Resource ratio based virtual machine placement in heterogeneous cloud data centres

SAIKISHOR JANGITI¹, SRIRAM E², ROHITH JAYARAMAN², HARINI RAMPRASAD² and V S SHANKAR SRIRAM^{1,*}

¹Centre for Information Super Highway (CISH), SASTRA Deemed University, Thanjavur, India

²School of Computing, SASTRA Deemed University, Thanjavur, India

e-mail: jsaikishor@gmail.com; sriram@it.sastra.edu

MS received 20 April 2018; revised 17 September 2019; accepted 10 October 2019

Abstract. Server consolidation through virtualization improves resource utilization significantly in Cloud Data Centres (CDCs). We study the case of a CDC hosting heterogeneous Physical Machines (PMs) as a variable size vector bin-packing problem. The PMs have different configurations of multiple resources like CPU, RAM, Disk Storage and Network Bandwidth. In this paper, we propose PMNeAR-vector heuristic for PM selection in PM-heterogeneity aware Virtual Machine (VM) initial placement. The proposed heuristic is compared with well-known heterogeneity aware FFD-DRR and BFD bin centric heuristics using a dataset with random instances of both VMs and PMs of heterogeneous configurations. Fifty rounds of VM initial placement simulation experiments were conducted to validate the average resource wastage. The results show that on average FFD-DRR and BFD bin centric heuristics are wasting 22.62% and 37.27% more resource units compared to the proposed PMNeAR-vector heuristic.

Keywords. Virtual machine placement; first-fit decreasing; resource wastage minimisation; resource ratio.

1. Introduction

Infrastructure-as-a-Service (a cloud service delivery model) has become an alternative to the traditional IT infrastructure owing to its hassle-free provision and management of computing infrastructure. Based on their requirements, the end-users can choose among the standard Virtual Machine (VM) instances with fixed resource offerings or request for a customized VM instance. The requested amount of each resource (CPU, RAM, NBW and Disk Storage), all from one Physical Machine (PM) are bundled and provided to end-user using virtualization. Placing the incoming VM requests onto a subset of PMs with the least total cost is known as VM initial placement, analogous to an NP-Hard problem called Vector Bin-Packing (VBP) problem, also known as a multi-capacity bin-packing problem. In our recent works, we proposed a bin-packing based virtual resource allocation method called VMNeAR (Virtual Machine Nearest and Available to Residual resource ratios of a physical machine) [1] and a hybrid bin-packing heuristic ENSEMBLE-HIDE-SPADE for a Cloud Data Centre (CDC) hosting homogeneous PMs.

The current research problem is to minimize the total resource wastage across the running PMs in a CDC hosting

heterogeneous PMs, analogous to another NP-Hard problem called Variable Size Vector Bin-Packing (VSVBP) problem. Here we extend our VMNeAR [1] to PMNeAR (Physical Machine Nearest to Average Resource ratios of remaining virtual machines) as a heuristic approach for VSVBP.

1.1 Variable Size Vector Bin-Packing (VSVBP) problem

The standard Bin-Packing Problem (BPP) is solved by packing a group of one-dimensional (1D) items into multiple 1D bins of equal capacity. The target is that it must utilize a minimum number of 1D bins. A solution for BPP can be called feasible; if each 1D item is placed only once and in one bin. Also, the sum of all the 1D item sizes placed in a 1D bin should not exceed the capacity of that bin.

Figure 1 depicts the architecture model of the CDC with heterogeneous PM clusters employed in the simulation experimentation. The 1D items in BPP can be seen as single-capacity items, whereas both VM and PM has multiple resource demands and resource capacities, respectively. We can draw an analogy to VM initial placement in a heterogeneous CDC from VSVBP using VM as an item and PM as a bin. As such, two requirements that must be met are: every VM must be placed in only one PM, and for

*For correspondence

Figure 1. A cloud data centre hosting heterogeneous PMs.

each resource capacity in a PM, the sum of corresponding resource demand of all the VM requests placed in that PM should not exceed the (corresponding) resource capacity of that PM. The VSVBP generalized bin-packing problem observes both items and bins of varying capacities.

