dBrick to host memory controller as per DDR4 case that in turn reduces cost, power consumption and footprint.

The round trip network latency between dBricks I/Os using either DDR4 or HMC memory elements and across different network infrastructures is illustrated in Fig. 3. It reflects a read/write transaction between processor dBrick and memory dBrick within a) the same dBoX, b) between different dBoxes of the same the dRack and c) between dBoxes of different dRacks. Also the overhead (additional latency in % points) compared to a direct-attached system using HMC or DDR4 memory dBrick is portrayed in this figure. The direct-attached HMC round trip latency (base latency) including the transceiver (i.e. serial/de-serializer) and protocol stack accounts for a total latency of 180 nanoseconds [44]. The direct attached round trip DDR4 latency is assumed to be 50 nsec. The latency contributions of conventional data centre system and network technologies reflected on case A on Fig. 3 correspond to 350 nsec for NIC (including PCIe and transceiver) and load dependent 300-600 nsec for EPS switching. This approach imposes very high latencies (0.9-5.1 usec) for either HMC or DDR4 implementations and in turn deem them unfavorable for memory disaggregation.

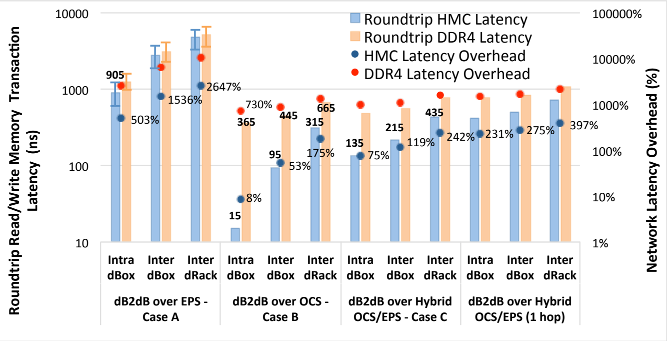


Fig. 3. Theoretical round-trip (memory read/write transaction) network latency (excluding base/direct-attach latency) and overhead on HMC and DDR4 base (direct-attached) latencies. *dB2dB: dBrick to dBrick

Further assumptions used here to calculate the latencies are: a) a link equal to 0.25 meter between the dBrick and dBOSM modules, b) a 3 meter link between dBOSM and dROSM, c) 10 meter link between dROSM and dDOSM, and c) pass-

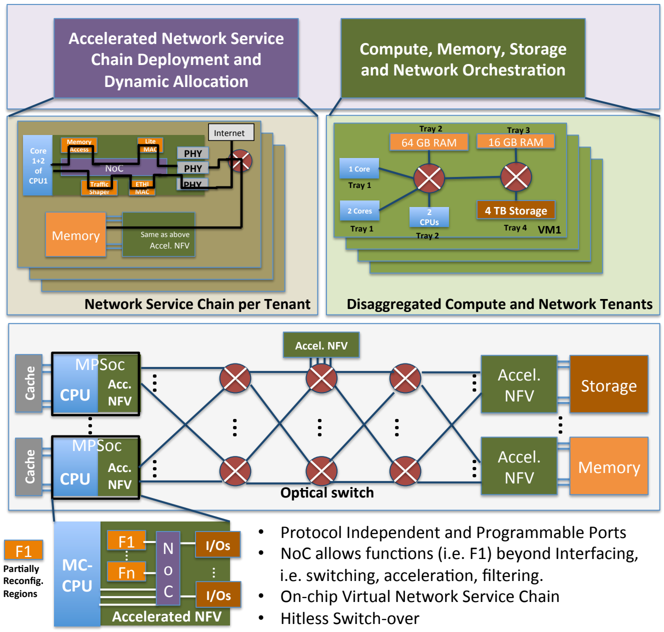

Accelerated Network Service Chain Deployment and Dynamic Allocation
Compute, Memory, Storage and Network Orchestration
Core 1+2 of CPU1
Memory Access
Lite MAC
NoC
Traffic Shaper
ETH MAC
PHY
PHY
PHY
Internet
Memory
Same as above Accel. NFV
Network Service Chain per Tenant
Tray 2
64 GB RAM
Tray 3
16 GB RAM
1 Core
Tray 1
2 Cores
Tray 1
2 CPUs
Tray 2
4 TB Storage
Tray 4
VM1
Disaggregated Compute and Network Tenants
Cache
MPSoc
CPU
Acc. NFV
Cache
MPSoc
CPU
Acc. NFV
Accel. NFV
Optical switch
Accel. NFV
Storage
Accel. NFV
Memory
F1
Partially Reconfig. Regions
MC-CPU
F1
Fn
N o C
I/Os
I/Os
Accelerated NFV
Protocol Independent and Programmable Ports
NoC allows functions (i.e. F1) beyond Interfacing, i.e. switching, acceleration, filtering.
On-chip Virtual Network Service Chain
Hitless Switch-over

Fig. 4. Architecture for composition of disaggregated compute and network functions.

through optical circuit switch (OCS) latency of 5ns. This proposed approach of using purely OCS (case B on Fig. 3) indicates only a 15 ns overhead when transactions remain within the boundaries of dBox, the additional latencies of 95 nsecs and 315 nanoseconds are expected when its required to scale out to dRack and a dCluster (Cluster of Racks) level. Compared to the case A overheads this approach delivers substantial latency reductions and is mostly limited by the speed of light in silica based fibre. These correspond to overheads (against direct-attached systems) of 8%, 53% and 175% when using HMC and 730%, 890%, and 1330% when using DDR4 with traditional PCIe-based NIC. When the on-chip switch as per [3] is used on dBricks (case C on Fig. 3) to increase flexibility (allow dynamic mapping of compute to memory dBricks) that contributes 60 ns latency, then the overall latency increases to 135, 215 and 435 ns for transactions within dBox, dRack and the dCluster respectively. Additional compute to memory latencies due to network functions introduce additional cost to the system since based on [10] more powerful processors might be needed to compensate processing and application performance. Otherwise the system and application performance might degrade. However, the on-chip switch can be used to support other non-latency sensitive network services such as packet forwarding, grooming, switching for CPU-to-user, CPU-to-Storage networking as will be described later on in this Section. More information on the cost model we propose, an adaptation of Abali's model [10], together with the impact of latency on the overall cost is described in Section VI.

The subsequent reason for using OCS is it's ability to allow scalability within architectures. Furthermore, the use of single-mode ultra-wideband optical switches with ultra-low insertion loss i.e. 1dB/cross-connection in conjunction with single mode fibers allows for the maximum number of transparent hops through a system that allows the architecture to scale-up and scale-out. Further information and results on experiments conducted using SiP on-board transceivers and single-mode optical switches is provided in Section VII. Another equally important benefit of using optical circuit switching is that the proposed architecture considers the dBricks (end-points) not as pure compute, memory or accelerator units with simple input/output interfaces but as elements that also provide a range of networking functionalities i.e. circuit/packet switching, transaction forwarding, monitoring, parsing, queuing, acceleration and diverse protocol support. A combination of functions such as service chain, can be composed [45] and linked to a set of ports available on a single Multi-Processor System on Chip (MPSoC). Thus, the combination of optical circuit switching in the core of the network and the hardware programmable on-chip switching for each of the end points allows for an efficient function and topology reconfiguration as was reported in [4]. This proposed architecture outperforms in blocking probability, cost and power consumption the conventional hybrid packet/circuit architecture such as HELIOS [4]. In particular, this combination creates a highly flexible and deeply programmable architecture that can serve the diverse networking needs of CPU-to-CPU, CPU-to-Memory, CPU-to-Storage, etc. transactions when pooling together all disaggregated resources to serve a VM request. For example Fig. 4 illustrates that such architecture can be composed and programmed to support a) ultra-low latency on-chip circuit switching with lite protocol to carry memory mapped data and optical circuit switching graph/topology for N CPU to M Memory units associated with enough ports to satisfy bandwidth needs, b) on-chip Ethernet based packet switching to carry VM data to other VMs or end-users again on certain ports, c) storage area network or network attached storage protocols, i.e. FibreChannel, supported on other port(s) to serve CPU to Storage data transfers.

