# Metaheuristic approaches to virtual machine placement in cloud computing: a review

Alboaneen, Dabiah Ahmed; Tianfield, Huaglory; Zhang, Yan

*Published in:*
15th International Symposium on Parallel and Distributed Computing (ISPDC), 2016

*DOI:*
10.1109/ISPDC.2016.37

*Publication date:*
2017

*Document Version*
Author accepted manuscript

# Metaheuristic Approaches to Virtual Machine Placement in Cloud Computing: A Review

Dabiah Ahmed Alboaneen*†, Huaglory Tianfield*, and Yan Zhang*
*School of Engineering and Built Environment, Glasgow Caledonian University, Glasgow, G4 0BA, United Kingdom
†College of Education, University of Dammam, Jubail, Kingdom of Saudi Arabia
Email: {dabiah.alboaneen; h.tianfield; yan.zhang}@gcu.ac.uk

***Abstract*—Virtual machine placement (VMP) is mapping virtual machines (VMs) to appropriate physical machines (PMs) to achieve satisfactory objectives such as minimised energy consumption or maximised performance. VMP is considered as a non-deterministic polynomial-time hard (NP-hard) problem. Metaheuristic techniques are able to find near-optimal solutions to NP-hard problems. This paper presents a review upon metaheuristic approaches to VMP in cloud computing.**

***Index Terms*—cloud computing; metaheuristic; virtual machine allocation; virtual machine placement.**

## I. INTRODUCTION

Cloud computing provides on-demand access to a shared pool of resources on a pay-as-you-go model with a guaranteed quality of service (QoS) to users. If the desired performance is not achieved, the users will hesitate to pay. To ensure meeting the QoS to users, it is necessary that virtual machines (VMs) are efficiently mapped to given physical machines (PMs). The process of mapping VMs to PMs is known as virtual machine placement (VMP). Obviously, VMP is one of the major issues in cloud computing.

The VMP problem in cloud computing is a kind of a bin-packing problem and a non-deterministic polynomial-time hard (NP-hard) problem [1]. Generally, it is difficult to develop algorithms for producing optimal solutions within a short time for this type of problems. Metaheuristic techniques can deal with these problems by providing near-optimal solutions within a reasonable time. Metaheuristics have become popular in the past years due to their efficiency to solve large and complex problems.

There are several surveys on VMP in cloud computing which mainly focus on specific issues such as energy-efficient techniques for resources allocation [2], [3] and [4], power-aware dynamic VMP algorithms based on bin-packing strategy [5]. Reviews on VMP literature also present different classifications [6], [7] and [8].

The remainder of this paper is organised as follows. Section II provides an overview of VMP. The current metaheuristic algorithms for VMP are reviewed in sections III, IV and V. Next, observations are discussed in Section VI to explore future research in this area. Final conclusions are presented in Section VII.

## II. OVERVIEW OF VIRTUAL MACHINE PLACEMENT

To solve VMP problem, we need to consider the optimisation algorithm, initial condition, objective function and experiment/simulation of cloud computing. In this paper, we focus on metaheuristic techniques for VMP namely simulated annealing (SA), genetic algorithm (GA), ant colony optimisation (ACO), particle swarm optimisation (PSO) and biogeography-based optimisation (BBO).

There are two types of initial conditions for VMP problems: (1) fresh VMP where a new VM is placed on PM, and (2) VM re-placement which is the optimisation of the existing placement of VMs. The main difference is that in VM re-placement, live VM migration is used to move a VM from one PM to another without noticeable service interruption [1].

The need for re-placing VMs is due to the change in the data centre (DC) environment, such as workload variations or hardware failures. Generally, applications located in VMs are usually associated with service level agreement (SLA). After a period of time, violations of SLA may occur due to factors such as high CPU utilisation or high memory usage of the PM. Hence, some VMs need to be migrated to avoid over-utilisation that causes VM performance degradation. On the other hand, some PMs may be switched off or turned to low-power modes to reduce the energy consumed by the underutilised PMs.

A number of metaheuristic algorithms have been used to solve the VMP problems in order to optimise either energy consumption, QoS, resource utilisation or all of them. The main objective functions for the optimisation of VMP in a cloud is illutrateded in Fig 1.

Metaheuristics can be classified into two categories: (i) individual-based metaheuristics (IBMs) which modify and improve a single candidate solution (e.g. SA) and (ii) population-based metaheuristics (PBMs) which improve multiple candidate solutions and use population characteristics to guide the search (e.g. ACO, PSO and GA). Moreover, PBMs can be classified based on process strategies into (i) PBMs with reproductive strategies which reproduce new solutions or generations (e.g. GAs) and (ii) PBMs with non-reproductive strategies (e.g. BBO). The taxonomy can be illustrated in Fig 2.

## III. INDIVIDUAL-BASED ALGORITHMS

### A. Simulated Annealing

SA was proposed by Kirkpatrick *et al.* [9]. It is inspired by nature behaviour. In metallurgy, annealing is a technique involving heating and controlled cooling of a material to