1.2 VSVBP and heterogeneity aware heuristics

The dominant-resource based heuristic FFD-DRR (First Fit Decreasing-Dominant Residual Resource) [2] that accounts for heterogeneous resource demands and BFD (Best-Fit Decreasing) Bin Centric [3] can be best compared with the current research. FFD places the item in the first bin, where that item fits. Otherwise, it opens a new bin. FFD feeds the items in decreasing sequence, in order to fit the larger items first and the smaller (less resource-demanding) items later. FFD-DRR heuristic splits the VMs into multiple groups (based on their dominating resource) and in decreasing order of size, similar to FFD. Based on the PM's dominating residual resource, the largest VM from the corresponding resource dominant VM group is placed by FFD-DRR. BFD Bin Centric attempts to pack the largest available item in the smallest bin according to their proposed measures for weighing items (VMs) and bins (PMs).

However, there is a limitation in the existing approaches for VM initial placement, i.e., they are either oblivious to PM selection or indirectly selecting using scalar weights for PMs. Our proposed PMNeAR heuristics are selecting PMs directly without converting the resource vectors to scalar weights.

The rest of the paper is organized as follows. Section 2 discusses the related work in solving the VM placement problem. The proposed heuristics PMNeAR and PMNeAR-vector are detailed in section 3. Section 4 analyses and discusses the obtained empirical results. Finally, the conclusions are provided in section 5.

2. Related work

Several works solved the virtual-resource allocation problem called VM placement; broadly, the works can be grouped as heuristics and metaheuristics. While heuristics

solve fast and produce near-optimal solutions, metaheuristics are time-consuming but may result in an optimal solution in most cases. However, there is no exact algorithm for the VM placement problem, which can always produce an optimal solution in polynomial time complexity. Some of the state-of-the-art heuristics and metaheuristics used to solve VM placement in the last decade are enumerated in table 1.

3. Proposed PMNeAR heuristics

The significant difference in VBP and VSVBP is all the bins are of equal capacity in VBP, whereas they are of different costs and capacities in VSVBP. The proposed PMNeAR heuristics can be used to augment the VBP methods into VSVBP methods. In VSVBP, choosing a PM into which the items are to be placed next is an issue which can be solved by PMNeAR, a PM choosing heuristic that works based on Resource Ratio Vector (RRV).

3.1 Resource ratio vector (RRV)

In VBP and VSVBP, both VMs (items) and PMs (bins) are represented by a vector of length d (number of dimensions or resources). RRV of a VM, denoted by r is the collection of ratios of $d - 1$ resource requirements to anyone resource requirement that is chosen as a pivot. RRV is computed using Equation (1).

$$RRV(r) = \left(\frac{resReq_2}{resReq_1}, \frac{resReq_3}{resReq_1}, \frac{resReq_4}{resReq_1}, \dots, \frac{resReq_d}{resReq_1} \right), \quad (1)$$

where is the i^{th} requirement of a VM request.

The pivot ($resReq_1$) used to find the resource ratios will be useful as *size*, when calculating the second part in Equation (2), i.e., when the RRV is entirely same for more than one VM request. The *size* attribute helps in finding the largest VM request.

Thus the original VM representation vector $\langle resReq_1, resReq_2, \dots, resReq_d \rangle$ will now be represented as shown in Equation (2).

$$\langle size, r \rangle = \left\langle resReq_1, \left\langle \frac{resReq_2}{resReq_1}, \frac{resReq_3}{resReq_1}, \frac{resReq_4}{resReq_1}, \dots, \frac{resReq_d}{resReq_1} \right\rangle \right\rangle \quad (2)$$

As the VMNeAR [1] heuristic results proved, there would be lesser resource wastage in the pivot resource when compared to other resources. Since CPU cores are precious, we used CPU cores as a pivot. VMNeAR is a geometric heuristic, and it attempts to fill all the other resources proportionately to the pivot resource.

Table 1. State of the art literature on VM placement.