Finally, the scalability of the proposed architecture in terms on number of ports per dBrick, dBricks, dBoxes and dRacks that can be supported relates the number of a) ports per dBrick (PPB), b) dBricks per dBox (BPB), c) dBoxes per dRack (BPR), d) dRacks e) the port number of the optical switches (PPS) used and f) the (over)-subscription ratio (i.e. N for level 1, M for level 2). These parameters then determine the number of dBOSM, dROSM and dDOSM switches required. The equations can be expressed as:

$$\begin{aligned} dBOSM\ number\ per\ dBox &= PPB \times BPB \\ &\times (1 + \frac{1}{N})/PPS \end{aligned} \tag{1}$$

$$\begin{aligned} dROSM\ number\ for\ single\ dRack &= PPB \times BPB \\ &\times \frac{1}{N} \times BPR/PPS \end{aligned} \tag{2}$$

$$\begin{aligned} dDOSM\ Number = PPB \times\ \ BPB \times \frac{1}{N} \times BPR \\ \times \frac{1}{M} \times dRack\ number/PPS \end{aligned} \tag{3}$$

Fig. 5 illustrates the effect of the (over)-subscription ratio on the number of dROSMs within a single dRack. Transparently,

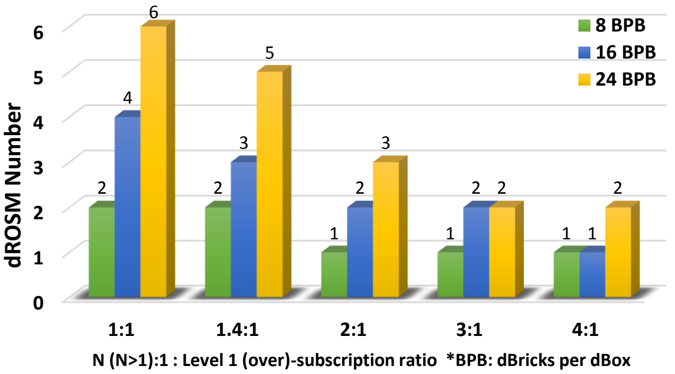


Fig. 5. dROSM number for single dRack (384*384 switch, 16 ports per dBrick, 12 dBoxes per dRack)

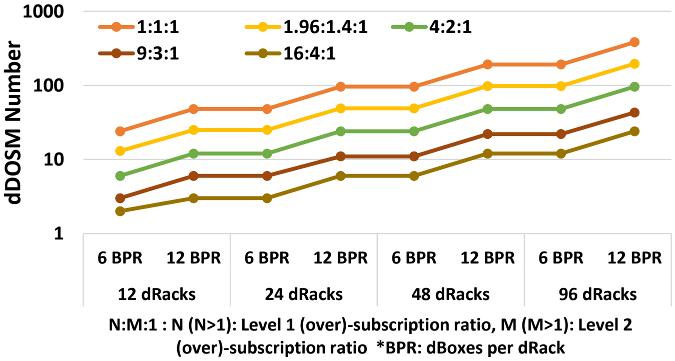


Fig. 6. Overall dDOSM number for data centre with multi-dRacks (384*384 switch, 16 ports per dBrick, 16 dBricks per dBox)

the number of dROSMS is proportional to the number of ports per dBrick, while the results reverse for that of the (over)-subscription ratio. In other words, the higher the ratio, the less the dROSMs are needed.

Fig. 6 depicts the numbers of dDOSMs required for a data centre cluster in a spine-leaf dROSM-dDOSM configuration consisting of 12, 24, 48 and 96 dRacks. Similar to the number of dROSMs, the number of dDOSMs is inversely proportional to the (over)-subscription ratio. Moreover, results indicate that larger scale data centres with greater number of dBoxes per dRack or/and dRacks can be easily supported by adding more swithes (i.e. dBOSM, dROSM and dDOSM).

V. Composition Algorithms

A. Proposed algorithms

A simulator in Matlab has been developed to investigate the performance of the disaggregated data centres considered in the dRedBox architecture. This simulator performs coordinated orchestration and allocation of IT resources together with the reservation of their network bandwidth and interconnection to serve VM requests. The overall resource allocation process is depicted in Fig. 7. Resources are reserved for a VM request only when sufficient IT (CPUs, memory and storage) and network resources are found, otherwise, the request will be dropped.

Maximizing IT and in particular CPU/memory resource utilization is one of the main goals of disaggregation. As such, a well-designed resource allocation algorithm is needed to boost the efficiency of the system and decrease the network load by selecting appropriate sets of IT resources. In this paper, four IT resource allocation algorithms are developed and compared:

First Fit (FF) resource allocation algorithm is the simplest algorithm developed. With this algorithm, resources are allocated on a first-fit basis, which indicates that every request will be allocated to the first available node(s) (dBricks) during the resource identification procedure. Since network availability is not considered during the allocation process, it is possible that the chosen nodes have sufficient IT resources but the links between them become saturated or unable to satisfy the bandwidth requirement. As a consequence, high blocking probability, particularly due to the algorithm failure when allocating network resources, will be experienced.

Best Fit (BF) resource allocation algorithm can be seen as an improved algorithm compared to the the previous one. The best possible combination of IT resource types are selected for every request since different types of resources (CPUs, memory or storages) are searched and allocated independently (using different data structures). The main advantage of this algorithm is that it can jump to different racks for every resource type which is not possible with the FF algorithm since all IT resources are managed using a single data structure. With this approach, the blocking probability can be significantly reduced.

Network-Unaware Locality Based (NULB) resource allocation algorithm realizes globally optimized IT resource allocation. The pseudo code for this scheme is presented in in Algorithm 1, where the data center graphs (G) contains the information of the nodes and link configurations. Data center config (C) includes the IT and network resource information.

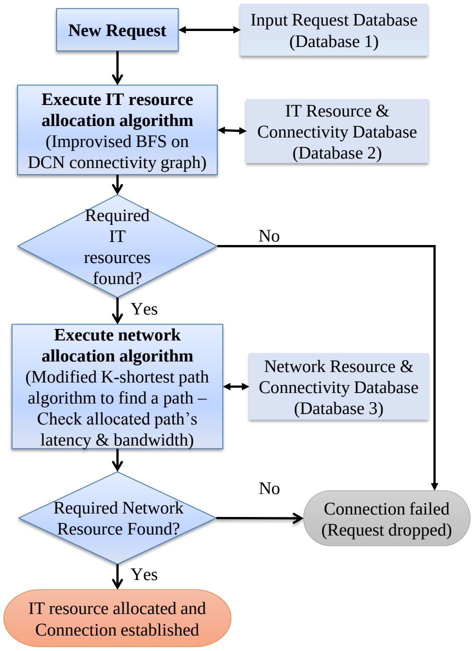

New Request
Input Request Database (Database 1)
Execute IT resource allocation algorithm (Improvised BFS on DCN connectivity graph)
IT Resource & Connectivity Database (Database 2)
Required IT resources found?
No
Yes
Execute network allocation algorithm (Modified K-shortest path algorithm to find a path – Check allocated path's latency & bandwidth)
Network Resource & Connectivity Database (Database 3)
Required Network Resource Found?
No
Connection failed (Request dropped)
Yes
IT resource allocated and Connection established

Fig. 7. A simple algorithmic flowchart illustrating the steps involved in the resource allocation process.

