Virtualization and Cloud Computing Project

On

ESP-VDCE: Energy, SLA, and Price-driven Virtual Data Center Embedding

Submitted by

212IS019. Rakshit Singh
212IS023. Rohit Panwar

Under the Guidance of

SOURAV KANTI ADDYA

Dept. of Information Technology,

NITK, Surathkal

Date of Submission: 04/04/2022

Department of Information Technology
National Institute of Technology Karnataka, Surathkal.

Abstract

We introduce a multi-objective Virtual Data Center Embedding (VDCE) scheme for multi-domain cloud computing configurations in this paper. The suggested technique is called ESP-driven VDCE and its major focus is on energy reduction, SLA assurance, and lower energy prices. We formulate the suggested scheme as an optimization problem in the first stages of this research. The defined issue is, however, redesigned and separated into three sub-problems (SP), namely data centre identification, virtual machine mapping, and virtual link embedding, due to its intractability. The search space can be greatly limited by using the output of one SP as an input to the next SP. Finally, the suggested VDCE technique is thoroughly tested against competing algorithms.The findings show that the suggested ESP-driven VDCE works.technique results in about 7.6 percent more energy-aware embeddingsSLA levels are 11.3 percent higher and about 23 percent lower Expenses for energy.

Keywords— Cloud Computing, Optimization, Heuristics, Virtual Data Center Embedding Problem, and Virtual Machine

Contents

1 Topic

primary focus of the proposed scheme is on Energy minimization, SLA assurance, and reduced energy Prices.

2 Introduction

With the increasing popularity of cloud computing, data centres have grown in importance in recent years. As a result, many technological behemoths such as Google, Facebook, Amazon, and Yahoo have widely deployed data centre networks (DCNs) to support large-scale applications such as high-performance computing, large-scale simulations, web services, and so on [1]. Furthermore, the rapidly increasing need for information systems has increased the value and expansion of data centres [2]. Large-scale data centre implementation, on the other hand, typically comes with a variety of issues related to power consumption, network performance, manageability, and security.Thus, optimal cloud infrastructure use is necessary to pave the way for cost-effective and long-term data centre architectures [3] [4].

Virtualization has received increased attention in order to achieve this goal and assure compliance with QoS criteria [5]. It supports a wide range of network services and topologies, allowing various distinct users to coexist on a common substrate network, resulting in a modular and fully isolated Virtual Network [6]. Most cloud providers provide virtual machine (VM) instances with various resource options to assign cloud resources to an application [7] [8].However, the shortage of bandwidth and longer delays make optimal allocation of substrate resources to numerous virtual networks problematic, limiting the performance of installed services and applications [9] [10].

Virtual Data Centers (VDCs) have evolved as a solution for optimal resource use to overcome these flaws [11]. A virtual data centre (VDC) is a collection of VMs with allocated compute and storage resources that are linked to bandwidth resources through virtual connections. VDCs, in comparison to previous techniques, provide flexible infrastructure allocation, lower operating costs, full automation, support for geographically scattered data centres, and simple VM migration [12]. Despite its advantages, providing VDCs as a service is difficult owing to the resource allocation challenge, which necessitates VDC embedding (VDCE) in the substrate network [13]. Finding an efficient mapping of VMs and virtual linkages to physical components while adhering to service level agreements (SLAs) is part of embedding a VDC [14].

3 System Framework

A typical DC is defined by its compute and network nodes, their relative resource capabilities, and the connections that connect them. Figure 1 displays a DC's design, which includes computing nodes and several types of switches, including top-of-the-rack (TOR), aggregation, and core. The resources at the DC may be defined using an undirected graph, G_{DC}=(N, L, R_N, R_L), where N and L represent the physical network's nodes (physical servers and physical switches) and connections, respectively. Their computation and network resources are represented by R_N and R_L, respectively.