Authors	Algorithm(s)
Xu and Fortes, 2010	Grouping Genetic Algorithm (GGA) [4]
Panigrahy <i>et al.</i> , 2011	FFDSum, FFDProd, Dot Product [5]
Wang <i>et al.</i> , 2013	Particle Swarm Optimization (PSO) [6]
Gao <i>et al.</i> , 2013	VMPACS [7], a VM placement based on ant colony system optimisation
Canali <i>et al.</i> , 2017	Class based placement [8]
Satpathy <i>et al.</i> , 2017	Crow search based virtual machine placement [9]
Jangiti <i>et al.</i> , 2019	First Fit Decreasing-Aggregated Rank [10]
Jangiti <i>et al.</i> , 2019	ENSEMBLE-HIDE-SPADE [11]

The VM and PM representation $\langle size, r \rangle$ supports the search for better PMs and VMs (a VM or PM results in minimum resource wastage). VMNeAR is used to select the best VM that minimizes resource wastage. While placing VMs, the residual resource capacities of a PM change. Hence in the case of PM representation, RRV is called as residual RRV.

3.2 Near function

The *Near* function used in both VMNeAR and PMNeAR is formally defined below.

```

Function Near( $V, S$ )
{
1.  for each  $i$  from 1 to  $|V|$ 
2.     $Near \leftarrow \emptyset$ 
3.    for each  $j$  from 1 to  $|S|$ 
4.       $D_j \leftarrow |V_i - S_{ji}|$ 
5.    end for
6.     $dist \leftarrow \text{minimum}(D)$ 
7.    for each  $j$  from 1 to  $|S|$ 
8.      if  $D_j == dist$ 
9.         $Near \leftarrow Near \cup S_j$ 
10.   end for
11.   if  $|Near| == 1$ 
12.     Return  $Near_1$ 
13.    $S \leftarrow Near$ 
14. end for
}

```

The *Near* function takes two arguments V and S . V is a vector and S is a set of distinct vectors (all the vectors are of length $d - 1$). Let V_i denote the i^{th} tuple of vector V and S_{ji} denote the i^{th} tuple of the j^{th} vector in S . *Near*(V, S) returns a vector from S that is nearest to V .

Both VMNeAR and PMNeAR use the function *Near* for the computation of nearest RRV, i.e., a VM group nearest to PM RRV and a PM nearest to average RRV of all the VMs respectively. Our proposed PM-NeAR heuristics choose the best PM that will minimise the resource wastage and then use VMNeAR to place the best VMs based on that PM's residual resource ratios.

3.3 VMNeAR

The VMs available for placement are grouped based on Resource Ratio Vectors. All VMs with the same r are grouped and the size ($resReq_1$ i.e., CPU Cores) attribute is used to identify the largest VM from the group.

Case study: Consider the current PM's residual resource ratios are 1.4, 2.3, 3.2 also, the available VM groups are with the following resource ratios: $\langle 1, 2, 3 \rangle$, $\langle 2, 1, 3 \rangle$, $\langle 1, 2, 4 \rangle$, $\langle 4, 3, 2 \rangle$.

VMNeAR: For the PM's first residual resource ratio 1.4, 1 is nearest among the VM's first resource ratios $\{1, 2, 4\}$. Now, there are two VM groups $\langle 1, 2, 3 \rangle$ and $\langle 1, 2, 4 \rangle$, with the first resource ratio as 1. The same is the case for the second resource ratio too. The third resource ratio 3.2 is nearer to $\langle 1, 2, 3 \rangle$ than $\langle 1, 2, 4 \rangle$. Hence $\langle 1, 2, 3 \rangle$ is the nearest VM group to the PM's current residual resource ratios according to VMNeAR.

3.4 Proposed PMNeAR-vector heuristic

The proposed PMNeAR-vector is a PM selection method for VM placement in heterogeneous PMs, a VSVBP problem. There is no concept of direct PM selection in the existing methods of VM placement. The existing FFD and BFD heuristics convert resource vectors of both VMs and PMs into scalar weights. Based on the scalar weights, they attempt to place the largest item in the smallest possible bin [3].

Figure 2. The flow of the proposed PMNeAR-vector heuristic.

Table 2. Resource wastage by VM placement algorithms.

