Optimal VDC Service Provisioning in Optically Interconnected Disaggregated Data Centers

Albert Pagès, Jordi Perelló, Fernando Agraz and Salvatore Spadaro, *Member, IEEE*

***Abstract*—Virtual Data Center (VDC) is a key service in modern Data Center (DC) infrastructures. However, the rigid architecture of traditional servers inside DCs may lead to blocking situations when deploying VDC instances. To overcome this problem, the disaggregated DC paradigm is introduced. In this paper, we present an Integer Linear Programming (ILP) formulation to optimally allocate VDC requests on top of an optically-interconnected disaggregated DC infrastructure, aiming to quantify the benefits that such architecture can bring when compared to traditional server-centric DCs. Moreover, a lightweight Simulated Annealing (SA)-based heuristic is provided for scenarios where the ILP scalability is challenged. The obtained numerical results reveal the substantial benefits yielded by the resource disaggregation paradigm.**

***Index Terms*—Data center networks, resource disaggregation, linear programming, optimization.**

I. Introduction

In traditional DCs, thousands of servers are utilized to execute several applications on top. However, such server-centric DC (SC-DC) architectures present some drawbacks in terms of efficient server utilization, which can lead to resource underutilization. For instance, Google has recently published data regarding the utilization of their DC infrastructures, which show high disparity of storage/memory to CPU usage for the tasks [1]. In such a case, it may happen that a task employs almost the totality of one server resource (e.g., memory), while the utilization of the other resources (e.g., CPU and storage) remains fairly low. Thus, in order to fully exploit the entire server resources, it becomes imperative to propose new DC architectures.

To this end, the disaggregated DC paradigm (hereafter referred as DA-DC) has been recently introduced [2]–[4]. In a DA-DC, the computing resources (CPU cores, storage and memory) are no longer hosted in server units, but spread over several standalone hardware blades interconnected through an intra-DC network (DCN) fabric. In this way, computing resources can be tightly assigned to tasks according to their needs. The different blades can be grouped into racks hosting all types of resources or confined in different mono-hardware racks, where a single type of resource is hosted [4]. In both situations, high throughput and low latencies are required for the communication among the different hardware modules. For these purposes, the utilization of optical technologies is envisioned inside DCs [4]. That is, an optical DCN is equipped for the intra- and inter-rack communication of the different resource blades and the applications running on top. DA-DCs promise a dramatically higher utilization of the computing resources. This feature is especially beneficial when providing complex infrastructure services, such as VDCs [5], [6]. To the best of our knowledge, no studies exist to date quantifying the benefits of optically-interconnected DA-DCs for efficiently allocating VDC services. This paper aims to provide insight into this question. To this goal, next section elaborates on the scenario under consideration.

The authors are with the Advanced Broadband Communications Center (CCABA), Universitat Politècnica de Catalunya (UPC), Barcelona, Spain (e-mail: {albertpages, agraz, spadaro}@tsc.upc.edu, perello@ac.upc.edu).

II. VDC allocation in disaggregated DCs

VDC is conceived as an Infrastructure as a Service (IaaS), allowing DC operators to lease part of their infrastructure to multiple tenants. These tenants may ask for a virtual infrastructure (VDC) composed of computing resources (Virtual Machines, VMs) interconnected through virtual links with a certain capacity. The requested VDC can afterwards be used as a platform to deploy applications on top by the tenant, offered as services to end users. Then, the DC operator must optimally map (i.e., allocate) the VDCs onto physical DC resources. Such a process involves the mapping of the VMs and virtual links onto computing and network resources, respectively. Their joint mapping generally allows for an enhanced utilization of the underlying physical infrastructure, increasing the number of supported VDCs [6]. In this regard, a DA-DC architecture should further increase the number of allocated VDCs, since the different VMs may exploit its potentially higher computing resource utilization. Before proceeding, let us note that the DA-DC paradigm is foreseen as a very long term solution to nowadays DC architectures, due to the use of complex optical technologies. Nevertheless, its potential benefits make it a very promising architecture worth to be studied.