The concept of contention ratio (CR) for each type of IT resources is introduced, which relates to the ratio between units required over the total units available (line 5-7). High CR indicates that this type of IT resource is highly demanded. Once a request comes, the algorithm will start scanning for IT resources on a node (dBrick) of a specific resource type that has the highest CR (line 8) and then search other types of IT resources around the allocated nodes via breath-first search (BFS) algorithm (line 20, line 45 and line 70). BFS is an algorithm for traversing and searching tree and graph data structures. It is adopted to find IT resource nodes (dBricks) that are neighboring each other, which makes it locality aware. Especially, it is particularly useful for requests with low latency constraints (i.e. CPU to memory).

Algorithm 1 NULB resource allocation algorithm

```
Require: $G$: Data centre graphs, $C$: Data centre config, $R$: Request
 1: procedure NULOCALITYBASED($G, C, R$)
 2:     $available.Cpu \leftarrow$ find slots $\geqslant 1$ in $C.CpuLocations$
 3:     $available.Mem \leftarrow$ find slots $\geqslant 1$ in $C.MemLocations$
 4:     $available.Sto \leftarrow$ find slots $\geqslant 1$ in $C.StoLocations$
 5:     $cr.Cpu \leftarrow \frac{R.Cpu}{available.Cpu}$                ▷ Cpu contention ratio
 6:     $cr.Mem \leftarrow \frac{R.Cpu}{available.Mem}$                ▷ Memory contention ratio
 7:     $cr.Sto \leftarrow \frac{R.Cpu}{available.Sto}$                ▷ Storage contention ratio
 8:     $cr.Max \leftarrow max(cr.Cpu, cr.Mem, cr.Sto)$        ▷ Find maximum contention ratio
 9:     $loopIncrement \leftarrow C.nSlots \times C.nBlades$        ▷ Size of a rack
10:     for each $c$ in $cr$ do        ▷ To try multiple contention ratios
11:         $s \leftarrow cr.Max$
12:         $cr.Max \leftarrow$ Update to new maximum contention ratio
13:         switch $s$ do
14:             case $Cpu$
15:                 if $available.Cpu < R.Cpu$ then        ▷ Inadequate cpu resources
16:                     $ITallocation \leftarrow Failure$
17:                     $ITfailureCause \leftarrow Cpu$
18:                 else
19:                     while $slot \leqslant C.CpuResources$ do
20:                         $[ITallocation, ITres] \leftarrow$ BFS($G, slot, R$)
21:                         if $ITallocation = Success$ then
22:                             $[NETallocation, NETres] \leftarrow$ NETALLOCATION($G, C, R, ITres$)
23:                             if $NETallocation = Success$ then
24:                                 break
25:                             else
26:                                 Remove failure nodes in $G.ITres$
27:                             end if
28:                         else
29:                             Evaluate failure cause
30:                             Update $ITfailureCause$
31:                             break
32:                         end if
33:                         $slot \leftarrow slot + loopIncrement$
34:                     end while
35:                 end if
36:                 if $ITallocation = Success$ and $NETallocation = Success$ then
37:                     break
38:                 end if
39:             case $Mem$
40:                 if $available.Mem < R.Mem$ then        ▷ Inadequate memory resources
41:                     $ITallocation \leftarrow Failure$
42:                     $ITfailureCause \leftarrow Mem$
43:                 else
44:                     while $slot \leqslant C.MemResources$ do
45:                         $[ITallocation, ITres] \leftarrow$ BFS($G, slot, R$)
46:                         if $ITallocation = Success$ then
47:                             $[NETallocation, NETres] \leftarrow$ NETALLOCATION($G, C, R, ITres$)
48:                             if $NETallocation = Success$ then
49:                                 break
50:                             else
51:                                 Remove failure nodes in $G.ITres$
52:                             end if
53:                         else
54:                             Evaluate failure cause
55:                             Update $ITfailureCause$
56:                             break
57:                         end if
58:                         $slot \leftarrow slot + loopIncrement$
59:                     end while
60:                 end if
61:                 if $ITallocation = Success$ and $NETallocation = Success$ then
62:                     break
63:                 end if
64:             case $Sto$
65:                 if $available.Sto < R.Sto$ then        ▷ Inadequate storage resources
66:                     $ITallocation \leftarrow Failure$
67:                     $ITfailureCause \leftarrow Sto$
68:                 else
69:                     while $slot \leqslant C.StoResources$ do
70:                         $[ITallocation, ITres] \leftarrow$ BFS($G, slot, R$)
71:                         if $ITallocation = Success$ then
72:                             $[NETallocation, NETres] \leftarrow$ NETALLOCATION($G, C, R, ITres$)
73:                             if $NETallocation = Success$ then
74:                                 break
75:                             else
76:                                 Remove failure nodes in $G.ITres$
77:                             end if
78:                         else
79:                             Evaluate failure cause
80:                             Update $ITfailureCause$
81:                             break
82:                         end if
83:                         $slot \leftarrow slot + loopIncrement$
84:                     end while
85:                 end if
86:                 if $ITallocation = Success$ and $NETallocation = Success$ then
87:                     break
88:                 end if
89:             if $ITallocation = Failure$ then        ▷ Inadequate IT resources
90:                 break
91:             end if
92:         end switch
93:     end for
94:     if $ITallocation = Success$ and $NETallocation = Success$ then
95:         Update $C.ITres$        ▷ Remove allocated IT resources
96:         Update $G.bMap$        ▷ Remove allocated network resources
97:     end ifreturn $ITallocation$, $ITres$, $NETallocation$, $NETres$
98: end procedure
```

Network-Aware Locality Based (NALB) resource allocation is an optimization of NULB algorithm by utilizing the modified breadth-first search (mBFS) algorithm instead of BFS algorithm (line 20, line 45 and line 70 in Algorithm 1). The mBFS considers the network resources (bandwidth available) when searching the nearby IT resources nodes. In this way, the performance of the network can be improved considerably since the probability of failure when allocating network resources is reduced. The pseudo code for mBFS is provided in Algorithm 2.

Algorithm 2 Modified breadth-first search (mBFS) algorithm

```
Require: $G$: Data centre graphs, $s$: Start vertex, $R$: Request
procedure MBFS($G, s, R$)
    $bMap \leftarrow G.bMap$                      ▷ Copy original bandwidth map
    $dMap \leftarrow G.dMap$                      ▷ Copy original distance map
    Remove all edges in $bMap$ and $dMap$ where $e_b < min(R.b_{cm}, R.b_{ms})$
    for each vertex $v$ in $G$ do                 ▷ Initialise each vertex
        $v.distance \leftarrow \infty$
        $v.parent \leftarrow null$
    end for
    $Q \leftarrow [\,]$                           ▷ Create an empty queue
    $s.distance \leftarrow 0$
    $Q.enqueue(s)$                                ▷ Enqueue start vertex
    $ITres \leftarrow$ Find IT resources on start vertex $s$
    while $Q \neq [\,]$ do                        ▷ Run until queue is empty
        $current \leftarrow Q.dequeue()$
        $neighbours \leftarrow$ All vertices adjacent to $current$ ▷ High bandwidth links have a higher priority (Network aware)
        $neighbours' \leftarrow$ Sort $neighbours$ in descending order
        for each vertex $v$ in $neighbours'$ do
            if $v.distance = \infty$ then
                $v.distance \leftarrow current.distance + 1$
                $v.parent \leftarrow current$
                $Q.enqueue(v)$                    ▷ Enqueue $v^{th}$ neighbour
                $ITres \leftarrow$ Find IT resources on vertex $v$
                if $R.Cpu$ and $R.Mem$ and $R.Sto$ found then ▷ All resources found
                    $breakWhile \leftarrow True$
                    break
                end if
            end if
        end for
        if $breakWhile = True$ then
            $ITallocation \leftarrow Success$
            $ITfailureCause \leftarrow None$
            break
        else
            $ITallocation \leftarrow Failure$
            $ITfailureCause \leftarrow$ Evaluate failure cause
            break
        end if
    end whilereturn $ITres, ITallocation$
end procedure
```