Resources	VM Placement Algorithms used					
	FFD-ARank [10]	FFD-DRR [2]	VMNeAR [1]	BFD Bin Centric [3]	Proposed PM NeAR	Proposed PMNeAR-vector
Wasted (W) remains unallocated in those PMs, hosting at least one VM						
CPU Cores	314637	334246	318043	373478	317961	183085
RAM (GB)	235073	258668	237154	319869	252734	160003
Network Bandwidth (Gbps)	334759	348430	332954	443744	294145	310217
Disk Storage (GB)	249261	262749	251077	283091	193123	217106
All resources	1133730	1204093	1139228	1420182	1057963	870411
Available (A) in those PMs, hosting at least one VM						
CPU Cores	461491	481100	464897	520332	464815	329939
RAM (GB)	442783	466378	444864	527579	460444	367713
Network Bandwidth (Gbps)	430836	444507	429031	539821	390222	406294
Disk Storage (GB)	403467	416955	405283	437297	347329	371312
All resources	1738577	1808940	1744075	2025029	1662810	1475258
Available in all the PMs in the cloud data centre						
CPU Cores	4176404	4176404	4176404	4176404	4176404	4176404
RAM (GB)	4303968	4303968	4303968	4303968	4303968	4303968
Network Bandwidth (Gbps)	4204842	4204842	4204842	4204842	4204842	4204842
Disk Storage (GB)	4138887	4138887	4138887	4138887	4138887	4138887
All resources	16824101	16824101	16824101	16824101	16824101	16824101
Allocated in those PMs, hosting at least one VM						
CPU Cores	146854	146854	146854	146854	146854	146854
RAM (GB)	207710	207710	207710	207710	207710	207710
Network Bandwidth (Gbps)	96077	96077	96077	96077	96077	96077
Disk Storage (GB)	154206	154206	154206	154206	154206	154206
All resources	604847	604847	604847	604847	604847	604847
Normalized Wastage ($\frac{W}{A}$)						
CPU Cores	0.682	0.695	0.684	0.718	0.684	0.555
RAM (GB)	0.531	0.555	0.533	0.606	0.549	0.435
Network Bandwidth (Gbps)	0.777	0.784	0.776	0.822	0.754	0.764
Disk Storage (GB)	0.618	0.630	0.620	0.647	0.556	0.585
All resources	2.607	2.663	2.613	2.793	2.543	2.338
Average Normalized Wastage in all the resources	0.652	0.666	0.653	0.698	0.636	0.585

Figure 3. Comparison of total resource wastage.

The flow of PMNeAR-vector is shown in figure 2. Until all the VMs are placed, a PM is chosen using *Near()* each time as discussed in section 3.2 and is filled with VMs suggested by VMNeAR as discussed in section 3.3. If the PMs exhausted without placing all the given VM requests, the situation is called as a failure.

The following example illustrates the PM selection. Consider the set of VM requests: $\{ \langle 1,2,3,4 \rangle, \langle 2,3,4,1 \rangle, \langle 3,4,1,2 \rangle \}$. Now the matrix M and row vector determined as per Step 2 are:

Figure 4. Comparison of average normalized resource wastage.

$$M = \begin{bmatrix} 2 & 3 & 4 \\ 1.5 & 2 & 0.5 \\ 1.33 & 0.33 & 0.67 \end{bmatrix} \text{ and Vector} = \langle 1.61, 1.77, 1.72 \rangle$$

Suppose the RRVs of the available PMs are $\{\langle 1, 2, 3 \rangle, \langle 2, 3, 4 \rangle, \langle 2, 4, 3 \rangle, \langle 3, 1, 2 \rangle\}$. According to the *Near()* discussed in section 3.2 the better PM which minimizes the resource wastage is the PM with $RRV = \langle 2, 3, 4 \rangle$.

3.5 Proposed PMNeAR heuristic

The working of PMNeAR is similar to PMNeAR-vector, except that the vectors are replaced with scalar values (which are obtained by taking the mean of the vector) in the immediate next step to PM availability check condition (in figure 2). In the above example, after converting vectors to scalars *Near*(1.7, {2, 3, 3, 2}) suggests the PM with $RRV = \langle 1, 2, 3 \rangle$. A tie in *Near* computation is broken by taking the first PM in the list. To demonstrate the superiority of straight near-vector computation over the introduction of scalars, the PMNeAR is used, and the results in section 4 demonstrate that PMNeAR is better than other VM placement methods except for PMNeAR-vector.