In this context, we analyze the mapping of VDCs onto an optically interconnected Dense Wavelength Division Multiplexing (DWDM)-based transparent DA-DC infrastructure, based in the one proposed in [4]. Figure 1 depicts the assumed architecture. We assume that all hardware blades of a rack are interconnected through dedicated fiber links between blades, so as to mitigate incurred latencies and bandwidth limitations when communicating hardware modules within the same rack. As for the inter-rack communication, blades are connected to the DCN through the corresponding Top of the Rack switch (ToR), which is considered fully optical, that is, it optically switches the signals coming from the resource blades (no electrical processing is done). This is possible thanks to the optical interfaces at each resource blade for inter-rack communications [4]. In turn, ToRs are interconnected through a

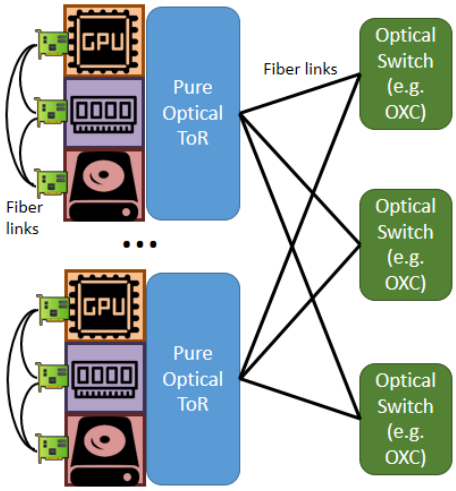

Fig. 1. Assumed DA-DC architecture.

set of optical switches, assumed to be Colorless, Directionless and Contentionless (CDC) Optical Cross Connects (OXCs). We then focus on the optimization problem where a known beforehand set of VDCs has to be mapped onto the physical resources of a DC, which are limited, with the objective of minimizing the number of blocked VDCs. We consider that all racks host all types of computing resources while we assume that a single VM is always mapped onto a single rack, since mapping it over different racks could lead to unacceptably high latencies between the different hardware modules (we assume no latency or bandwidth limitations for the intra-rack communication thanks to the direct by-pass between hardware blades within a rack). The case where a VM can be mapped onto different racks is left for future studies. Moreover, we also consider that VMs belonging to the same VDC are mapped over different racks, so as to provide some degree of protection against rack failures. Note, however, that VMs belonging to different VDCs can still be mapped to the same physical rack. As for the virtual links, they must respect the wavelength continuity constraint (a transparent DCN is considered) and different VDCs must not share optical resources between them (for isolation purposes).

Once a VDC is allocated no further physical infrastructure reconfigurations follow. Instead, the tenant can start using it. For example, the tenant could install applications on the VMs composing the VDC, which could exchange layer-2/3 flows sharing the allocated virtual link capacities. This work does not address the efficient routing of such application flows between VMs of a VDC, but only the efficient mapping of the VDC required capacity onto the underlying DCN and IT resources.

The following section presents an ILP formulation to tackle the addressed optimization problem. Additionally, an SA-based heuristic is also presented for scenarios where the execution times of the ILP model become unrealistically long. For benchmark purposes, we also developed a variation of the presented approaches focusing on the SC-DC scenario. In this case, a VM has to be mapped over a single server of a rack. The rest of the constraints remain the same as in the DA-DC case. Due to the lack of space, we will focus on the optimization approaches for the targeted DA-DC scenario, while only sketching the modifications required to contemplate the benchmark SC-DC scenario.