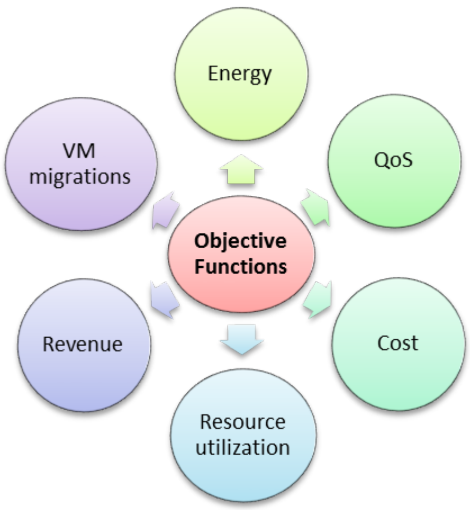


Fig. 1: Objective functions in VMP

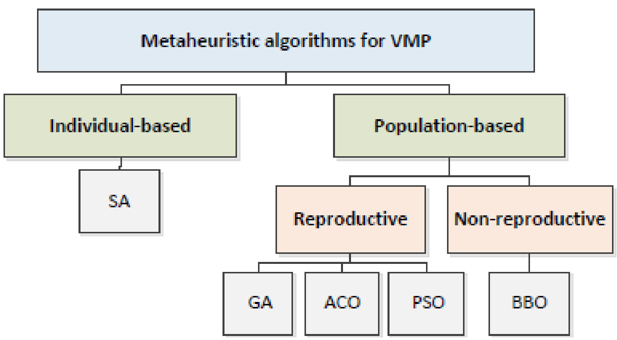


Fig. 2: Metaheuristic algorithms for VMP

increase the size of its crystals and reduce their defects. The pseudocode of SA can be presented in Algorithm 1.

**Algorithm 1** SA

```
Generate an initial solution S0 & initial temperature T0
while termination condition not met do
  Initialise a neighbour S1 of S0 randomly
  if fitness(S1) < fitness (S0) then
    Set S0 ← S1
  end if
end while
return the final solution
```

A SA-based algorithm to solve the VMP problem (SAVMP) and optimise the power consumption was proposed, for the first time, by Wu *et al.* [10]. The proposed algorithm was a single-objective and considered two resources which were CPU and memory. To evaluate the performance of SAVMP, it was simulated and ran 10 times. The average percentage of energy saving was compared to first fit decreasing (FFD) and multi-start random searching (MSRS) algorithms. The results demonstrated that the SA algorithm performed better than the others. It saved more energy than FFD by 0-25% in an acceptable time frame. In addition, SAVMP was also better than MSRS, which only performed well in small sized problems.

Another VMP algorithm based on SA was proposed by Khalilzad *et al.* [11]. It also aimed to minimise energy consumption in cloud DC. VMs were consolidated in a minimum number of PMs while meeting the time requirement of VMs. The VM consolidation problem was formulated as an integer linear optimisation to minimise the total power of the set of PMs. The work considered three allocation levels while most of the existing works only considered one of these levels. The three levels were: (1) from task to VM, global EDF (gEDF) was used for task allocation, (2) from VM to core allocation, by using the worst fit (WF), and (3) the VM placement algorithm, by using a combined max-min ant system (MMAS) and SA algorithms. However, the work assumed homogeneous PMs. In addition, there were no experiment results presented for the proposed algorithms.

It can be noticed from [10] and [11] that the proposed algorithms focused only on minimising energy consumption and ignored the QoS in a cloud DC. In addition, the dynamic nature of the workload was not been taken into account. Marotta and Avallone [12] proposed a novel mixed integer linear programming (MILP) model for the VM re-placement problem based on the SA algorithm. The goal was to determine the set of VM migrations that minimised the linear combination of the power consumption of the active PMs normalised to the total initial power and the number of migrations normalised to the number of VMs. The algorithm was implemented in Java and compared with FFD and Sercon. The simulation results showed that the proposed algorithm had a better reduction than FFD: between 27% and 37% in the number of active PMs, and between 31% and 44% in the power consumption. The comparison with Sercon demonstrated that the proposed algorithm also had a better reduction in the number of active PMs in a range of 9%-17% and of 14%-24% for the energy consumption. In addition, the authors compared the results with best fit (BF), first fit (FF) and random policies. Although the number of consolidated PMs was lower than the one achieved with the random allocation, the proposed algorithm still outperformed BF and FF. However, the proposed algorithm focused only on minimising energy consumption and ignored the QoS, as in the previous studies. Therefore, QoS needs to be investigated besides energy when formulating the VMP problem.

## IV. Reproductive Population-Based Algorithms

### A. Genetic Algorithm

GA was first proposed by Holland in 1975 [13]. It generates solutions using techniques inspired by natural evolution, such as selection, crossover and mutation. The pseudocode of GA can be presented in Algorithm 2.

**Algorithm 2** GA

1: Generate a population.
2: Evaluate population using fitness function.
3: **while** termination condition not met **do**
4: Select the chromosomes using selection operator for reproduction.
5: Apply the crossover operation on the pair of chromosomes obtained in step 4.
6: Apply the mutation operation on the chromosome.
7: Evaluate the fitness value of new generated chromosomes "offsprings".
8: Update the population by replacing bad solutions with better chromosomes from offsprings.
9: **end while**
10: **return** best chromosome as the final solution.

GA has been extensively used in the literature to solve the VMP problem in order to optimise different objective functions. Xu *et al.* [14] studied the VMP as a multi-objective optimisation problem. They proposed a modified GA called Grouping GA (GGA) for efficiently searching global optimal solutions. The objectives to be met were the minimisation of total resource wastage, power consumption and thermal dissipation cost. In order to combine these different objectives, a fuzzy-logic based evaluation approach was developed to obtain a suitable fitness function regarding all the objectives. The authors considered two different levels for resource allocation. The first level was to allocate VMs to resources and the second one was to allocate VMs to PMs. The proposed algorithm was compared with four bin-packing algorithms: two FFD algorithms (FFD-CPU and FFD-MEM) and two best fit decreasing algorithms (BFD-CPU and BFD-MEM) and two single-objective approaches using power-consumption and thermal-dissipation models (SGGA-P and SGGA-T). The simulation results showed that the proposed algorithm had good performance, scalability and robustness.