The network resource allocation process is implemented after the IT resource allocation. In this paper, a *modified K Shortest Path Algorithm* is developed. Based on the graph theory, we use a new weight factor (W) instead of distance (original weight factor) in the traditional *K Shortest Path Algorithm*, which can be expressed as:

$$W = f \times (1 - \frac{available\ bandwidth\ of\ the\ current\ link}{max\ bandwidth\ of\ one\ link}) + (1-f) \times \frac{distance\ of\ the\ current\ link}{max\ link\ distance} \quad (4)$$

where f is a variable between 0 and 1. The value close to 1 means bandwidth is favored, whereas a value close to 0 means distance/latency is favored. This algorithm was utilized for network resource allocation in all the simulations and both network factors were weighted equally (f = 0.5).

VI. Simulations

A. Simulation environment and assumptions

In this section the performance of the disaggregated data centre (DDC, dRedBox architecture) and the traditional data centre (TDC, server-centric architecture) is compared with the simulator mentioned above. In particular, three type of disaggregated rack structures are investigated; a) Homogeneous dRack and homogeneous dBoxes (S1, each dRack can only host one type of dBrick), b) Heterogeneous dRack with homogeneous dBoxes (S2, each dRack can support multiple types of dBricks but only one type per dBox) and c) Heterogeneous dRack with heterogeneous dBoxes (S3, each dBox can host any type of dBrick). Each of these structures has a 1/3 ratio of compute, memory and storage resources across the whole multi-Rack system.

Table II describes the IT and network requirements for a single request and the data centre configurations. To simulate different application environments, we assume six types of VM requests demanding varying number of CPUs and RAMs. All requests are dynamically generated following a Poisson distribution with an average inter-arrival time of 10 time units. Each request contains the information of CPU core number, RAM size, storage size (128 GB for each request), CPU-RAM latency & bandwidth required, RAM-STO latency & bandwidth required and holding time. The holding time starts from 6300 time units and increases 360 time units for every 100 requests.

B. Simulation results

1) Comparison of algorithms and structures in DDC: Since three type of dRacks and four IT resource allocation algorithms are considered in DDC, we compare their combinations first to identify the best one. Fig. 8 presents the maximum CPU and network utilization for each combination as well as the corresponding number of blocked VM request after processing 1000 requests.

In terms of blocking probability, the BF, NULB and NALB algorithms provide similar performance which is much better than that of the FF algorithm. Among them, the BF algorithm is unable to maximize CPU utilization (maximal 91%) for minimum network resource usage. Both NULB and NALB algorithms obtain nearly 100% CPU utilization combined with S2 and S3. To clarify the best option among these four combinations we can only see that the NALB-S3 requires the least network resources to achieve the same CPU utilization, this keeps the added network cost and complexity to the minimum.

The Cumulative Distribution Function (CDF) of the round-trip (memory read/write transaction) network latency for all algorithm-structure combinations is presented in Fig. 9. Transparently, S2 and S3 perform much better than S1 for each

TABLE II
SIMULATION ASSUMPTIONS AND VM RESOURCE REQUEST PROFILES

Configurations	12 dRacks	6 dBoxes/dRack		8 dBricks/dBox	16 units/dBrick	
4 cores/CPU unit		4 GB/RAM unit		64 GB/Storage Unit	8 I/Os per dBrick @ 25Gb/s	
Request Requirements	Type 0	Type 1	Type 2	Type 3	Type 4	Type 5
	random	high RAM	high CPU	half & half	more RAM	more CPU
CPU&RAM	1-32 core, 1-32 GB	1-8 core, 24-32 GB	24-32 core, 1-8 GB	Type 1 (50%), Type 2 (50%)	1-16 core, 17-32 GB	17-32 core, 1-16 GB
CPU-RAM Bandwidth (fixed for all requests)		5 Gb/s/unit		RAM-STO Bandwidth (fixed for all requests)	1 Gb/s/unit	
CPU-RAM Round-trip Latency		180-480 ns		RAM-STO Round-trip Latency	480-780 ns	

algorithm. Particularly, S3 realizes 67% (average) latency reduction on S1 (from 238 ns to 77.6 ns). The reason behind this is that the heterogeneous rack configuration holds most of the traffic within one rack (e.g. 86% and 85% of the traffic with NALB-S2 and NALB-S3 respectively) whereas at least three racks should be involved for serving one request with S1, indicating more inter-rack traffic in DCN. Similarly, as heterogeneous dBoxes are configured in S3, 174% more traffic is generated within the dBoxes than that of S2 with the NALB algorithm. This factor can also be used to explain the reduction in network utilization of NALB-S3 compared to NALB-S2 when all CPUs are utilized in Fig. 8. In terms of algorithm with S3, NULB and BF algorithms outperform the FF algorithm, the NALB algorithm seems to be the best choice, which holds highly 63% of the traffic within the dBoxes and achieves 61%, 58% and 52% latency reduction on FF algorithm, BF algorithm and NULB algorithm respectively. To sum up, the structure of the DCN and in particular the placement of resources plays an important role on reducing the latency. NALB-S3 will be utilized as the benchmark for identifying the advantages of disaggregation against traditional server-centric architecture.

2) Comparison of blocking probability: Fig. 10 shows the blocking probability of DDC and TDC under six different types of input requests described in Table II. Results show that the 575th request is the first blocked request in TDC with type 0 requests and the blocking probability rises afterwards as the holding time increases with request number. In contrast, no blocking occurs until the 925th request in DDC. Since identical amount of CPU units and memory units are provided in both types of data centres, the two schemes with high demand of one specific IT resource (type 1 and type 2) provide similar performance and the worst among all other types. This is due to disproportionate amount of memory requested in case of type 1 and CPU in type 2 to the total available resources. Although high blocking probability is obtained in both cases, DDC still outperforms TDC. When the previous two request types are combined to one, type 3, to create a more diverse scenario the blocking probability of DDC is considerably reduced. Moreover, the results show an 88% increase in request number in the DDC compared to the TDC architecture using type 3 requests with a blocking probability under 0.01. The performance of DDC under type 4 and type 5 requests appear to be similar and considerably better than TDC. To sum up, resources are more efficiently utilized with type 0 and type 3 requests, and DDC performs better than TDC under all types of input request in terms of the blocking probability.

