

基于分组遗传算法的数据中心虚拟机节能映射

吴小东^{1,2,3,4}, 王荣海^{1,2,3,4}, 林国新^{2,3,4}

- 泉州师范学院 数学与计算机科学学院, 福建 泉州 362000
- 福建省大数据管理新技术与知识工程重点实验室, 福建 泉州 362000
- 智能计算与信息处理福建省高等学校重点实验室, 福建 泉州 362000
- 数字福建智能制造大数据研究所, 福建 泉州 362000

摘要:近年来,随着人们对云计算业务需求持续增长,数据中心能耗日益增加,由此不仅增加了运营成本,巨大的碳排放对生态环境也产生严重的影响,数据中心节能已成为当前亟须解决的重要难题。对云数据中心的虚拟机放置(Virtual Machine Placement, VMP)进行优化能有效地提高资源利用率,同时,VMP也是减少数据中心能耗的重要技术之一;针对数据中心的能耗感知VMP问题,提出一种基于分组遗传算法的节能算法EEGGA(Energy-efficient Grouping Genetic Algorithm),算法将节能VMP问题视为装箱问题(Bin Packing Problem, BPP),并应用基于分组编码的遗传算法对其进行求解,通过减少活动物理主机的数量(装箱数量)以实现降低数据中心能耗的目标;在算法迭代过程的交叉和变异等阶段,设计了多种启发优化策略提升子代染色体的适应度,从而提高算法的节能性能和加快迭代收敛的速度;通过仿真实验,在收敛速度和求解性能等方面将提出的算法与传统的节能遗传算法进行对比,实验结果表明:提出的算法能够有效地减少数据中心的能耗,在节能性能和求解收敛速度方面均优于其他算法。

关键词:虚拟机放置;节能;分组遗传算法;装箱问题;数据中心

中图分类号:TP312 **文献标识码:**A **doi:**10.16055/j.issn.1672-058X.2024.0004.012

An Energy-efficient Virtual Machine Mapping in Data Centers Based on Grouping Genetic Algorithm

WU Xiaodong^{1,2,3,4}, WANG Ronghai^{1,2,3,4}, LIN Guoxin^{2,3,4}

- School of Mathematics and Computer Science, Quanzhou Normal University, Fujian Quanzhou 362000, China
- Fujian Provincial Key Laboratory of Data Intensive Computing, Fujian Quanzhou 362000, China
- Key Laboratory of Intelligent Computing and Information Processing (Fuzhou Province University), Fujian Quanzhou 362000, China
- Fujian Provincial Big Data Research Institute of Intelligent Manufacturing, Fujian Quanzhou 362000, China

Abstract: In recent years, with the continuous growth of the demand for cloud computing, the energy consumption of cloud data centers has been increasing, which not only brings economic problems, but also has a serious impact on the ecological environment caused by huge carbon emissions. Therefore, data center energy saving has become an important problem to be solved urgently. In cloud data centers, Virtual Machine Placement (VMP) optimization can effectively improve resource utilization. VMP is also one of the important technologies to reduce the energy consumption of data centers. Aiming at the problem of energy-aware VMP of cloud data centers, an Energy-efficient Grouping Genetic Algorithm (EEGGA) based on grouping genetic algorithm (GA) was proposed. The algorithm considered the energy-

收稿日期:2023-03-16 **修回日期:**2023-05-18 **文章编号:**1672-058X(2024)04-0097-07

基金项目:福建省自然科学基金项目(2020J01784);福建省教育厅中青年教师教育科研项目(JAT220280).

作者简介:吴小东(1979—),男,江西赣州人,博士,副教授,从事并行与分布式计算、节能算法等研究.

引用格式:吴小东,王荣海,林国新.基于分组遗传算法的数据中心虚拟机节能映射[J].重庆工商大学学报(自然科学版),2024,41(4):97-103.

WU Xiaodong, WANG Ronghai, LIN Guoxin. An energy-efficient virtual machine mapping in data centers based on grouping genetic algorithm[J]. Journal of Chongqing Technology and Business University (Natural Science Edition), 2024, 41(4): 97-103.

efficient VMP problem as a bin packing problem (BPP) and applied a genetic algorithm based on grouping coding to solve it. The goal of reducing the overall energy consumption of data centers was achieved by reducing the number of active physical hosts (the number of containers). At the same time, several heuristic optimization strategies were designed to improve the fitness of offspring chromosomes in the crossover and mutation stages of the iteration process, so as to improve the energy-saving performance of the algorithm and accelerate the speed of convergence. In the simulation experiments, the proposed algorithm was compared with traditional algorithms in the aspects of convergence speed and energy-saving performance. Experimental results show that the proposed algorithm can effectively reduce energy consumption and is superior to other algorithms in terms of energy-saving performance and convergence speed.