However, Jiang *et al.* [15] claimed that GA was able to provide the best solution, but had poor stability. The authors formulated the energy-efficient initial VMP problem in cloud DCs by taking into account multiple resources. Three heuristic algorithms (i.e. FF, next fit (NF), BF) and GA were presented to minimise the energy consumption and maximise the QoS. However, the proposed algorithms were simulated on CloudSim and evaluated with homogeneous PMs.

VMs in a DC can communicate with each other through communication devices, such as switches, which also consume an amount of energy that needs to be minimised. Wu *et al.* [16] proposed a single-objective GA for VMP in cloud DCs. They considered energy consumption in communication networks as well as in PMs in the proposed algorithm. The authors assumed a three-tier architecture for the DC. The proposed algorithm was implemented in Java and compared with the FFD heuristic. According to the results, GA could reduce energy consumption more efficiently than FFD: the solutions produced by the proposed algorithm were $3.5-23.5\%$ better than those produced by FFD.

An extension of [16] was proposed by Tang and Pan [17]. The authors proposed a hybrid GA (HGA) for the VMP problem that considered the energy consumption in both PMs and the communication network in a DC. The HGA extended the GA approach using a repairing procedure and a local optimisation procedure, which were used to enhance the exploitation capacity and the convergence of the original GA. The main aim of the local optimisation was to minimise the number of PMs used in VM allocation. Experimental results showed that the HGA significantly outperformed the original GA, and also the HGA was scalable when the number of VMs and PMs increased. The mean total energy consumption of the HGA for the 30 different test problems with the same configuration was $27.36-43.90\%$ less than that of the original GA while the mean computation time of the HGA was reduced by $73.30-88.61\%$.

Similar to [16] and [17], Yang *et al.* [18] presented a novel VMP and traffic configuration algorithm (VPTCA) using GA to minimise the power consumption in a DC network. However, in VPTCA, interrelated VMs were assigned into the same PM or pod to reduce the amount of transmission load. In the layer of traffic message, VPTCA optimally used switch ports and link bandwidth to balance the load and avoid congestions, enabling the DC network to increase its transmission capacity, and saving a significant amount of network energy. The proposed algorithm was evaluated via NS-2 simulations and compared with two DC network management algorithms, global FF and ElasticTree. The experimental results showed that VPTCA outperformed those two algorithms in providing DC network more transmission capacity while consuming less energy. Particularly, VPTCA saved energy by 29.2% and 25.6% compared with Global FF and ElasticTree.

Liu *et al.* [19] proposed a multi-objective VMP algorithm based on GA to simultaneously minimise the number of active PMs, communication traffic and balance multidimensional resources. The improved multi-objective algorithm incorporated the non-dominated sorting genetic algorithm (NSGA-II) into the grouping GA (GGA). To validate the proposed algorithm, the authors compared it with four algorithms: GGA [14], BA, cluster and cut and greedy algorithm. The results claimed that the proposed algorithm outperformed other algorithms because it adopted not only NSGA-II to approach the pareto-optimal front but also GG operators to avoid self-stagnating in the process of evolution. In addition, GGA and BA achieved the second-least number of active PMs because they both aimed to consolidate VMs into a smaller number of PMs, and so resulted in fewer active PMs. Among the five algorithms, greedy was the worst in all the objectives because it only had a simple rule to place VMs.

Maximising the economical revenue for cloud providers was one of the objective functions of VMP algorithm proposed by Pires and Barn [20]. The authors proposed, for the first time, a purely multi-objective formulation for the VMP problem. In order to solve the formulated problem, a novel multi-objective memetic algorithm was proposed to minimise the energy consumption and network traffic and maximise the economical revenue. The experimental tests were run with real data of PMs, VMs and traffic network among VMs from the Itaipu Technological Park DC in Paraguay. The proposed algorithm was run with different scenarios and experimental results were compared to the exact solution obtained using an exhaustive search algorithm when possible. The results showed that the proposed algorithm found the complete Pareto front (100%).

Pascual *et al.* [21] proposed an enhancing placement policy based on GA with network-aware optimisations, trying to simultaneously improve application performance, resource and power efficiency. Experiments demonstrated that allocating applications using optimisation-based policies (i.e., NSGA-II, strength pareto evolutionary 2 (SPEA2) and hypervolume estimation (Hype)) resulted in a lower utilisation of resources while improving the performance of applications.

Adamuthe *et al.* [22] formulated a VMP as a multi-objective optimisation problem. The objectives were maximising profit, load balancing and minimising the resource wastage. Results of GAs, NSGA and NSGA-II were compared with common solution representations, penalty and benefit values. All the three algorithms reported good solutions whereas GA and NSGA were subjected to premature convergence and duplicate solutions. NSGA-II gave a good and diversified range of solutions.

Kaaouache and Bouamama [23] proposed a hybrid GA using BFD (HGBF-BP) to deal with infeasible solutions because of the bin-used representation. Due to infeasible chromosomes exceeding the bin capacity, the BFD packing strategy was proposed to place that package and repair the chromosome. The aim of the proposed algorithm was to minimise the total number of PMs used and therefore to minimise the energy consumption. The HGBF-BP was coded in Java. It had a good result due to the fact that infeasible solutions were corrected to prevent overflow of the bin. This improvement could reduce the computation time but at the cost of reducing the accuracy of the solution.

Jamali and Malektaji [24] modelled the VMP problem using vector packing problem to reduce power consumption by minimising the number of PMs used and also maximising resource usage efficiency. The authors proposed the improved GGA (IGGA) for encoding and generating new solutions regarding the VMP optimisation objective. The proposed algorithm was evaluated using CloudSim and compared with three algorithms: GGA [14], single-objective FFD heuristic, and multi-objective grouping genetic algorithm (MGGA).The results demonstrated that the proposed IGGA algorithm was able to achieve the lowest average power consumption and resource wastage while FFD consumed the highest energy and had the biggest resource wastage.