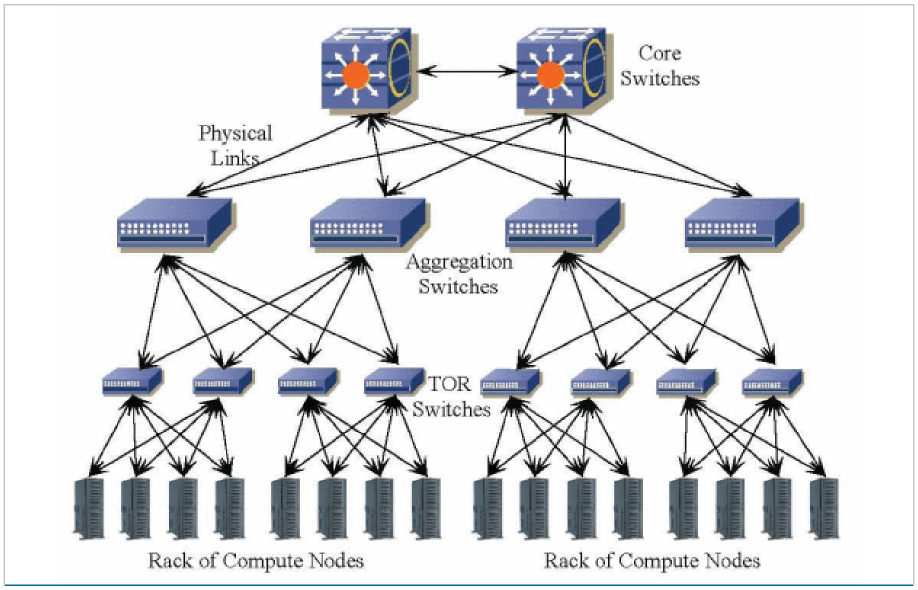

Figure 1: A typical data center setup.

Similarly, virtual data centres (VDCs) are basically DCs built on virtualization techniques, allowing them to virtually share resources such as VMs, virtual links (VLs), routers, and switches. A VDC's resources can alternatively be described using an undirected graph, i.e.,G^*_{VDC}=(N^*, $L^* R_N$, R^*_L). The variables N^* and L^* relate to the virtual nodes (referred to as Virtual Machines, VMs) and VLs associated with a VDC, respectively. Furthermore, the computational resources required by a VDC are represented by RN, whilst the network resources are represented by R^*_L. As a result, the VDCE issue may be defined as the process of embedding graph $G^*_{1/DC}$G1/7DC in a subset of G_{DC}. In other words, the GDC subset is simply the mirror image of G^*_{VDC} in G_{DC}.i.e,

$G^*_{VDC} \rightarrow G_{DC}$.

4 Problem Definition

Aside from the aforementioned mapping, an unique contribution of this study is to achieve minimum energy usage across diverse geographical DCs with little SLA violations and total power expenditures. In this context, the set of examined objectives and their related limitations are as follows. In addition, the many decision factors utilised in the formulation of the VDCE issue are given below.

4.1 Decision Variables

The following decision variables are commonly used in the cloud computing context to achieve the ESP-driven VDCE. The binary decision variable (Yj(n, m)) is used to represent the mapping of the nth belong to N^* VM of the j th VDC request on the mth(M) physical machine.

$$\mathbb{Y}_j(n,\ m) = \begin{cases} 1: \text{ Ir the } n^{\text{th}} \text{ VM or the } j^{\text{th}} \text{ VDC is mapped} \\ \text{on the } m^{\text{th}} \text{ physical machine} \\ 0: \text{ Otherwise} \end{cases}$$

$$\mathbb{Z}_j(q,p) = \begin{cases} 1: \text{ If the VL} q \in L^* \text{ of the } j^{\text{th}} \text{VDC is} \\ \text{ mapped on the } p^{\text{th}} (\in L) \text{ physical link} \\ 0: \text{ Otherwise} \end{cases}$$

Furthermore, Xj(q, p) is used to express the mapping of virtual linkages over physical links . This variable is binary in nature and is defined by the equation below.

4.2 Objective Functions

The major objectives of the proposed work are specified using the choice factors listed above.

4.2.1 Minimal Energy Consumption:

The primary goal of the proposed system is to consume as little energy as possible while encapsulating the VDC request in terms of VM and VL mapping. This is accomplished by employing the following objective function (SE(.)):

$$\mathrm{F}_E\left(V_j(n,m), Z_j(q,p)\right) = \sum_{m=1}^{|N|} \sum_{n=1}^{\left|N^*\right|} E_m^{Node} \times Y_j(n,m) + \sum_{p=1}^{|L|} \sum_{q=1}^{\left|L^*\right|} E_p^{Switch} \times Z_j(q,p), 4$$

where $\mathrm{En}_{mN}ode$ and $\mathrm{En}_{ps}witch$ Enswitch.p denote the energy consumption of the mth physical compute node and pth switch.