3) Cost efficiency analysis : To clarify the benefits of disaggregation comprehensively, a cost model has been proposed by

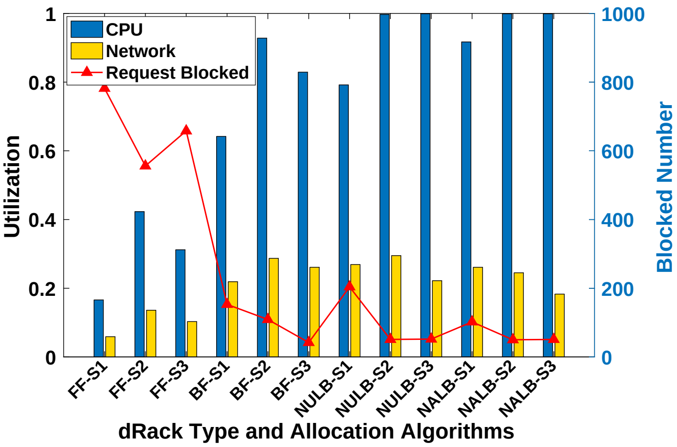


Fig. 8. Comparison of dRack types using four allocation algorithms

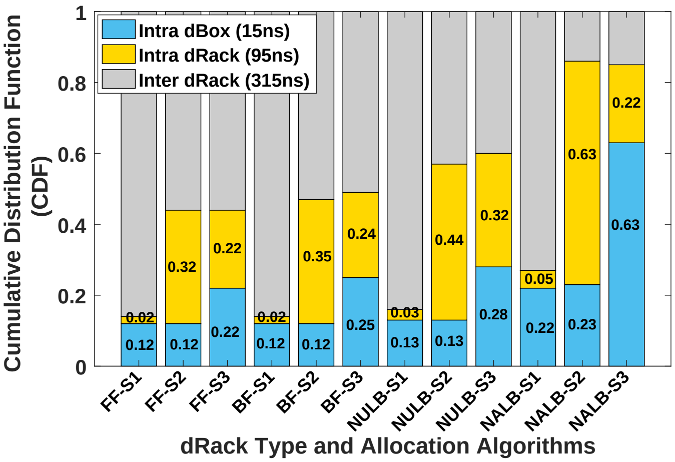


Fig. 9. CDF of the round-trip (memory read/write transaction) network latency for all combinations

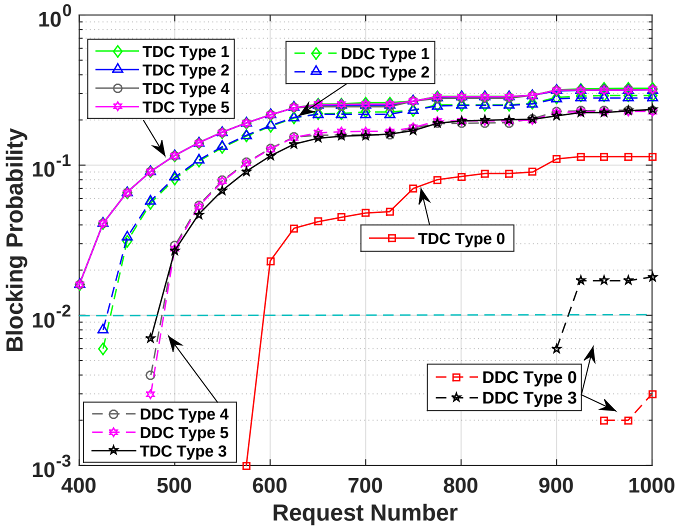


Fig. 10. Comparison of blocking probability using six different input requests

modifying the model reported in [46] which can be expressed as:

$$G = CS + MS - (CN + CL) \tag{5}$$

$$\begin{aligned} CS = CPU\ Utilization\ Difference \\ \times Total\ Amount\ of\ CPU \times CPU\ Unit\ Price \end{aligned} \tag{6}$$

$$\begin{aligned} MS = RAM\ Utilization\ Difference \\ \times Total\ Amount\ of\ RAM \times RAM\ Unit\ Price \end{aligned} \tag{7}$$

$$\begin{aligned} CL = 310 \times \frac{Increased\ Latency}{Base\ Latency} \times \\ CPU\ Utilization \times Total\ Amount\ of\ CPU \end{aligned} \tag{8}$$

Where G is the net gain achieved by disaggregation. Since one of the main benefits of disaggregation is the increased resource utilization, CS and MS denote the CPU savings and memory savings. In DDC, traffic between CPU and memory moves to the whole DCN scale, which requires high bandwidth supported from the network that also results in increased latency. As such, CN and CL represents the extra cost of network resources and latency of DDC comparing to TDC separately. CS and MS are calculated by Eq. 6 and Eq. 7 respectively. According to [46], CL can be calculated by Eq. 8, where *310* is a coefficient in the model (SPEC-FP on single threaded cores). CN equals to the sum of the costs for extra Tx & Rx and optical switch ports (channels).

Fig. 11 shows the resource utilization increase of DDC against TDC, which could be used for CS, MS and CN calculations. Fig. 12 depicts the total net gain of disaggregation under six type of input requests. Where the price of memory unit (4GB HMC), optical switch port and Tx & Rx per channel is $600, $100 and $1 per Gb/s separately. The (HMC) base latency is 180 ns. Evidently, DDC is more cost effective in almost all cases. Moreover, the value of net gain is proportional to the CPU price and the CPU price poses little influence on the net gain in low CPU utilization types, including type 1 and type 4. On the contrary, wide variation of net gains arise in the remaining types owing to the high CPU utilization increases shown in Fig. 11.

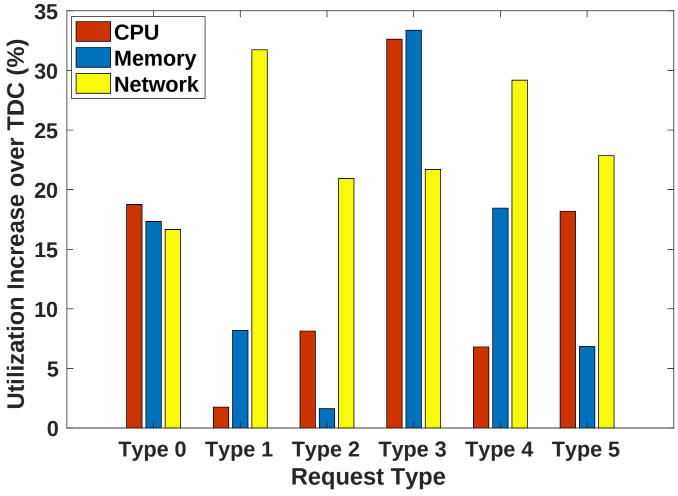


Fig. 11. Utilization increase of DDC over TDC

VII. Experiment and Results

In this section the description of the experimental setup developed in this work to verify the performance of the proposed disaggregated data center architecture employing SiP mid-board optics (MBO) and OCS is presented. Fig. 13 (a) illustrates the experimental setup used to evaluate the number of switching layers that the SiP MBOs can accommodate for in a multi tier disaggregated topology (Fig. 13 (b)) . The two MBO modules are both connected to an Xilinx FPGA, each signifying a dBrick element connected to a common optical switch emulating the dRedBox multi-tier network. Each individual SiP MBO is comprised of eight optical channels, which are multiplexed spatially by a MPO based fiber ribbon. The reference clock for the on chip FPGA transceiver is provided by the Si5341 clock generator manufactured by Silicon Lab. Bit error rate (BER) measurements are used to evaluate the performance of the communication link between various