4. Results and discussion

All the VM initial placement heuristic algorithms namely VMNeAR [1], FFD-DRR [2], FFD-ARank [10], BFD Bin Centric [3] and the two proposed PMNeAR variants were implemented in a Python 2.7 environment having an Intel® Core™ i7 processor @ 2.5 GHz system with 4 GB RAM running Windows 10 operating system. A synthetic dataset had been generated with 50 schedules of VSVBP instances and was used to evaluate the packing efficiency of the proposed PMNeAR variants. Table 2 lists the simulation results. A comparison of the total and average resource wastage for all the initial placement heuristic algorithms is shown in figures 3 and 4, respectively. The VBP algorithms are extended for VSVBP by choosing the PMs sequentially.

5. Conclusion

The proposed PMNeAR heuristics enhances the revenue of cloud service providers by allowing them to handle more requests. Simulation experiments have been conducted to compare the total resource wastage of our proposed heuristics and the heterogeneity aware FFD-DRR, BFD bin centric along with two of our earlier works in the direction of resource wastage minimization. The results have shown that the proposed PMNeAR-vector heuristic can reduce the total resource wastage by 37.27% compared to BFD bin centric heuristic.

The proposed PMNeAR heuristics are the first vector (multi-capacity) bin-packing heuristics for VSVBP without

scalar conversion. The rationale behind the supremacy of our vector-based heuristic approach is as follows.

- (i) Collision is possible while converting vector items to scalar weights. As a result, two or more completely different vector items may be assigned with same scalar weights, which leads to the wrong choice of PM and VM selection by the BFD bin-centric heuristic.
- (ii) FFD-DRR is focusing on a single dominant resource for VM selection and is unaware of PM selection.

Acknowledgements

The authors thank the Department of Science and Technology for their financial support (SR/FST/ETI-349/2013) under Fund for Improvement of S&T Infrastructure in Universities and Higher Educational Institutions.

References

- [1] Jangiti S and Sriram V S S 2018 Scalable and direct vector bin-packing heuristic based on residual resource ratios for virtual machine placement in cloud data centers. *Comput. Electr. Eng.*
- [2] Zhang Y and Ansari N 2013 Heterogeneity aware dominant resource assistant heuristics for virtual machine consolidation. In: *GLOBECOM - IEEE Glob. Telecommun. Conf.*, pp. 1297–1302
- [3] Gabay M and Zaourar S 2016 Vector bin packing with heterogeneous bins: application to the machine reassignment problem. *Ann. Oper. Res.* 242(1): 161–194
- [4] Xu J and Fortes J A B 2010 Multi-objective virtual machine placement in virtualized data center environments. In: *Proceedings - 2010 IEEE/ACM International Conference on Green Computing and Communications, GreenCom 2010, 2010 IEEE/ACM International Conference on Cyber, Physical and Social Computing, CPSCom 2010*, Dresden, Germany, pp. 179–188
- [5] Panigrahy R, Talwar KUyeda L and Wieder U 2011 Heuristics for Vector Bin Packing. *Res. Microsoft*, pp. 1–14
- [6] Wang S, Liu Z, Zheng Z, Sun Q, and Yang F 2013 Particle swarm optimization for energy-aware virtual machine placement optimization in virtualized data centers. In: *Proceedings of the International Conference on Parallel and Distributed Systems - ICPADS*, 2013, pp. 102–109
- [7] Gao Y, Guan H, Qi Z, Hou Y and Liu L 2013 A multi-objective ant colony system algorithm for virtual machine placement in cloud computing. *J. Comput. Syst. Sci.* 79(8): 1230–1242
- [8] Canali C and Lancellotti R 2017 Scalable and automatic virtual machines placement based on behavioral similarities. *Computing* 99(6): 575–595
- [9] Satpathy A, Addya S K, Turuk A K, Majhi B and Sahoo G 2017 Crow search based virtual machine placement strategy in cloud data centers with live migration. *Comput. Electr. Eng.*, Dec.

- [10] Jangiti S, Ram E S and Sriram V S S 2019 Aggregated Rank in First-Fit-Decreasing for Green Cloud Computing. In: *Cognitive Informatics and Soft Computing*, Springer, pp. 545–555
- [11] Jangiti S, Sri Ram E, Ravi L and Sriram V S S 2019 Scalable hybrid and ensemble heuristics for economic virtual resource allocation in cloud and fog cyber-physical systems. *J. Intell. Fuzzy Syst.* 36(6): 4519–4529. <https://doi.org/10.3233/JIFS-179004>