4.2.2 Maximal SLA Assurance:

This objective function (SS(.)) tries to maximize the chances of mapping the VMs and VLs to their respective nodes.

$$\mathrm{F}_S(Y_j(n,\ m),\ Z_j(q,p)) = \sum_{m=1}^{|N|} \sum_{n=1}^{|N^*|} Y_j(n,\ m) + \sum_{p=1}^{|L|} \sum_{q=1}^{|L^*|} Z_j(q,p),$$

4.2.3 Constraints:

The list of the constraints assumed in the formulated problem are discussed as under.

A.VM Embedding Constraints: These constraints are set in order to achieve VM embedding across the DCs architecture. The first limitation ensures that each VDC request for VM allocation is always assigned to a single physical server. The following equation expresses it technically.

$\sum_{m=1}^{N} Y_j(n,\ m) \leq 1_{\rm i} \cdot \forall j,\ n,$

Another important limitation to consider when mapping VMs to real machines is resource availability. As a result, the following restriction guarantees that the resources allocated to all VMs (R_n^*) mapped to a physical server never exceed the node's overall resource limit (R_m).

$\sum_{n=1}^{N^*} Y_j(n,\ m).R_n^* \leq R_{m7} \cdot \forall j,\ m,$

B.Virtual Link Embedding Constraints: The VL embedding across DCs is controlled by these restrictions. The first limitation, which is defined as follows, is connected to the allocation/mapping of a VL across various physical pathways.

$\sum_{p=1}^{L} Z_j(q,p) \geq 1_{\}} \cdot \forall q, j$

Furthermore, the VLs' mapping on the available physical links should correspond to the resource capacity, i.e., the link's bandwidth constraint. It is represented by the following equation.

$\sum_{q=1}^{L^*} X_{:i}(q,p).R_q^* \leq R_{p\}} \cdot \forall p, j,$

Here, (R_q^*)denotes the resources required by the qth virtual link for the mapping of a physical path.

C.Decision Variable Constraints: These constraints impose restrictions on the considered set of decision variables, i. e., Yj(n, m) and Zj(q, p). These are represented using the following equations.

$Y_j(n,\ m) \in \{0,1\}_7 \cdot \forall j,\ m,\ n,$

$Z_j(q,p) \in \{0,1\}_{\}} \cdot \forall j, p,\ q$

5 Proposed Vdc Embedding Approach

This section proposes a heuristic-based method for solving the above-described problem in real-time. The offered techniques are effective because they divide the overall problem into different sub-problems (SPs). These SPs, in turn, aid in gradually limiting the solution space and achieving a suboptimal solution in polynomial time. In order to achieve energy, cost, and power-aware VDCE in cloud computing installations, three SPs are examined.