Sharma and Reddy [25] designed, for the first time, an energy-efficient algorithm to optimise resource allocation in a DC using both dynamic voltage frequency scaling (DVFS) and GA. DVFS changed the frequency of a single PM according to the current workload and then the GA was used for energy-efficient VMs allocation in the DC. The proposed algorithm aimed to reduce both static and dynamic energy requirements. In addition, the mean PM shutdown time at the DC was also minimised by efficiently utilising the PMs. Once all the resources were efficiently consolidated, some PMs might become idle and could be switched off to save more energy. The simulation results based on CloudSim showed that the proposed energy-efficient algorithm consumed 22.4% less energy and increased the average resources utilisation of the DC by 0.6% on specified workloads.

Joseph *et al.* [26] implemented a memory-efficient algorithm using GA for allocating VMs. The objective of the proposed algorithm was to reduce the high runtime and memory requirement of the class of GA solutions to the VM allocation problem. The experimental results obtained from CloudSim showed that the energy consumption decreased by 55%. Overall SLA violation decreased by 90% on average and the runtime was reduced by 73%.

From the literature on VMP algorithms based on GA, it can be noticed that the above algorithms were developed to solve the VMP problem and the optimisation of the existing VMP was not considered. In addition, most algorithms initiated random populations. To improve the quality of the solution for the optimisation technique, local search techniques can be used to generate initial population.

### *B. Ant Colony Optimisation*

The novel approach of ACO was introduced by Dorigo in 1992 in his Ph.D. thesis [27] and was originally called ant system (AS). There are a number of ant algorithms, such as MMAS and ant colony system (ACS). All ACO algorithms share the same idea which is inspired by the foraging behaviour of real ant colonies. While moving from their nest to a food source and back, ants deposit a pheromone on the ground in order to mark some favourable paths that should be followed by other members of the colony. Other ants can smell the pheromone and tend to prefer paths with a higher pheromone concentration. The pseudocode of ACO can be presented in Algorithm 3.

**Algorithm 3** ACO

```
Initialise pheromone trails and parameters.
while termination condition not met do
    while each ant not completes a tour do
        Update local pheromone trail
    end while
    Analyse tours
    Update global pheromone trail
end while
```

The VMP problem was formulated as a multidimensional bin-packing (MDBP) problem which means PMs were bins and VMs represented the objects to be packed. Feller *et al.* [28] designed, for the first time, a novel MMAS metaheuristic based single-objective (i.e., minimise the number of PMs and then the energy consumption would be minimised) algorithm for the consolidation of dynamic VMs in a cloud DC. This algorithm considered multi-resources which are CPU cycles, CPU cores, disk size, RAM size and network bandwidth. The evaluation was done by simulation tools developed in Java and compared with FFD and CPLEX algorithms. The results showed that ACO was more energy efficient than FFD. It conserved 4.1% of energy and reduced the number of PMs used by 4.7%. However, the proposed algorithm was a single-objective algorithm which aimed to minimise the number of PMs and it needed more computation time compared with the FFD algorithm.

Similar to [28], Liu *et al.* [29] proposed a VMP algorithm based on ACO to reduce the number of running PMs. Unlike other works, which deposited pheromones between the PM and the VM, the proposed algorithm deposited pheromone between every two VMs to record the historical desirability of placing them in the same PM. Moreover, the heuristic information was defined between the VM and PM to measure how the resource utilisation ratio could be improved if the VM was placed on this PM. Thus, the heuristic information could further help the proposed algorithm to place VMs on the most suitable PMs. The simulation results demonstrated that the proposed algorithm outperformed the FFD algorithm in reducing the number of active PMs by 14% taking 600 VMs. In contrast to [28] and [29], Gao *et al.* [30] proposed a multi-objective VMP algorithm based on ACS (VMPACS). The aim was to obtain a Pareto set that simultaneously minimised total resource wastage and power consumption in an efficient way. The proposed algorithm was evaluated by comparing with the multi-objective GA (MGGA) [14] and two single-objective algorithms: FFD algorithm [28] and MMAS algorithm.The work considered two types of resources (i.e. CPU and memory). VMPACS algorithm outperformed MGGA and one single-objective ACO algorithms in terms of power and resource wastage.

Similar to [30], Ferdaus *et al.* [31] integrated ACS with balanced resource utilisation of PMs for different resource types (i.e. CPU, network I/O and memory). The proposed algorithm was to minimise energy consumption and resource wastage. Pheromone levels were associated to all VM-to-PM assignments to perform the desirability of assigning a VM to a PM. Heuristic values were computed dynamically for each VM-to-PM assignment to represent the favourability of assigning a VM to a PM in terms of both overall and balanced resource utilisation of the PM. The simulation results based on CloudSim showed that the algorithm reduced power consumption by 2.20%, 5.77%, 11.06% and 11.94% compared with ACO-based workload consolidation algorithm [28], a greedy algorithm, FFDVolume and modified FFD based on L1 norm mean estimator, respectively. However, all previous proposed algorithms were simulated in a homogeneous environment and did not consider QoS.

Reducing energy consumption while maintaining the desired QoS was proposed by Farahnakian *et al.* [32]. A multi-agent system architecture for a VM re-placement was proposed. ACS-based VM consolidation (ACSVMC) approach tried to find a near-optimal solution based on a specified objective function. VMP in ACSVMC was based on three resources: CPU, memory, and network Input/Output (I/O). The proposed algorithm was simulated on CloudSim and compared with the algorithm presented in [31] and modified best fit decreasing (MBFD) [33]. Simulation results on real workload traces showed that ACSVMC outperformed the compared algorithms in reducing energy consumption, the number of VM migrations, and amount of SLA violations.