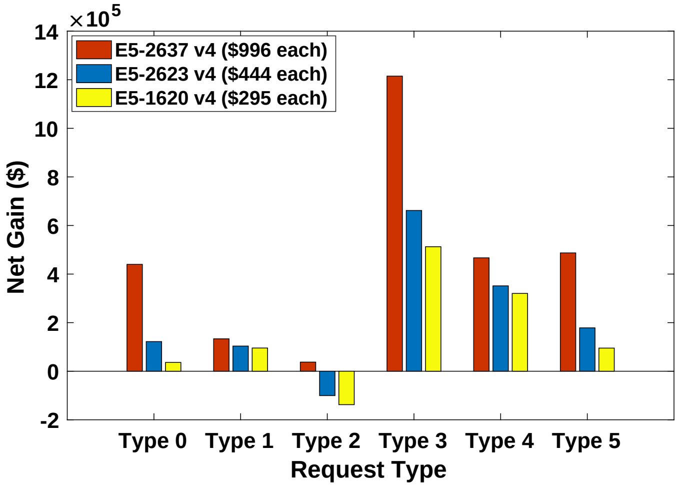
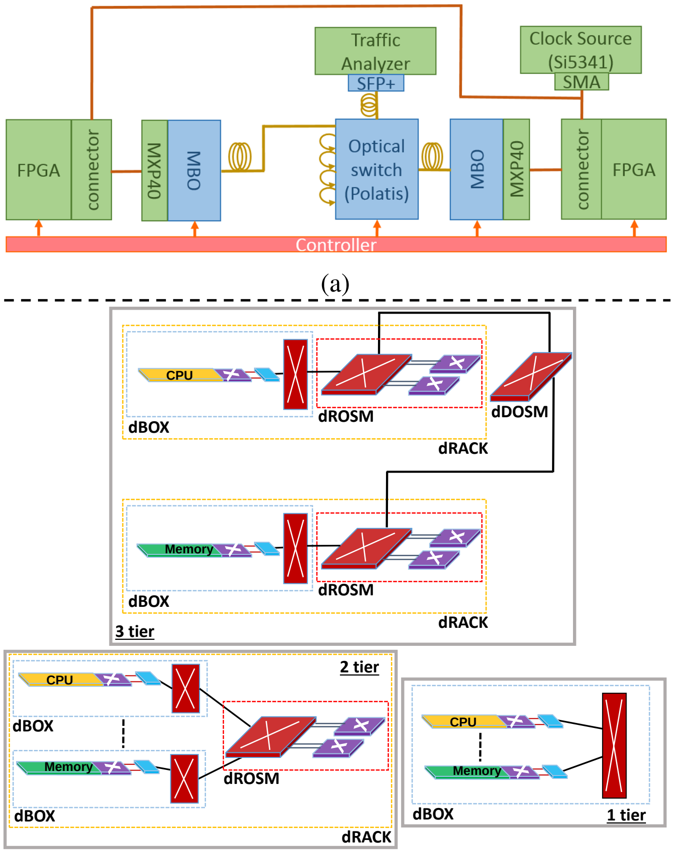


Fig. 12. Cost gain achieved from disaggregation



(a)

(b)

Fig. 13. a) Experimental setup of network testbed using FPGAs, SiP mid-board optics and miniaturized optical switches and b) visual representation of the multi tier disaggregated architecture being emulated by the experimental setup

elements, which are made possible by the MBOs and the optical switch. Thus, to obtain these BER reading in real time the Xilinx iBERT toolkit is used directly in conjunction with the FPGAs driving the MBOs. To overcome some of the limitations of the communication channel, the iBERT module is equipped with functionalities such a signal pre/post emphasis, CDR and receiver side channel equalization. Furthermore, to achieve optical circuit switching a miniaturized Polatis optical switch with 48 ports [43] is employed. To evaluate the number of switching hops, which this architecture can accommodate for, a single polatis switch is interconnected to emulate a multi tier topology. Moreover, a central network controller is responsible to manage the resource access configuration to each individual dBRICK and the optical switch.

The individual nodes or bricks employed in the this experimental test bed can operate at various link rates of up to 25 Gb/s. The measured BER trends for both the 16 Gb/s (matching the HMC maximum data rate) and 25 Gb/s scenarios with respect to the number of hops passed through the optical switch cross-connections are shown in Fig. 14. As it can be clearly seen, whilst operating at 25 Gb/s, the system was able to accommodate for three switching hops (2-tier network) through the optical switch before reaching the BER of 10^{-12}, whereas the 16 Gb/s system was able to reach up to 7 hops (4-tier network). By ensuring operation below the BER of 10^{-12} a FEC free and low latency operation can be guaranteed, this is crucial as the inclusion of FEC can contribute an additional 100 nsec latency to the system which is critical for compute to memory transactions in a disaggregated system. Furthermore, enabling the CDR at 25 Gb/s offers an additional 1 dB improvement in performance, thus resulting in the system achieving a total of 4 hops at the required BER threshold for the 25 Gb/s system. Considering that a 12-fibre MPO-to-LC fan-out harnesses were used between the SiP MBOs and optical switch with 1dB loss each it is expected to achieve 2 more hops (1dB/hop loss) with same performance when using same connector type. This corresponds to 3-tier (or 5-hop connection as per proposed architecture shown on Fig. 2 and even 5-tier (9-hop connection) networks respectively.

VIII. Acknowledgment

This work was partly supported by EU H2020 project dRedBox under the grant agreement no 687632. The authors will also like to acknowledge Huber+Suhner Polatis and Luxtera for the support and valuable discussions.

IX. Conclusion

This article reported and defined a set of fundamental network requirements which are key for the realization and evaluation of resource disaggregation. It also surveyed the latest work on optical interconnect technologies and highlighted the need for a) FEC-free operation that minimizes the latency, b) highest bandwidth density, c) Energy efficiency d) Low cost per bit and e) single mode operation to allow for the utilization of key optical switching technologies which can pave the way for scale-out architectures. A function and topology reconfigurable network architecture for resource disaggregation on the Rack-Scale data center was also proposed. This architecture is capable of supporting the diverse network needs i.e. CPU-to-memory, CPU-to-Storage, CPU-to-end user, etc.. At its core it supports a pure OCS network while offering hardware programmable functions at the end-points. Furthermore, these end-points do not act only as

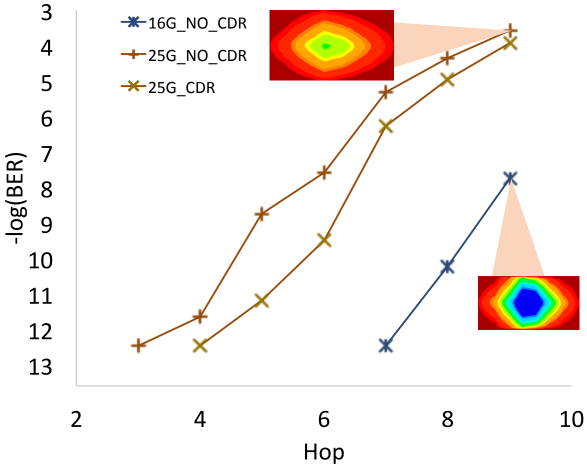


Fig. 14. BER results

compute elements, since by using Multi-Processor System on Chip (MPSoC) and methods developed they can support both compute and network functionalities. These network functionalities can include operations such as on-chip hybrid circuit and packet switching, be protocol programmable at each port. This eliminates the use of current overlay networks, one per each protocol (i.e. Infiniband, Ethernet, FibreChannel) support. It also eliminates the need the network interface cards since both compute and network functions are performed on the same chip. This also enhances the system performance in terms of latency, cost, footprint and power consumption.