Keywords: virtual machine placement; energy saving; grouping genetic algorithm; bin packing problem; data centers

1 引言

近年来,随着人们对云计算业务的需求持续增长,云数据中心的建设规模不断扩大,由此带来的能量消耗也急剧上升。据估计,美国数据中心每年需要消耗 1 350 亿千瓦时的能量^[1],而全球数据中心每年的能源使用量则估计在 200~500 TWh 之间^[2]。云数据中心如此巨大的能量消耗不仅是一个重要的经济问题,同时还带来大量的碳排放,而碳减排的压力^[3]也是一个生态环境问题。如何降低数据中心的能耗已成为当前的研究热点。

虚拟机放置(Virtual Machine Placement, VMP)是云数据中心对资源进行分配和管理的重要方式,VMP 根据一定的方法和策略,将各个虚拟机(Virtual Machine, VM)映射放置到特定的物理机(Physical Machine, PM)上运行,以实现优化资源分配的目标。对虚拟机的优化放置能够有效地提高云计算数据中心的资源利用效率以及降低数据中心能量消耗。

目前,已经有一些针对云计算数据中心进行 VMP 优化的相关研究。例如,文献[4]证明了 VMP 优化是一个 NP-hard 问题,在多项式时间内无法找到最优解,即其求解时间通常会随着问题规模的增大而急剧增加。这就意味着在问题规模较大的情况下,使用有限的计算资源在可接受的短时间内得到 VMP 优化的最优解是不现实的^[5],这也使得 VMP 优化仍是一个尚未完全解决的重要挑战。因此,现有的相关研究成果大多提出采用智能启发式算法和仿生优化算法等近似优化算法,通过获取近似最优解的方式来解决 VMP 的优化问题。例如,文献[6]提出一种改进的遗传算法,通过结合虚拟层次结构模型和遗传算法来进行虚拟机放置,从而优化云数据中心的可用性和能耗。文献[7]提出一种基于二进制引力搜索算法的 VMP 优化方案以减少异构数据中心的能量消耗。文献[8]指出 VMP 是一个 NP-hard 问题,即使对于小型数据中心也无法找到最佳的解决方案,并提出一种混合遗传算法,通过最小化托管虚拟机的活动服务器数量来提高云数据中心的能效。文献[9]提出一种基于灰狼优化(GWO, Grey

Wolf Optimization)技术的算法来减少活动主机的数量以提高硬件的利用率和减少能量消耗。文献[10]提出一种基于花朵授粉优化算法(Flower Pollination Algorithm)的节能算法来解决数据中心复杂的 VMP 问题,花朵授粉优化算法是一种受自然启发的优化技术,用于数值优化问题。

综上,对于 VMP 的节能优化问题,目前已经取得一些相关研究成果,这些研究普遍采用启发类方法求取其近优解。然而,启发式优化算法大多存在进化动力不足、收敛速度较慢等缺陷^[11],而现有的多数相关研究在对能耗进行优化的同时往往缺乏对算法进化和收敛速度方面的考虑,而对于云数据中心的运营而言,在短时间内对虚拟机的放置作出决策非常重要,尤其是虚拟机在线放置的情景^[12]。因此,VMP 节能优化算法在进行能耗优化的时候,有必要同时考虑提高其求解速度、减少求解时间。

针对此问题,提出了一种基于分组遗传算法(Grouping Genetic Algorithm, GGA)的虚拟机节能放置算法。提出的算法采用分组的方式对染色体进行编码,通过在交叉、变异等阶段的遗传算子中设计多种启发优化策略,以实现在优化 VMP 节能性能、降低数据中心能量消耗的同时,加快遗传算法的收敛速度、减少算法迭代求解的执行时间。实验结果表明,与同类算法相比,提出的算法具有更快的收敛速度,并能取得更好的节能性能。

2 问题定义

现将给出所研究的数据中心系统模型及其能耗消耗模型,然后给出 VMP 节能优化问题的形式化描述。

2.1 系统模型

假设数据中心包含 n 个物理主机,表示为: $P = \{p_1, p_2, \dots, p_n\}$,主机的计算能力用 MIPS (Millions Instructions Per Second)表示。简单起见,假设所有的物理主机具有相同的计算能力,记为 p^{cap_mips} 。待映射放置到物理机的虚拟机一共有 m 个,表示为 $V = \{v_1, v_2, \dots, v_m\}$,第 i 个虚拟机 v_i 对计算力的需求量表示为 v_i^{mips} 。