5.1 SP1: Locating the optimal DC for VDCE

This SP aids in the selection of the best DC for embedding VDC requests across physical nodes and connections. This is accomplished by applying Algorithm 1 to the following multi-objective optimization problem (S1(Xij)). Because the examined optimization issue has a restricted number of variables (i.e., the total number of geographically scattered DCs), the defined problem may be solved in real-time using an ILP solver. In this study, we utilised MATLAB to answer the following F1 problems (Xij).

```
Result:
–• Embedding of VLs with due consideraion to energy
efficiency and SLA assurance
Input  : –•VDC Requests $j \in G^*_{VDC}$
         –•$\mathbb{X}_{ij}$ : DC $i$ for embedding VDC request $j$
         –•$\mathbb{PP}$ : List of physical paths in the DC
         –•$R_p$ : Resources available with $p^{\text{th}}$ physical path
         –•$\mathbb{VL}$ : List of virtual links (VLs) to be mapped
         –•$R^*_q$ : Resource demand of the $q^{\text{th}}$ virtual link
Output: –•$\mathbb{Z}_j(q,p)$; $\forall j,p,q$
while ($\mathbb{VL} \neq Empty$) do
    Set $q \leftarrow 1$
    $\mathbb{PP}_{vdc} \leftarrow$ Physical paths assigned to associated VDCs
    if $\mathbb{PP}_{vdc} \neq Empty$ then
        for $\forall p \in \mathbb{PP}_{vdc}$ do
            Compute link utilization level $\eta_p$
        end
        sort($\mathbb{PP}_{vdc}$, value=$\eta_p$, order='desc')
        for Physical path $\mathbb{PP}_p \in \mathbb{PP}_{vdc}$ do
            if $R^{avail}_p \geq R^*_q$ then
                Set $\mathbb{Z}_j(q,p) \leftarrow 1$
                Compute $R^{avail}_p \leftarrow R^{avail}_p - R^*_q$
                $\mathbb{PP} \leftarrow \mathbb{PP}/\mathbb{PP}_m$
                Return
            end
        end
    end
    $\mathbb{PP} \leftarrow \mathbb{PP}/\mathbb{PP}_{vdc}$
    for $\forall p \in \mathbb{PP}$ do
        Compute link utilization level $\eta_p$
    end
    sort($\mathbb{PP}$, value=$\eta_p$, order='desc')
    while $\mathbb{PP} \neq Empty$ do
        $\mathbb{PP}_{Top10} = dequeue(\mathbb{PP}, 1:10)$
        sort($\mathbb{PP}_{Top10}$, value='Shortest Path', order='asc')
        for $\mathbb{PP}_{Top10} \neq Empty$ do
            Set $l \leftarrow 1$
            $\mathbb{PP}_l = dequeue(\mathbb{PP}_{Top10}, 1)$
            for Physical path $\mathbb{PP}_p \in \mathbb{PP}_{Top10}$ do
                if $R^{avail}_l \geq R^*_q$ then
                    Set $\mathbb{Z}_j(q,l) \leftarrow 1$
                    Compute $R^{avail}_l \leftarrow R^{avail}_l - R^*_q$
                    $\mathbb{PP} \leftarrow \mathbb{PP}/\mathbb{PP}_l$
                    Return
                end
                $l \leftarrow l + 1$
            end
            $\mathbb{PP} \leftarrow \mathbb{PP}/\mathbb{PP}_{Top10}$
        end
    end
    Set $q \leftarrow q + 1$
    Return $\mathbb{Z}_j(q,p)$
end
```

Figure 2: Locating the optimal DC for VDCE.

$$\mathrm{iF}_1(\mathrm{X}_{ij}) = \begin{cases} \min \sum_{i=1}^{|G_{DC}|} \mathrm{E}\mathbb{P}_{ij} \times \mathrm{X}_{ij} \\ \min \sum_{i=1}^{|G_{DC}|} \ \mathrm{PU\ E}_{ij} \times \mathrm{X}_{ij} \\ \max \sum_{i=1}^{|G_{DC}|} |\mathrm{VDCE}(j)|_i \times \mathrm{X}_{ij} \end{cases}$$

$$\mathbb{X}_{ij} = \begin{cases} 1; \mathrm{Ifthe} i^{\mathrm{th}} \mathrm{DCisselectedforembedding} \\ \mathrm{of\ the} j^{\mathrm{th}} \mathrm{vDCrequest} \\ 0; \mathrm{Otherwise} \end{cases}$$

subject to $\sum_{i=1}^{|G_{DC}|} |\mathrm{X}_{ij} \leq 1_{\}} \cdot \forall j$.

The goal function S1(Xij) in the preceding issue deals with lowering energy expenditures, minimising the PUE index (which implies improved energy efficiency), and consolidating maximum relevant VDC demands on a single DC. Here, the decision variable Xiii is binary and is defined as follows.

5.2 SP2: Locating the optimal physical node for VM placement

This is the second SP, which uses the information acquired from the previous SP to find the appropriate physical node for mapping the VMs of the incoming VDC request. This mapping of VMs to physical nodes not only provides improved energy savings but also achieves the desired SLA. It is attained in accordance with the Algorithm 2.

5.3 SP3: Deducing the optimal path for VL embedding

The best route for VLs helps to lower the VDCE's overall cost. The allocation technique focuses on combining numerous VLs along a physical path in order to lower the overall energy cost. These paths are also subjected to resource availability and shortest path criteria in order to choose the best path for VL embedding. Algorithm 2 explains the overall rationale in the following manner.