Resource wastage only was optimised for VMP by Tawfeek *et al.* [34]. The proposed algorithm aimed to simultaneously optimise total CPU and memory resource wastage. To solve the VMP problem, the ACO algorithm was proposed to search the solution space efficiently and obtain a Pareto set. The proposed algorithm was simulated on CloudSim and evaluated by comparing with FFD-CPU, FFD-MEM, BFD-CPU, BFD-MEM algorithms and VMPACS algorithm [30]. The simulation results demonstrated that the proposed algorithm was superior and outperformed the compared algorithms in terms of resource wastage.

Optimising communication traffic in a DC is one of the VMP objective functions. Dong *et al.* [35] proposed a multi-resource VMP algorithm to reduce the total communication traffic in a DC network and optimise network maximum link utilisation (MLU). In the proposed algorithm, the 2-opt local search was combined with ACO to improve search speed and accelerate convergence speed. The proposed algorithm ran on different topologies, such as Tree, VL2 and fat-tree, and was compared with local search (LS) and SA algorithms. The simulation results demonstrated that the proposed algorithm was able to obtain better optimisation results. However, the proposed algorithm focused only on the performance of DC network.

Malekloo and Kara [36] modelled VMP as a multi-objective optimisation to minimise power consumption, resource wastage and energy communication cost between network elements within a DC. ACO algorithm was proposed to obtain a Pareto set to solve the multi-objective problem. The proposed algorithm modified the probabilistic decision rule and heuristic information formula as in [28]. The proposed algorithms were simulated using CloudSim. The performance of the algorithm was compared with three single-objective algorithms (FFD, DVFS, local regression (LR)) and a multi-objective GA (MGA). The simulation results showed that FFD yielded the highest energy consumption due to the sorting mechanism of the VMs to the first available PMs without any attention to the resources available in other PMs. The proposed algorithm yielded the lowest energy consumption due to the randomness of a metaheuristic technique. On average, 39.19% of energy were saved by the proposed algorithm whereas MGA saved energy by almost 22.175%.

### C. Particle Swarm Optimisation

PSO was developed by Kennedy and Eberhart in 1995 [37] and motivated by the social behaviour of particles. Each particle in a swarm represents a feasible solution of the problem. Every particle has two parameters: velocity and position. The position is associated with a fitness value, which is used to evaluate the quality of the solution. The pseudocode of PSO algorithm can be presented in Algorithm 4.

**Algorithm 4** PSO

```
Initialise a population of particles with random values positions and velocities.
while termination condition not met do
  for each particle do
    Calculate fitness value (f) = (nBest)
    if (nBest) is better than the best fitness value (pBest) in history then
      Set (nBest) as the new (pBest)
    end if
    Select the best particle of swarm as (gBest)
    if (nBest) is better than the best fitness value (gBest) in global then
      Set (nBest) as the (gBest)
    end if
  end for
  for all particles do
    Update velocity of the particle
    Update the position of the particle
  end for
end while
return Best particle as the final solution.
```

PSO was implemented to solve an energy-aware VMP optimisation problem in cloud DC by Wang *et al.* [38]. The authors improved the PSO algorithm by redefining the parameters and operators of the PSO, adopting an energy-aware local fitness first strategy to update the particle position and improve the problem-solving efficiency and designing a two-dimensional particle encoding scheme. To evaluate the proposed approach, it was compared with MBFD [33], FF and BF. Experimental results demonstrated that the proposed approach significantly outperformed the other approaches in terms of energy reduction. It could reduce energy consumption by 13-23%.

Besides energy consumption, resource utilisation was also optimised and formulated as the total Euclidean distance to determine the optimal point between resource utilisation and energy consumption as in [30]. Xiong and Xu [39] proposed an energy-aware VMP algorithm, MREE-PSO, based on an energy-efficient multi-resource allocation model and PSO method. The advantage of this algorithm was that it avoided falling into local optima. The proposed algorithm was simulated on CloudSim and its results were compared with MBFD algorithm [33] and the consolidation algorithm [40]. The results showed that the proposed algorithm significantly outperformed the compared algorithms in terms of energy savings and resource utilisation. The total Euclidean distance increased with the increasing number of VMs. However, the total Euclidean distance was lower for MREE-PSO; it also increased more slowly with the increasing number of VMs.

A multi-objective PSO algorithm to place VMs was used by Gao and Tang [41]. The objectives to be met were the minimisation of total resource utilisation of PMs and the number of VM migrations. To validate the algorithm, a comparison was conducted with BFD resource utilisation, single-objective PSO of resource utilisation (PSO-R) and VM migration (PSO-M). Simulation results showed that the proposed algorithm had a VM migration time shorter than the PSO-R and PSO-M algorithms. However, the algorithm did not consider the energy consumption in the cloud DC.

Minimising energy consumption while maintaining the required QoS is one of the main challenges in a cloud DC. Dashti and Rahmani [42] modified the PSO to place migrated VMs from the overloaded PMs and also dynamically consolidate the underloaded PMs to save more energy while maintaining the required QoS. The proposed algorithm was compared with MBFD [33] and two algorithms, FF and BF. Two strategies (a single-threshold and a DVFS) were used and compared in terms of energy consumption, number of VM migrations and total simulation time. Simulation results on CloudSim showed that the proposed algorithm could save about 14% energy and the number of VM migrations and simulation time were reduced.