Extensive simulation studies were also presented in this work which reported on the importance of resource locality between compute dBricks and memory dBricks, these studies further benchmarked disaggregated architectures against traditional data centre architectures. Four algorithms were developed and simulated, the results obtained highlight the importance of network aware algorithms in terms of bandwidth and latency to a) maximize IT resource utilization, b) minimize round-trip latency, and c) minimize overall cost of network. The best algorithm (NALB) together with the best placement of CPU and memory elements (S3 - heterogeneous dRacks with heterogeneous dBoxes) achieve round trip network latency overhead of only 15 ns, 95 ns and 315ns for 63%, 22% and 15% of CPU-to-memory connections across all established VMs. The same algorithm also achieve maximum CPU utilization. It was also shown that the additional cost of network required to serve compute to memory transactions, which doesn't exist in traditional data centers can impact the overall cost savings from having higher CPU and memory utilization. Thus, it is critical to develop technologies, architectures and algorithms which can further reduce the system costs while achieving substantial financial gains. Finally, an experiment was conducted with the use of FPGAs, 200 Gb/s Silicon Photonic mid-board optics and miniaturized optical cross-connects. It demonstrated the ability to deliver 10^{-12} (FEC free) over three or even five tier network at 25 Gb/s/channel and 16 Gb/s/channel respectively.

References

[1] S. Han, N. Egi, A. Panda, S. Ratnasamy, G. Shi, and S. Shenker, "Network support for resource disaggregation in next-generation datacenters," in *Proceedings of the Twelfth ACM Workshop on Hot Topics in Networks*, ser. HotNets-XII. New York, NY, USA: ACM, 2013, pp. 10:1–10:7. [Online]. Available: http://doi.acm.org/10.1145/2535771.2535778

[2] L. A. Barroso and U. Holzle, "The case for energy-proportional computing," *Computer*, vol. 40, no. 12, pp. 33–37, Dec 2007.

[3] Q. Chen, V. Mishra, and G. Zervas, "Reconfigurable computing for network function virtualization: A protocol independent switch," in *2016 International Conference on ReConFigurable Computing and FPGAs (ReConFig)*, Nov 2016, pp. 1–6.

[4] A. Peters and G. Zervas, "Network synthesis of a topology reconfigurable disaggregated rack scale datacentre for multi-tenancy," in *2017 Optical Fiber Communications Conference and Exhibition (OFC)*, March 2017, pp. 1–3.

[5] A. Klimovic, C. Kozyrakis, E. Thereska, B. John, and S. Kumar, "Flash storage disaggregation," in *Proceedings of the Eleventh European Conference on Computer Systems*, ser. EuroSys '16. New York, NY, USA: ACM, 2016, pp. 29:1–29:15. [Online]. Available: http://doi.acm.org/10.1145/2901318.2901337

[6] P. X. Gao, A. Narayan, S. Karandikar, J. Carreira, S. Han, R. Agarwal, S. Ratnasamy, and S. Shenker, "Network requirements for resource disaggregation," in *12th USENIX Symposium on Operating Systems Design and Implementation (OSDI 16)*. GA: USENIX Association, 2016, pp. 249–264. [Online]. Available: https://www.usenix.org/conference/osdi16/technical-sessions/presentation/gao

[7] B. Stiller, T. Bocek, F. Hecht, G. Machado, P. Racz, and M. Waldburger, "Roadmap report on photonics for disaggregated data centers workshop," University of Zurich, Department of Informatics, Tech. Rep., 01 2010.

[8] P. S. Rao and G. Porter, "Is memory disaggregation feasible?: A case study with spark sql," in *Proceedings of the 2016 Symposium on Architectures for Networking and Communications Systems*, ser. ANCS '16. New York, NY, USA: ACM, 2016, pp. 75–80. [Online]. Available: http://doi.acm.org/10.1145/2881025.2881030

[9] H. Meyer, J. C. Sancho, J. V. Quiroga, F. Zyulkyarov, D. Roca, and M. Nemirovsky, "Disaggregated computing. an evaluation of current trends for datacentres," *Procedia Computer Science*, vol. 108, pp. 685 – 694, 2017, international Conference on Computational Science, ICCS 2017, 12-14 June 2017, Zurich, Switzerland. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1877050917306968

[10] B. Abali, R. J. Eickemeyer, H. Franke, C. Li, and M. Taubenblatt, "Disaggregated and optically interconnected memory: when will it be cost effective?" *CoRR*, vol. abs/1503.01416, 2015. [Online]. Available: http://arxiv.org/abs/1503.01416

[11] Y. Yan, G. M. Saridis, Y. Shu, B. R. Rofoee, S. Yan, M. Arslan, T. Bradley, N. V. Wheeler, N. H. L. Wong, F. Poletti, M. N. Petrovich, D. J. Richardson, S. Poole, G. Zervas, and D. Simeonidou, "All-optical programmable disaggregated data centre network realized by fpga-based switch and interface card," *Journal of Lightwave Technology*, vol. 34, no. 8, pp. 1925–1932, April 2016.

[12] A. Pages, R. Serrano, J. Perell, and S. Spadaro, "On the benefits of resource disaggregation for virtual data centre provisioning in optical data centres," *Comput. Commun.*, vol. 107, no. C, pp. 60–74, Jul. 2017. [Online]. Available: https://doi.org/10.1016/j.comcom.2017.03.009

[13] http://www.eejournal.com/article/20170102-hbm-hmc/, accessed: 2017-07-18.

[14] http://www.itrs2.net/.

[15] C. Kachris and I. Tomkos, "A survey on optical interconnects for data centers," *IEEE Communications Surveys Tutorials*, vol. 14, no. 4, pp. 1021–1036, Fourth 2012.

[16] P. Dong, J. Lee, Y. K. Chen, L. L. Buhl, S. Chandrasekhar, J. H. Sinsky, and K. Kim, "Four-channel 100-gb/s per channel discrete multitone modulation using silicon photonic integrated circuits," *Journal of Lightwave Technology*, vol. 34, no. 1, pp. 79–84, Jan 2016.

[17] "Luxtera: Products," http://www.luxtera.com/luxtera/products , accessed: 2017-07-18.

[18] L. Chen, C. R. Doerr, P. Dong, and Y. kai Chen, "Monolithic silicon chip with 10 modulator channels at 25 gbps and 100-ghz spacing," *Opt. Express*, vol. 19, no. 26, pp. B946–B951, Dec 2011. [Online]. Available: http://www.opticsexpress.org/abstract.cfm?URI=oe-19-26-B946

[19] P. Dong, J. Lee, K. Kim, Y.-K. Chen, and C. Gui, "Ten-channel discrete multi-tone modulation using silicon microring modulator array," in *Optical Fiber Communication Conference*. Optical Society of America, 2016, p. W4J.4. [Online]. Available: http://www.osapublishing.org/abstract.cfm?URI=OFC-2016-W4J.4

[20] T. Aoki, T. Akiyama, A. Sugama, A. Hayakawa, H. Muranaka, T. Simoyama, S. Tanaka, M. Nishizawa, N. Hatori, Y. Sobu, Y. Chen, T. Mori, S. Sekiguchi, S. hwan Jeong, Y. Tanaka, and K. Morito, "Low crosstalk simultaneous 12 ch x 25 gb/s operation of high-density silicon photonics multichannel receiver," in *Optical Fiber Communication Conference*. Optical Society of America, 2017, p. W1A.2. [Online]. Available: http://www.osapublishing.org/abstract.cfm?URI=OFC-2017-W1A.2

[21] C. Li, T. Li, G. Guelbenzu, B. Smalbrugge, R. Stabile, and O. Raz, "Chip scale 12-channel 10 gb/s optical transmitter and receiver subassemblies based on wet etched silicon interposer," *Journal of Lightwave Technology*, vol. 35, no. 15, pp. 3229–3236, Aug 2017.

[22] K. Schmidtke, F. Flens, A. Worrall, R. Pitwon, F. Betschon, T. Lamprecht, and R. Krähenbühl, "960 gb/s optical backplane ecosystem using embedded polymer waveguides and demonstration in a 12g sas storage array (june 2013)," *J. Lightwave Technol.*, vol. 31, no. 24, pp. 3970–3975, Dec 2013. [Online]. Available: http://jlt.osa.org/abstract.cfm?URI=jlt-31-24-3970

[23] "firefly: Application design guide," http://suddendocs.samtec.com/ebrochures/firefly-brochure.pdf , accessed: 2017-07-18.