The algorithm accepts the following parameters as input: VDC request (jGVDC), DC i for embedding VDC request j(Xij), list of physical paths in the DC (PP), resources available with the pth physical path (Rp), list of virtual links (VLs) to be mapped (VL), and resource demand of the qth VL (Rq). The algorithm accepts one VL at a time and the process continues till all the VLs find the appropriate mappings.

Result:
–• Selection of DC for VDCE with due consideraion to energy prices, energy efficiency, and SLA assurance
Input : –•$\mathbb{EP}_{ij}$: Energy Prices across DCs
–•$\mathbb{PUE}_{ij}$: PUE value of DCs
–•$|\mathbb{VDCE}(j)|_i$: Number of VDCs embedded across DCs associated with the considered VDC request
Output: –•$\mathbb{X}ij$; $\forall i, j$

for *DC* $i \in G_{DC}$ **do**
  for *VDC Requests* $j \in G^{*}_{VDC}$ **do**
    Compute $\mathbb{EP}_{ij}$; $\forall i$
    Compute $\mathbb{PUE}_{ij}$; $\forall i$
    Compute $|\mathbb{VDCE}(j)|_i$; $\forall i$
  end
end
$\mathbb{X}ij \leftarrow$ Solve Eq. (13) against Constraint (Eq. (14))
Return $\mathbb{X}ij$

Figure 3: Algorithm 2:Locating the physical node for VM placement.

6 Observation and Analysis

This section compares the proposed VDCE scheme to baseline algorithms, taking into account several characteristics such as energy efficiency, energy pricing, and SLA adherence. The following are the relevant details.

6.1 Simulation Setup

The proposed scheme's programming was done in MATLAB, and the simulation settings were taken from [23]. For the purposes of evaluation, a total of 5 geographically spread DCs were considered. As noted in Section II, different electricity costs have an impact on these DCs. The DCs are controlled using a fat-tree topology, which is typical of real-world DCs. The network switches each have 16 ports, totaling 320 switches, 1024 physical compute nodes, and 3072 physical paths between them. The compute nodes and physical links each have a CPU capacity of [50, 100]. Backbone links, on the other hand, are more powerful and so have resources in the range of [500, 1000] Mbps.Apart from that, the resource requirements of VDC requests for both VMs and VLs vary between [0, 50]. Furthermore, a physical computing node's baseline energy consumption is set at 150 W, with the node's power varying with CPU utilisation at 2W/CPU Unit. Similarly, the fixed part of the energy consumption for physical switches is set at 300W, whereas the energy consumption of an active port is 5W.

6.2 Baseline Algorithms

The suggested ESP-driven VDCE scheme's performance is compared to three algorithms: random fit, worst fit, and best fit. The VDC requests are randomly mapped/embedded to the available physical nodes in the random fit if the resource limitations are met. The best fit strategy, on the other hand, focuses on mapping all VDC demands to available physical nodes while taking into account resource consumption. On the other side, the poorest fit strategy ignores resource consumption and performs similarly to the best fit method.

6.3 Comparative Analysis

The evaluation is based on energy efficiency, energy pricing, and compliance with SLAs. In detail, the suggested ESP-driven VDCE system (hereafter referred to as "ESP," random fit (RND), worst fit (WRST), and best fit (BEST) techniques consume 403 kW, 436 kW, 455 kW, and 469 kW, respectively, over the simulation duration. This means that the ESP method outperforms the BEST algorithm by around 7.6%. The ESP's conscious effort to condense the VMs and VLs on a small number of active nodes and links is thought to be the cause of this speed. On a similar note, similar results were reported in the case of energy pricing. Supporting VDC requests across different DCs cost an average of 5967.9, 7763.7, 9185.5, and 10701. In this situation, the performance of ESP was found to be the best among its competitors, with a gain of nearly 23%. WRST algorithm, on the other hand, had the lowest results since it ignored both energy pricing and energy efficiency. The proposed design, on the other hand, devoted close attention to selecting the DC with the lowest energy rates for VDCE.