III. Optimization approaches

Let $G_n = (N_f, E_f)$ be the graph of the optical DCN, with N_f the set of nodes (racks/ToRs and optical switches) and E_f the set of physical links. Each link is equipped with W wavelength channels. Let P be the set of paths in the DCN, with P_{e_f} the set of paths that traverse link e_f. Additionally, let $R \subset N_f$ be the set of racks in the DC infrastructure, with each rack r holding an aggregated capacity in terms of CPU cores, storage and memory equal to C_r, H_r and M_r, respectively. Moreover, let D be the set of VDCs. Each VDC is characterized by the virtual graph $G_d = (N_v^d, E_v^d)$, with N_v^d the set of VMs and E_v^d the set of virtual links. Each VM requests a set of computing resources (cores, storage, memory) denoted by $(C_{n_v}, H_{n_v}, M_{n_v})$, while each virtual link requests a capacity in wavelengths equal to B_{e_v}. Finally, we denote as $a(\cdot)$ and $b(\cdot)$ the source and destination endpoints of a virtual link e_v or a physical path p. With these definitions, we proceed introducing the proposed optimization approaches.

A. ILP formulation

This sub-section presents the ILP formulation for the DA-DC case, named ILP-Virtual Data Center Planning (ILP-VDCP). The ILP model variables are:

$x_{d,e_v,p,w}$: binary; 1 if virtual link e_v of VDC d is mapped onto physical path p and wavelength w, 0 otherwise.

$y_{d,n_v,r}$: binary; 1 if VM n_v of VDC d is mapped onto rack r, 0 otherwise.

z_d: binary; 1 if VDC d is blocked, 0 otherwise.

The ILP formulation is detailed below:

$$\min \sum_{d \in D} z_d \text{ s.t.} \tag{1}$$

$$\sum_{r \in R} y_{d,n_v,r} \leq 1, \forall d \in D, n_v \in N_v^d \tag{2}$$

$$\sum_{n_v \in N_v^d} y_{d,n_v,r} \leq 1, \forall d \in D, r \in R \tag{3}$$

$$\sum_{d \in D} \sum_{n_v \in N_v^d} \begin{Bmatrix} C_{n_v} \\ H_{n_v} \\ M_{n_v} \end{Bmatrix} \cdot y_{d,n_v,r} \leq \begin{Bmatrix} C_r \\ H_r \\ M_r \end{Bmatrix}, \forall r \in R \tag{4}$$

$$\sum_{d \in D} \sum_{e_v \in E_v^d} \sum_{p \in P_{e_f}} x_{d,e_v,p,w} \leq 1, \forall e_f \in E_f, w \in W \tag{5}$$

$$\sum_{w \in W} x_{d,e_v,p,w} \leq B_{e_v} \cdot \begin{Bmatrix} y_{d,a(e_v),a(p)} \\ y_{d,b(e_v),b(p)} \end{Bmatrix}, \begin{matrix} \forall d \in D, e_v \\ \in E_v^d, p \in P \end{matrix} \tag{6}$$

$$|N_v^d| \cdot z_d + \sum_{n_v \in N_v^d} \sum_{r \in R} y_{d,n_v,r} = |N_v^d|, \forall d \in D \tag{7}$$

$$|E_v^d| \cdot z_d + \sum_{e_v \in E_v^d} \frac{1}{B_{e_v}} \sum_{p \in P} \sum_{w \in W} x_{d,e_v,p,w} = |E_v^d|, \forall d \in D \tag{8}$$

Objective function (1) minimizes the number of blocked VDCs. Constraint (2) ensures that a VM is mapped to only one physical rack, while constraint (3) ensures that the VMs of a VDC are mapped onto different racks. Constraint (4) guarantees that the computing resource capacity of a physical rack is not exceeded. Constraint (5) prevents that two light-paths employ the same wavelength over the same physical link. Constraints (6) restrict the virtual link mapping over physical paths interconnecting the nodes onto which its remote endpoints have been mapped. Finally, constraints (7) and (8) discriminate if a VDC is either blocked or not. If not, all of its resources (VMs and links) must be successfully mapped. Regarding the ILP model for the SC-DC case, instead of defining variables $y_{d,n_v,r}$, we define variables $y_{d,n_v,r,s}$ so as to account for the particular server inside a rack onto which a VM is mapped. All the remaining constraints stay the same but they account for the server dimension and the capacity of a single server, instead of the aggregated capacity of the whole rack.