[24] "Finisar: Product guide," https://www.finisar.com/ , accessed: 2017-07-18.

[25] "Amphenol-icc.com, leap: On-board transceiver," http://www.amphenol-icc.com/, accessed: 2017-07-18.

[26] "Reflexphotonics, lightable: 40g and 120g," http://reflexphotonics.com , accessed: 2017-07-18.

[27] K. Nagashima, N. Nishimura, A. Izawa, T. Kise, and H. Nasu, "28-gb/s x 24-channel cdr-integrated vcsel-based transceiver module for high-density optical interconnects," in *2016 Optical Fiber Communications Conference and Exhibition (OFC)*, March 2016, pp. 1–3.

[28] B. G. Lee, D. M. Kuchta, F. E. Doany, C. L. Schow, C. Baks, R. John, P. Pepeljugoski, T. F. Taunay, B. Zhu, M. F. Yan, G. E. Oulundsen, D. S. Vaidya, W. Luo, and N. Li, "Multimode transceiver for interfacing to multicore graded-index fiber capable of carrying 120-gb/s over 100-m lengths," in *2010 23rd Annual Meeting of the IEEE Photonics Society*, Nov 2010, pp. 564–565.

[29] F. E. Doany, B. G. Lee, D. M. Kuchta, A. V. Rylyakov, C. Baks, C. Jahnes, F. Libsch, and C. L. Schow, "Terabit/sec vcsel-based 48-channel optical module based on holey cmos transceiver ic," *Journal of Lightwave Technology*, vol. 31, no. 4, pp. 672–680, Feb 2013.

[30] K. Hasharoni, S. Benjamin, A. Geron, S. Stepanov, G. Katz, I. Epstein, N. Margalit, D. Chairman, and M. Mesh, "A 1.3 tb/s parallel optics vcsel link," pp. 89 910C–89 910C–8, 2014. [Online]. Available: http://dx.doi.org/10.1117/12.2038073

[31] M. H. Eiselt, N. Eiselt, and A. Dochhan, "Direct detection solutions for 100g and beyond," in *2017 Optical Fiber Communications Conference and Exhibition (OFC)*, March 2017, pp. 1–3.

[32] F. Li, J. Yu, Z. Cao, J. Zhang, M. Chen, and X. Li, "Experimental demonstration of four-channel wdm 560 gbit/s 128qam-dmt using im/dd for 2-km optical interconnect," *Journal of Lightwave Technology*, vol. 35, no. 4, pp. 941–948, Feb 2017.

[33] H. Mardoyan, M. A. Mestre, J. M. Estarán, F. Jorge, F. Blache, P. Angelini, A. Konczykowska, M. Riet, V. Nodjiadjim, J.-Y. Dupuy, and S. Bigo, "84-, 100-, and 107-gbd pam-4 intensity-modulation direct-detection transceiver for datacenter interconnects," *J. Lightwave Technol.*, vol. 35, no. 6, pp. 1253–1259, Mar 2017. [Online]. Available: http://jlt.osa.org/abstract.cfm?URI=jlt-35-6-1253

[34] K. Zhong, W. Chen, Q. Sui, J. Man, A. P. T. Lau, C. Lu, and L. Zeng, "Experimental demonstration of 500gbit/s short reach transmission employing pam4 signal and direct detection with 25gbps device," in *2015 Optical Fiber Communications Conference and Exhibition (OFC)*, March 2015, pp. 1–3.

[35] R. Urata, H. Liu, X. Zhou, and A. Vahdat, "Datacenter interconnect and networking: From evolution to holistic revolution," in *2017 Optical Fiber Communications Conference and Exhibition (OFC)*, March 2017, pp. 1–3.

[36] S. Kanazawa, H. Yamazaki, Y. Nakanishi, T. Fujisawa, K. Takahata, Y. Ueda, W. Kobayashi, Y. Muramoto, H. Ishii, and H. Sanjoh, "Transmission of 214-gbit/s 4-pam signal using an ultra-broadband lumped-electrode eadfb laser module," in *Optical Fiber Communication Conference Postdeadline Papers*. Optical Society of America, 2016, p. Th5B.3. [Online]. Available: http://www.osapublishing.org/abstract.cfm?URI=OFC-2016-Th5B.3

[37] M. Morsy-Osman, M. Chagnon, and D. V. Plant, "Four-dimensional modulation and stokes direct detection of polarization division multiplexed intensities, inter polarization phase and inter polarization differential phase," *Journal of Lightwave Technology*, vol. 34, no. 7, pp. 1585–1592, April 2016.

[38] M. Chagnon and D. Plant, "504 and 462 gb/s direct detect transceiver for single carrier short-reach data center applications," in *Optical Fiber Communication Conference*. Optical Society of America, 2017, p. W3B.2. [Online]. Available: http://www.osapublishing.org/abstract.cfm?URI=OFC-2017-W3B.2

[39] F. Chang, "First demonstration of pam4 transmissions for record reach and high-capacity swdm links over mmf using 40g/100g pam4 ic chipset with real-time dsp," in *Optical Fiber Communication Conference*. Optical Society of America, 2017, p. Tu2B.2. [Online]. Available: http://www.osapublishing.org/abstract.cfm?URI=OFC-2017-Tu2B.2

[40] K. Zhong, W. Chen, Q. Sui, J. Man, A. P. T. Lau, C. Lu, and L. Zeng, "Experimental demonstration of 500gbit/s short reach transmission employing pam4 signal and direct detection with 25gbps device," in *2015 Optical Fiber Communications Conference and Exhibition (OFC)*, March 2015, pp. 1–3.

[41] D. M. Kuchta, A. V. Rylyakov, F. E. Doany, C. L. Schow, J. E. Proesel, C. W. Baks, P. Westbergh, J. S. Gustavsson, and A. Larsson, "A 71-gb/s nrz modulated 850-nm vcsel-based optical link," *IEEE Photonics Technology Letters*, vol. 27, no. 6, pp. 577–580, March 2015.

[42] K. Hasharoni, S. Benjamin, A. Geron, S. Stepanov, G. Katz, I. Epstein, N. Margalit, D. Chairman, and M. Mesh, "A 1.3 tb/s parallel optics vcsel link," pp. 89 910C–89 910C–8, 2014. [Online]. Available: http://dx.doi.org/10.1117/12.2038073

[43] N. Parsons, A. Hughes, and R. Jensen, "High radix all-optical switches for software-defined datacentre networks," in *ECOC 2016; 42nd European Conference on Optical Communication*, Sept 2016, pp. 1–3.

[44] J. Schmidt and U. Bruning, "openhmc - a configurable open-source hybrid memory cube controller," in *2015 International Conference on ReConFigurable Computing and FPGAs (ReConFig)*, Dec 2015, pp. 1–6.

[45] Q. Chen, V. Mishra, N. Parsons, and G. S. Zervas, "Hardware programmable network function service chain on optical rack-scale data centers," in *Optical Fiber Communication Conference*. Optical Society of America, 2017, p. Th2A.35. [Online]. Available: http://www.osapublishing.org/abstract.cfm?URI=OFC-2017-Th2A.35

[46] B. Abali, R. J. Eickemeyer, H. Franke, C. Li, and M. Taubenblatt, "Disaggregated and optically interconnected memory: when will it be cost effective?" *CoRR*, vol. abs/1503.01416, 2015. [Online]. Available: http://arxiv.org/abs/1503.01416