## V. Non-Reproductive Population-Based Algorithms

### A. *Biogeography-Based Optimisation*

BBO was proposed by Simon [43]. It studies the geographical distribution of species. The habitability (suitability for biological residence) of an island is indicated by its habitat suitability index (HSI), which is determined by a number of independent variables called suitability index variables (SIVs). The higher the HSI of an island, the more the species on the island, the lower its immigration rate, and the higher its emigration rate. The pseudocode of BBO can be presented in Algorithm 5.

The first VMP based on a BBO algorithm was proposed by Ali and Lee [44]. It aimed to minimise the energy consumption. To validate the proposed algorithm, BBO was simulated and compared with GA in terms of energy consumption and Matlab time. The results showed that BBO outperformed GA. However, the proposed algorithm did not consider the QoS, specifically the SLA violation. In addition, it just focused on the fresh VMP.

Zheng *et al.* [45] also proposed a VMP based on a BBO algorithm called VMPMBBO which considered the VMP problem as a complex system. The aim of VMPMBBO was to optimise the VMP in order to simultaneously minimise resource wastage, energy consumption, inter-VM network, storage traffic, VM migration cost and perform load balancing.The evaluation results compared the proposed VMPMBBO with three multi-objective VMP algorithms: MGGA [14], VMPACS [30] and a Pareto-based BF algorithm [46]. The results showed that the BF algorithm yielded the highest costs because it optimised a weighted sum of objectives in the Pareto set for each single VM request, and it tended to achieve the

**Algorithm 5** BBO

Initialise a population of solutions (islands) $x_k$ of size N
Set emigration probability $\mu_k$
Set immigration probability $\lambda_k$
**while** termination condition not met **do**
  **for** each solution $x_k$ **do**
    Set $z_k$ is a temporary population
    $z_k \leftarrow x_k$
  **end for**
  **for** each individual $z_k$ **do**
    **for** each independent variable index $s$ **do**
      Use $\lambda_k$ to probabilistically decide whether to immigrate to $z_k$
      **if** immigrating **then**
        Use $\mu_i$ to probabilistically select the emigrating individual $x_j$
        $z_k(s) \leftarrow x_j(s)$
      **end if**
    **end for**
    Probabilistically mutate $z_k$
  **end for**
  $x_k \leftarrow z_k$
**end while**

locally optimal solution. The other three algorithms produced the lowest costs because they were able to search the solution space more efficiently and globally.

To improve the VMPMBBO [45], Zheng *et al.* [47] conducted extensive experiments using synthetic data, where VMPMBBO was compared with two multi-objective optimisation algorithms: MGGA [14] and VMPACS [30]. It was shown that VMPMBBO had better convergence characteristics and was more computationally efficient as well as robust. However, both algorithms did not consider the QoS in their objective functions, specifically the violation in the SLA.

## VI. Discussion

An overview of the optimisation techniques from the 31 reviewed articles is illustrated in Table I.

Most of the papers consider minimising energy consumption while maintaining the QoS in VMP at three different levels: (1) assign the workload of application to the existing VMs, (2) place VMs to PMs, and (3) re-place the VMs to other PMs due to the dynamic workloads. Researchers often addressed and evaluated the three levels individually, although Xu *et al.* [14] addressed the first and second levels, and Khalilzad *et al.* [11] considered all three levels. A generalised framework for the three levels should be considered to obtain better results.

Implementing hybrid metaheuristic algorithms may get benefits from both algorithms; the limitations of one algorithm can be overcome by the advantages of the other algorithm. Hybrid metaheuristic algorithms can improve the quality of the solution or convergence speed of metaheuristic algorithms. However, it can be noticed from the literature that hybrid metaheuristic algorithms have rarely been used in the VMP context.

TABLE I: VMP techniques for VMP.

| Optimisation Technique | Objective Function | Ref |
|---|---|---|
| SA | Energy | [10] [11] |
| | Energy & VMs migration | [12] |
| GGA | Energy & resource utilisation | [14] [19] [24] |
| HGA | Energy & revenue | [20] |
| | Energy | [17] |
| GA | Energy & QoS | [15] |
| | Energy | [16] [18] [23] [25] |
| | Resource utilisation & cost | [22] |
| | Energy & resource utilisation | [26] |
| | Energy & cost | [21] |
| ACO | Energy | [28] [29] |
| | Energy & resource utilisation | [30] [31] [36] |
| | Resource utilisation | [34] |
| | Energy & QoS | [32] |
| ACO & 2-opt | QoS | [35] |
| PSO | Resource utilisation CPU & RAM | [41] |
| | Energy | [38] [39] |
| | Energy & QoS | [42] |
| BBO | Energy | [44] |
| | Energy & resource utilisation | [45] [47] |

Most of the papers reviewed here consider the scalability of the proposed algorithms. The number of PMs and VMs was changed in each experiment to check whether the algorithm was scalable or not [12], [14], [15], [16], [17], [20], [30], [31], [34], [41] and [45]. Moreover, the robustness of the algorithm was validated by changing the initial solution size and the number of generations in [14] and [45].

However, the convergence of algorithms should also be taken into account when validating the proposed algorithms.

## VII. Conclusion

This paper has reviewed metaheuristic techniques for VMP in cloud computing. The analysis of VMP algorithms compared the optimisation techniques (SA, GA, ACO, PSO and BBO) and objective functions (energy consumption, QoS, cost, revenue, VMs migration and resource utilisation).

Regarding the objective functions in VMP in the literature, most of the authors have focused on minimising the energy consumption of DCs. Some authors have also addressed issues related to performance and resource utilisation. The main challenge is to reduce energy consumption of DCs without degrading performance or violating SLA constraints.

## References


[1] N. Bobroff, A. Kochut, and K. Beaty, "Dynamic placement of virtual machines for managing sla violations," in *Integrated Network Management, 10th IFIP/IEEE Int. Symp. on*, pp. 119–128, IEEE, 2007.