7 CONCLUSION

VDCE is seen as a significant research challenge in growing cloud systems that must be addressed. Effective embedding solutions enable a DC to accommodate a large number of tenants, increasing income for the service provider. In this line, mapping VMs and VLs to appropriate physical nodes and pathways necessitates a thorough examination of DC load factors, regional energy prices, and resource availability. As a result, we introduce the ESP-driven VDCE scheme in this study in order to lower cloud DC energy prices and consumption while maintaining acceptable SLA levels. To achieve the aforementioned goals, we first created a multi-objective optimization problem. The problem at hand, however, is NP-hard and cannot be addressed in real time.Only if the problem is small enough can the solutions be deduced. As a result, the paper proposes an effective heuristic-based solution to the VDCE problem. Splitting the entire problem into different sub-problems, such as DC identification, VM mapping, and VL embedding, makes the offered solutions more effective. The acquired results show that the proposed approach is capable of producing better results in terms of energy efficiency, energy pricing, and SLA adherence.

References

[1] C. Guo, K. Xu, S. Bose, M. Zukerman and G. Shen, "Efficient and green embedding of virtual data centers with mixture of unicast and multicast services", IEEE Transactions on Cloud Computing, 2019.

[2] K. Kaur, S. Garg, N. Kumar, G. S. Aujla, K. R. Choo and M. S. Obaidat, "An Adaptive Grid Frequency Support Mechanism for Energy Management in Cloud Data Centers", IEEE Systems Journal, 2019.

[3] R. Ranjan, "Streaming Big Data Processing in Datacenter Clouds", IEEE Cloud Computing, vol. 1, no. 1, pp. 78-83, 2014.

[4] K. Kaur, S. Garg, G. Kaddoum, E. Bou-Harb and K. R. Choo, "A Big Data-Enabled Consolidated Framework for Energy Efficient Software Defined Data Centers in IoT Setups", IEEE Transactions on Industrial Informatics, vol. 16, no. 4, pp. 2687-2697, 2020.

[5] K. Kaur, S. Garg, G. Kaddoum, F. Gagnon, N. Kumar and S. H. Ahmed, ”An Energy-driven Network Function Virtualization for Multi-domain Software Defined Networks”, IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS), pp. 121-126, April 2019.

[6] A. Hbaieb, M. Khemakhem and M. B. Jemaa, ”A survey and taxonomy on virtual data center embedding”, The Journal of Supercomputing, pp. 1-37, 2019.

[7] C. Liang and F. R. Yu, ”Wireless Network Virtualization: A Survey Some Research Issues and Challenges”, IEEE Communications Surveys Tutorials, vol. 17, no. 1, pp. 358-380, 2015.

[8] R. Jain and S. Paul, ”Network virtualization and software defined networking for cloud computing: a survey”, IEEE Communications Magazine, vol. 51, no. 11, pp. 24-31, 2013.

[9] M. Li, C. Chen, C. Hua and X. Guan, ”Intelligent Latency-Aware Virtual Network Embedding for Industrial Wireless Networks”, IEEE Internet of Things Journal, vol. 6, no. 5, pp. 7484-7496, 2019.

[10] K. Kaur, S. Garg, G. Kaddoum, S. H. Ahmed and D. N. K. Jayakody, ”En-OsCo: Energy-aware Osmotic Computing Framework Using Hyperheuristics”, Proceedings of the ACM MobiHoc Workshop on Pervasive Systems in the loT Era ser. PERSIST-loT ’19, pp. 19-24, 2019.

[11] Y. Yang, X. Chang, J. Liu and L. Li, ”Towards Robust Green Virtual Cloud Data Center Provisioning”, IEEE Transactions on Cloud Computing, vol. 5, no. 2, pp. 168-181, 2017.

[12] Z. Li, Z. Lu, S. Deng and X. Gao, ”A Self-Adaptive Virtual Network Embedding Algorithm Based on Software-Defined Networks”, IEEE Transactions on Network and Service Management, vol. 16, no. 1, pp. 362-373, 2018.

[13] J. Pei, P. Hong, K. Xue and D. Li, ”Efficiently embedding service function chains with dynamic virtual network function placement in geo-distributed cloud system”, IEEE Transactions on Parallel and Distributed Systems, vol. 30, no. 10, pp. 2179-2192, 2018.

[14] E. S. Correa, L. A. Fletscher and J. F. Botero, ”Virtual Data Center Embedding: A Survey”, IEEE Latin America Transactions, vol. 13, no. 5, pp. 1661-1670, 2015.