B. Heuristic approach

As will be shown in section IV, the execution times of ILP-VDCP substantially grow with the size of the problem instance to solve. Hence, we also propose a SA-based heuristic, called SA-VDCP that achieves close to optimal results in a shorter time. Algorithm 1 depicts its pseudo-code. Basically, SA-VDCP is structured in two phases. In the first phase, the initial solution is constructed. To this end, it tries to successfully map the VDCs in the demand set iteratively. It firstly tries to map the VMs of a VDC onto the physical racks. For this, SA-VDCP computes the normalized difference of the available resources in the rack and the requested resources of the VM using expression (9), conditioned to the resource availability, where C_r^a, H_r^a and M_r^a are the current available resources in the rack. Then, it maps the VM to the rack minimizing this difference.

$$\Delta_r^{n_v} = \frac{1}{3} \cdot \left(\frac{C_r^a - C_{n_v}}{C_r} + \frac{H_r^a - H_{n_v}}{H_r} + \frac{M_r^a - M_{n_v}}{M_r} \right) \quad (9)$$

If all the VMs are successfully mapped, it proceeds with the mapping of the virtual links. A K-Shortest Path (SP) routing strategy is employed to this end, assigning the wavelengths in a First-Fit (FF) fashion. The second phase is the actual solution improvement and cooling procedure. Being any VDC not mapped, a neighboring solution is produced. For this, an accepted VDC is randomly extracted from the solution. Next, it randomly sorts the unaccepted VDCs and tries to map them according to the procedure of phase 1. The new solution is accepted according to the current temperature and the differences with the best solution found so far. This process repeats until either a final temperature is reached (t_f), all VDCs are served or a number of iterations without improvement is met ($maxIt$). At the end, the best solution is returned. For the SC-DC case, the VM mapping is decided according to the rack that hosts the server minimizing expression (9), employing the resources per server instead of those of the whole rack. That is, the VM is placed in the server that provides a tighter fit.

Algorithm 1: SA-VDCP pseudo-code.

```
Inputs: D, G_n, t_i, t_f, α, maxIt; Outputs: Sol
Sol ← ∅
Phase 1: Initial solution building
for d = 1 to |D| do
    for n_v = 1 to N_v^d do
        Map n_v to the rack r that minimizes Δ_r^{n_v}
    if All n_v ∈ N_v^d are mapped then
        for e_v = 1 to E_v^d do
            Interconnect endpoints of e_v with a K-SP-FF strategy
    if d is fully mapped then
        Sol ← Sol∪ mapping of d
Obj(Sol) ← blocked demands
Phase 2: Solution improvement
if Obj(Sol) ≠ 0 then
    t ← t_i, It ← 0
    while t > t_f and Obj(Sol) ≠ 0 and It < maxIt do
        auxSol ← generateNeighbour(Sol)
        if Obj(auxSol) < Obj(Sol) then
            Sol ← auxSol
            It = 0
        else
            It = It + 1
            Δ = Obj(auxSol) − Obj(Sol)
            R = random(0, 1)
            if R < e^{−Δ/t} then
                Sol ← auxSol
        t = α · t
Return Sol
```