[2] A. Beloglazov, J. Abawajy, and R. Buyya, "Energy-aware resource allocation heuristics for efficient management of data centers for Cloud computing," *Future Generation Computer Sys.*, vol. 28, no. 5, pp. 755–768, 2012.

[3] A. Hameed, A. Khoshkbarforoushha, R. Ranjan, P. P. Jayaraman, J. Kolodziej, P. Balaji, S. Zeadally, Q. M. Malluhi, N. Tziritas, A. Vishnu, S. U. Khan, and A. Zomaya, "A survey and taxonomy on energy efficient resource allocation techniques for cloud computing systems," *Computing*, 2014.

[4] L. Salimian and F. Safi, "Survey of energy efficient data centers in cloud computing," in *Proc. of IEEE/ACM 6th Int. Conf. on Utility and Cloud Computing*, pp. 369–374, IEEE Computer Society, 2013.

[5] R. Ranjana and J. Raja, "A survey on power aware virtual machine placement strategies in a cloud data center," *Intl. Conf. on Green Computing, Communication and Conservation of Energy*, pp. 747–752, 2013.

[6] F. Lopez-Pires and B. Baran, "Virtual machine placement literature review," *arXiv preprint arXiv:1506.01509*, 2015.

[7] A. Varasteh and M. Goudarzi, "Server consolidation techniques in virtualized data centers: A survey," 2015.

[8] Z. Á. Mann, "Allocation of virtual machines in cloud data centersa survey of problem models and optimization algorithms," *ACM Computing Surveys*, vol. 48, no. 1, p. 11, 2015.

[9] S. Kirkpatrick, M. P. Vecchi, *et al.*, "Optimization by simmulated annealing," *science*, vol. 220, no. 4598, pp. 671–680, 1983.

[10] Y. Wu, M. Tang, and W. Fraser, "A simulated annealing algorithm for energy efficient virtual machine placement," *IEEE Intl. Conf. on Sys., Man, and Cybernetics*, pp. 1245–1250, 2012.

[11] N. Khalilzad, H. R. Faragardi, and T. Nolte, "Towards energy-aware placement of real-time virtual machines in a cloud data center," in *High Performance Computing and Communications, IEEE 7th Int. Symp. on CSS, IEEE 12th Int. Conf. on ICESS, IEEE 17th Int. Conf. on*, pp. 1657–1662, IEEE, 2015.

[12] A. Marotta and S. Avallone, "A simulated annealing based approach for power efficient virtual machines consolidation," in *Cloud Computing, IEEE 8th Int. Conf. on*, pp. 445–452, IEEE, 2015.

[13] J. H. Holland, "Genetic algorithms and the optimal allocation of trials," *SIAM Journal on Computing*, vol. 2, no. 2, pp. 88–105, 1973.

[14] J. Xu and J. A. Fortes, "Multi-objective virtual machine placement in virtualized data center environments," in *Green Computing and Communications, IEEE/ACM Int. Conf. on & Int. Conf. on Cyber, Physical and Social Computing*, pp. 179–188, IEEE, 2010.

[15] D. Jiang, P. Huang, P. Lin, and J. Jiang, "Energy efficient VM placement heuristic algorithms comparison for cloud with multidimensional resources," *Lecture Notes in Computer Science*, vol. 7473 LNCS, pp. 413–420, 2012.

[16] G. Wu, M. Tang, Y.-C. Tian, and W. Li, "Energy-efficient virtual machine placement in data centers by genetic algorithm," in *Neural Information Processing*, pp. 315–323, Springer, 2012.

[17] M. Tang and S. Pan, "A Hybrid Genetic Algorithm for the Energy-Efficient Virtual Machine Placement Problem in Data Centers," *Neural Process. Lett.*, vol. 41, no. 2, pp. 211–221, 2015.

[18] T. Yang, Y. C. Lee, and A. Y. Zomaya, "Energy-Efficient Data Center Networks Planning with Virtual Machine Placement and Traffic Configuration," *IEEE 6th Intl. Conf. on Cloud Computing Tech. and Science*, pp. 284–291, 2014.

[19] C. Liu, C. Shen, S. Li, and S. Wang, "A new evolutionary multi-objective algorithm to virtual machine placement in virtualized data center," in *Software Eng. and Service Science, 5th IEEE Int. Conf. on*, pp. 272–275, IEEE, 2014.

[20] F. L. Pires and B. Barán, "Multi-objective virtual machine placement with service level agreement: A memetic algorithm approach," in *Proc. of IEEE/ACM 6th Int. Conf. on Utility and Cloud Computing*, pp. 203–210, IEEE Computer Society, 2013.

[21] J. a. Pascual, T. Lorido-Botran, J. Miguel-Alonso, and J. a. Lozano, "Towards a Greener Cloud Infrastructure Management using Optimized Placement Policies," *Journal of Grid Computing, Special Issue*, 2014.

[22] A. C. Adamuthe, R. M. Pandharpatte, and G. T. Thampi, "Multiobjective Virtual Machine Placement in Cloud Environment," *Intl. Conf. on Cloud & Ubiquitous Computing & Emerging Tech.*, pp. 8–13, 2013.

[23] M. A. Kaaouache and S. Bouamama, "Solving bin Packing Problem with a Hybrid Genetic Algorithm for VM Placement in Cloud," *Procedia Computer Science*, vol. 60, pp. 1061–1069, 2015.

[24] S. Jamali and S. Malektaji, "Improving grouping genetic algorithm for virtual machine placement in cloud data centers," *4th Intl. Conf. on Computer and Knowledge Eng.*, pp. 328–333, 2014.