虚拟机到物理机的映射关系用二进制变量 $y_{i,j}$ 表示,若将虚拟机 v_i 映射放置到物理机 p_j 上,则令 $y_{i,j}=1$,否则 $y_{i,j}=0$ 。因此,放置在主机 p_j 上的所有虚拟机的计算需求总量 $p_j^{req_mips}$ 可计算为

$$p_j^{req_mips} = \sum_{i=1}^m (v_i^{mips} \times y_{i,j}) \quad (1)$$

进而可得到物理主机 p_j 的 CPU 利用率 u_j 为

$$u_j = \frac{p_j^{req_mips}}{p^{cap_mips}} \quad (2)$$

2.2 能耗模型

研究采用基于利用率的能耗模型。基于系统利用率的模型是当前数据中心能耗研究中应用最广的能耗模型之一^[13]。根据此模型,物理主机的能量消耗与其 CPU 的利用率之间可以表现为线性的关系^[14-15]。当物理主机 p_j 以利用率 u_j 运行时,其能耗可以表示为^[15]

$$E(p_j) = k \times e^{\max} + (1-k) \times e^{\max} \times u_j \quad (3)$$

其中, e^{\max} 为物理主机以最高利用率运行时的最大能耗; k 是一个正常数,表示主机处于空闲状态的能耗占其最大能耗 e^{\max} 的比例^[15]。无负载的空闲物理机可切换至休眠模式以节省能耗^[16]。

2.3 虚拟机节能放置问题描述

假设 M 为一个虚拟机放置方案,依照方案 M 进行虚拟机映射所需要的物理机(PM)数量表示为 n_M 。不失一般性,假设方案 M 所使用的 n_M 个 PM 为 $\{p_1, p_2, \dots, p_{n_M}\}$ 。那么, VMP 节能优化问题的目标即是寻找一个映射方案 M ,使得总能耗:

$$E(M) = \sum_{j=1}^{n_M} E(p_j) \quad (4)$$

最小,并且同时满足以下约束条件式(5)和式(6):

$$\sum_{j=1}^n y_{i,j} = 1, \forall v_i \in V, 1 \leq i \leq m \quad (5)$$

$$\sum_{i=1}^m (v_i^{mips} \times y_{i,j}) \leq p^{cap_mips}, \forall p_j \in P, 1 \leq j \leq n \quad (6)$$

其中,式(5)限制每个虚拟机必须且仅能放置到一个物理机上;式(6)确保映射到任何一个物理机上的所有虚拟机所需的 MIPS 总量不超过此物理机的最大计算能力。

从能耗计算式(3)可以看出,在虚拟机到物理机的映射方案中,只要一个物理主机处于活动状态,即使其利用率很低甚至接近空负载,也至少需要消耗其空闲能量部分($k \times e^{\max}$)。因此,减少被使用的活动物理主机的数量有助于降低系统能耗。

命题1 若物理主机能耗与 CPU 利用率呈线性关系,则在虚拟机放置方案中,减少物理主机的使用数量可以降低数据中心的能耗。

证明: 假定有 VMP 映射方案 M_1 和 M_2 , 两个方案

所需使用物理主机的数量分别表示为 n_{M_1} 和 n_{M_2} 。不失一般性,假设方案 M_2 比方案 M_1 使用了更多的物理主机,即 $n_{M_2} > n_{M_1}$ 。

根据式(2)、式(3)、式(4),可得方案 M_2 所需的能量消耗为

$$\begin{aligned} E(M_2) &= \sum_{j=1}^{n_{M_2}} E(p_j) = \\ &= \sum_{j=1}^{n_{M_2}} (k \times e^{\max} + (1-k) \times e^{\max} \times u_j) = \\ &= \sum_{j=1}^{n_{M_2}} \left(k \times e^{\max} + (1-k) \times e^{\max} \times \frac{p_j^{req_mips}}{p^{cap_mips}} \right) = \\ &= n_{M_2} \times k \times e^{\max} + (1-k) \times e^{\max} \times \frac{\sum_{j=1}^{n_{M_2}} p_j^{req_mips}}{p^{cap_mips}} \end{aligned}$$