IV. Results and Discussion

In this section, we evaluate the performance of a DA-DC when compared to a traditional SC-DC. For this, in the SC-DC case, we consider that each rack is equipped with a limited number of servers, while in the DA-DC case each rack holds the total aggregated computing resources of a rack in the SC-DC case. We then consider a DC scenario consisting of 8 racks, where all racks are interconnected through the corresponding ToR to a central OXC in a tree-like structure. For the demand set, VDCs are generated with 2-5 VMs interconnected randomly with virtual links, preventing the generation of non-connected VDCs. For simplicity, we assume that all virtual links request a wavelength each, since our study focuses on the evaluation of the higher computing resource allocation flexibility that a DA-DC can offer. Focusing on the VMs, we considered four different configurations in terms of (CPU cores, storage, memory): T_1 (8,600,48), T_2 (16,600,48), T_3 (8,1400,48) and T_4 (8,600,112). These configurations are inspired on the Amazon Elastic Compute Cloud service [7]. In this regard, we assumed two different scenarios: 1) 60% of the VMs are T_1, while the remaining 40% is equally distributed between T_2, T_3 and T_4; and 2) VMs are equally distributed between T_2, T_3 and T_4. The capacity of a single server is set to (24,2000,128).

We firstly validate the performance of SA-VDCP against ILP-VDCP. For this, we focus on scenario 1, considering 10 servers per rack and 40 wavelengths per physical link. The obtained results are depicted in Table I, in which every data point has been obtained by averaging 20 random instances. All executions in this section have been run in PCs with i7-4770 at 3.4GHz CPUs and 16GB of memory, employing CPLEX v12.5 optimization software [8]. Regarding SA-VDCP, we employ $\alpha = 0.999$, $maxIt = 10^4$ and t_i, t_f have been set according to the procedures explained in [9] and [10], respectively. Looking at the obtained results, it can be seen that SA-VDCP achieves

TABLE I
SA-VDCP VALIDATION AGAINST ILP-VDCP

Scenario	$\|D\|$	ILP-VDCP Time (s.)	ILP-VDCP Obj.	SA-VDCP Time (s.)	SA-VDCP Obj.	SA-VDCP % Error
Server -centric	20	436.2	20	0.21	20	0
	40	$>$ 12h.	38.3	2.94	36.8	3.9
	60	$>$ 12h.	47	6.21	44.9	4.4
Dis- aggregated	20	185.72	20	0.015	20	0
	40	4287.86	40	0.019	40	0
	60	$>$ 12h.	59.7	0.137	59.25	0.75

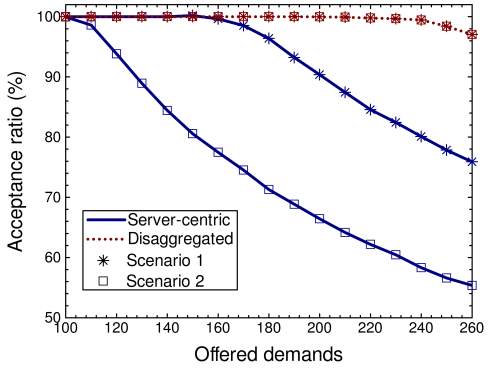

Fig. 2. VDC acceptance ratio.

close to optimal results in much lower times than ILP-VDCP, confirming both its accuracy and speed.

Next, we analyzed the VDC acceptance ratio as a function of the size of D. For this, we consider 48 servers per rack and all physical links having a capacity of 160 wavelength channels per link, assuming a total C-band spectrum of 4THz and a channel spacing of 25GHz. The obtained results for both scenario 1 and 2 are depicted in Figure 2. All the results presented hereafter have been averaged over 100 random instances per data point, employing the proposed SA-VDCP heuristic. As shown, in a DA-DC the VDC acceptance ratio is almost 100% in the considered scenarios, only decreasing slightly for larger sizes of D. On the other hand, the VDC acceptance ratio in the SC-DC case decreases substantially with larger demand sets, especially with the presence of highly specialized VMs (scenario 2). This is due to the fact that, when allocating a VM, a particular computing resource of a server may be almost fully utilized while the utilization of the rest remains fairly low. In such a case, it may not be possible to allocate the VM in the server, thus requiring another one. Such a phenomena can lead to a substantial underutilization of the server and rack resources, finally resulting into significant VDC blocking. Conversely, in a DA-DC, as the computing resources are not confined to a single server, but may be utilized from the available pool of resources to tightly fit the VM needs, it is possible to achieve a higher utilization of the computing resources, substantially increasing the acceptance of VDCs (up to around 50%). Only in scenarios where the network starts being the bottleneck, a DA-DC may face some blocking situation.