[25] N. K. Sharma and G. Reddy, "Novel energy efficient virtual machine allocation at data center using genetic algorithm," in *Signal Processing, Communication and Networking, 3rd Int. Conf. on*, pp. 1–6, IEEE, 2015.

[26] C. T. Joseph, K. Chandrasekaran, and R. Cyriac, "Improving the efficiency of genetic algorithm approach to virtual machine allocation," in *Computer and Communication Tech., Int. Conf. on*, pp. 111–116, IEEE, 2014.

[27] M. Dorigo, V. Maniezzo, and A. Colorni, "Ant system: optimization by a colony of cooperating agents," *Sys., Man, and Cybernetics, Part B: Cybernetics, IEEE Trans. on*, vol. 26, no. 1, pp. 29–41, 1996.

[28] E. Feller, L. Rilling, and C. Morin, "Energy-Aware Ant Colony Based Workload Placement in Clouds," *Grid Computing, 12th IEEE/ACM Intl. Conf. on*, pp. 26–33, 2011.

[29] X.-F. Liu, Z.-H. Zhan, K.-J. Du, and W.-N. Chen, "Energy aware virtual machine placement scheduling in cloud computing based on ant colony optimization approach," in *Proc. of the conf. on Genetic and evolutionary computation*, pp. 41–48, ACM, 2014.

[30] Y. Gao, H. Guan, Z. Qi, Y. Hou, and L. Liu, "A multi-objective ant colony system algorithm for virtual machine placement in cloud computing," *Journal of Computer and Sys. Sciences*, vol. 79, no. 8, pp. 1230–1242, 2013.

[31] M. H. Ferdaus, M. Murshed, R. N. Calheiros, and R. Buyya, "Virtual machine consolidation in cloud data centers using aco metaheuristic," in *Euro-Par Parallel Processing*, pp. 306–317, Springer, 2014.

[32] F. Farahnakian, A. Ashraf, and T. Pahikkala, "Using Ant Colony System to Consolidate VMs for Green Cloud Computing," *IEEE Trans. on Services Computing*, vol. 8, no. 2, pp. 187–198, 2015.

[33] A. Beloglazov and R. Buyya, "Optimal online deterministic algorithms and adaptive heuristics for energy and performance efficient dynamic consolidation of virtual machines in Cloud data centers," *Concurrency Computation Practice and Experience*, vol. 22, no. 6, pp. 1397–1420, 2011.

[34] M. a. Tawfeek, a. B. El-Sisi, a. E. Keshk, and F. a. Torkey, "Virtual machine placement based on ant colony optimization for minimizing resource wastage," *Communications in Computer and Information Science*, vol. 488, pp. 153–164, 2014.

[35] J. K. Dong, H. B. Wang, Y. Y. Li, and S. D. Cheng, "Virtual machine placement optimizing to improve network performance in cloud data centers," *Journal of China Universities of Posts and Tele.*, vol. 21, no. 3, pp. 62–70, 2014.

[36] M. Malekloo and N. Kara, "Multi-objective ACO virtual machine placement in cloud computing environments," *IEEE Globecom Workshops*, pp. 112–116, 2014.

[37] J. Kennedy and R. E. P. S. Optimization, "Ieee int," in *Conf. on Neural Networks*, vol. 4, 1995.

[38] S. Wang, Z. Liu, Z. Zheng, Q. Sun, and F. Yang, "Particle swarm optimization for energy-aware virtual machine placement optimization in virtualized data centers," *Intl. Conf. on Parallel and Dist. Sys.*, pp. 102–109, 2013.

[39] A.-p. Xiong and C.-x. Xu, "Energy Efficient Multiresource Allocation of Virtual Machine Based on PSO in Cloud Data Center," *Mathematical Problems in Eng.*, vol. 2014, pp. 1–8, 2014.

[40] S. Srikantaiah, A. Kansal, and F. Zhao, "Energy aware consolidation for cloud computing," in *Proc. of the Conf. on Power aware computing and Sys.*, vol. 10, San Diego, California, 2008.

[41] J. Gao and G. Tang, "Virtual Machine Placement Strategy Research," *Intl. Conf. on Cyber-Enabled Dist. Computing and Knowledge Discovery*, no. 2, pp. 294–297, 2013.

[42] S. E. Dashti and A. M. Rahmani, "Dynamic VMs placement for energy efficiency by PSO in cloud computing," *Journal of Experimental & Theoretical Artificial Intelligence*, vol. 0, no. 0, pp. 1–16, 2015.

[43] D. Simon, "Biogeography-based optimization," *Evolutionary Computation, IEEE Trans. on*, vol. 12, no. 6, pp. 702–713, 2008.

[44] H. M. Ali and D. C. Lee, "A biogeography-based optimization algorithm for energy efficient virtual machine placement," *IEEE Symp. on Swarm Intelligence*, pp. 1–6, 2014.

[45] Q. Zheng, R. Li, X. Li, and J. Wu, "A Multi-objective Biogeography-Based Optimization for Virtual Machine Placement," *15th IEEE/ACM Intl. Symp. on Cluster, Cloud and Grid Computing*, pp. 687–696, 2015.

[46] T. Glatard, M.-E. Rousseau, P. Rioux, R. Adalat, and A. Evans, "Controlling the deployment of virtual machines on clusters and clouds for scientific computing in cbrain," in *Cluster, Cloud and Grid Computing, 14th IEEE/ACM Int. Symp. on*, pp. 384–393, May 2014.

[47] Q. Zheng, R. Li, X. Li, N. Shah, J. Zhang, F. Tian, K.-M. Chao, and J. Li, "Virtual machine consolidated placement based on multi-objective biogeography-based optimization," *Future Generation Computer Sys.*, 2015.