To analyze this, we also extracted the average rack and physical link utilization for scenario 1 with both DC architectures (Figure 3). Looking at the obtained results, it can be seen

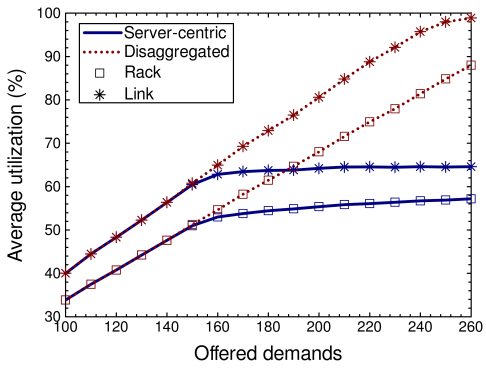

Fig. 3. Average rack and link resource utilization (scenario 1).

that a DA-DC increases substantially the resource utilization of the rack (by around 60%) in comparison with a SC-DC, where the main bottleneck is the constrained capacity of a single server, thus the reason behind the observed low rack utilization. Additionally, it can be appreciated that only in scenarios where the network utilization is close to 100% the DA-DC starts to face blocking situations. Such results confirm that a DA-DC architecture may overcome the limitations of SC-DCs in terms of computing resource utilization.

V. CONCLUSIONS

VDC provisioning is a key service to enable DCs towards future cloud infrastructures. In this regard, optically interconnected DA-DCs are a very promising candidate for efficient allocation of VDCs. We have shown that a DA-DC allows for a substantially higher acceptance ratio of VDCs (up to 50% more) thanks to its finer modularity in terms of hardware configuration when mapping VMs. As a consequence, a more efficient utilization of the computing resources can be achieved (around 60%).

ACKNOWLEDGMENT

This work is supported by the Spanish Government through project SUNSET (TEC2014-59583-C2-1-R) with FEDER contribution.

REFERENCES

[1] C. Reiss et al., "Google cluster-usage traces: format + schema", *Google Technical Report*, 2012, http://code.google.com/p/googleclusterdata/wiki/TraceVersion2
[2] S. Han et al., "Network Support for Resource Disaggregation in Next-Generation Datacenters", *Proc. of HotNets 2013*, College Park (USA), Nov. 2013.
[3] J. Weiss et al., "Optical interconnects for disaggregated resources in future datacenters", *Proc. of ECOC 2014*, Cannes (France), Sept. 2014.
[4] G. Saridis et al., "EVROS: All-Optical Programmable Disaggregated Data Centre Interconnect Utilizing Hollow-Core Bandgap Fibre", *Proc. of ECOC 2015*, Valencia (Spain), Sept. 2015.
[5] Interoute VDC service, https://cloudstore.interoute.com/
[6] A. Pagès et al., "Optimal Virtual Slice Composition Toward Multi-Tenancy Over Hybrid OCS/OPS Data Center Networks", *IEEE/OSA J. Opt. Commun. Netw.*, vol. 7, num. 10, pp. 974-986, 2015.
[7] Amazon EC2 service, https://aws.amazon.com/ec2/
[8] IBM CPLEX Optimizer, http://www-01.ibm.com/software/commerce/optimization/cplex-optimizer/
[9] W. Ben-Ameur, "Computing the initial temperature of simulated annealing", *Computational Optimization and Applications*, vol. 29, no. 3, pp. 369-385, 2004.
[10] M. Lundy et al., "Convergence of an Annealing Algorithm", *Mathematical Programming*, vol. 34, pp. 111-124, 1986.