由于方案 M_2 将 m 个虚拟机放置到 n_{M_2} 台物理主机上,所以 n_{M_2} 台物理主机的 CPU 计算量需求之和亦即是所有 m 个虚拟机的 CPU 请求量之和,即:

$$\sum_{j=1}^{n_{M_2}} p_j^{req_mips} = \sum_{i=1}^m v_i^{mips}$$

因此,方案 M_2 的能量消耗可以表示为

$$E(M_2) = n_{M_2} \times k \times e^{\max} + (1-k) \times e^{\max} \times \frac{\sum_{i=1}^m v_i^{mips}}{p^{cap_mips}} \quad (7)$$

同理,可以得到方案 M_1 的能量消耗:

$$E(M_1) = n_{M_1} \times k \times e^{\max} + (1-k) \times e^{\max} \times \frac{\sum_{i=1}^m v_i^{mips}}{p^{cap_mips}} \quad (8)$$

综合式(7)和式(8),可以得到:

$$E(M_2) = E(M_1) + (n_{M_2} - n_{M_1}) \times k \times e^{\max}$$

由于 $n_{M_2} > n_{M_1}$, $k > 0$, $e^{\max} > 0$, 因而有:

$$E(M_2) > E(M_1) \quad (9)$$

也就是说,映射方案 M_1 比方案 M_2 消耗更少的能量。

因此,在主机的能耗和利用率之间具有线性关系的前提下,通过减少 VMP 映射方案中物理主机的使用数量能够降低系统总能耗。

由命题1, VMP 节能优化问题可以看作装箱问题(Bin Packing Problem, BPP),即将一组已知大小的物品放入一些容量相同的箱子中,在满足每个箱子中物品大小之和不超过箱子容量的前提下,使装箱的数量最少。

研究将数据中心的 VMP 节能优化问题视为装箱问题,并提出一种基于分组遗传算法的 EEGGA (Energy-Efficient Grouping Genetic Algorithm) 节能算法

来求解 VPM 映射方案/BPP 装箱方案,通过减少 VPM 使用的物理主机数量/装箱数量,以达到降低系统能耗的目的;同时,在遗传算子中设计启发优化策略,使算法在实现降低能耗的同时能够加快其收敛和求解的速度。

3 基于分组遗传算法的虚拟机放置

现详细介绍提出的 EEGGA 节能算法,包括种群中染色体的编码方法、个体适应度的计算公式以及遗传迭代过程中所采用的选择算子、交叉算子和变异算子。

3.1 编码方案

研究采用遗传算法求解节能 VPM 问题/装箱问题,为了提高编码表示效率和避免不可行编码,提出的算法采用基于分组的方式并结合启发方法进行染色体编码。

由于 VPM 优化/BPP 是基于分组的问题,如果采用遗传算法传统的编码方式对染色体进行编码,将会存在编码冗余和对遗传算子上下文不敏感的问题,其性能往往表现相对较弱^[17-18]。为了避免此问题,文献[18]提出了分组遗传算法(GGA),以分组的方式对染色体进行编码。

依照基于分组的方式进行编码,每条染色体分两个部分以表示一个装箱方案:一是物品部分,二是箱子部分。物品部分的编码长度固定为 m ,即物品的数量;箱子部分的编码长度为装箱的数量。由于不同的装箱方案所需要使用箱子数量可能不同,即装箱部分对应的编码长度并不固定,因而此编码是一种变长编码方案。

例如,图 1(a)所示的染色体编码表示的是将 8 个物品映射到 4 个箱子的一个方案。物品编码的长度为 8,分别表示 8 个物品对应的装箱编号;箱子的编码部分表示使用了 4 个箱子:A、B、C 和 D。此染色体表示的装箱方案如图 1(b)所示。

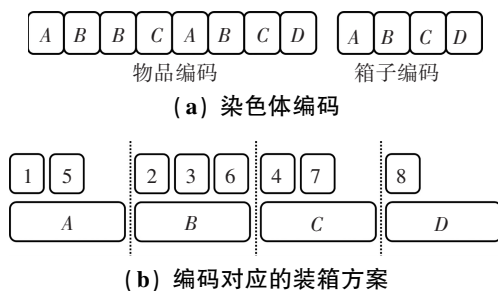


图 1 基于分组的染色体编码方案

Fig. 1 Group-based chromosome coding scheme

另外,由于此种染色体编码方法未对箱子的剩余容量进行判断,因而无法保证分配到各个箱子中的物品总量不超过其容量,因此,直接依此方法对种群中的染色体进行随机编码可导致种群出现不可行的个体^[18],这不但将增加算法求解的计算量,也会影响整体的优化效果。为避免这种情况,通过结合 First-Fit、Next-Fit、Best-Fit 等启发式算法^[19]生成染色体可确保

得到可行的装箱方案。在上述例子中,假设箱子的容量为 100,物品的放置顺序为:1→2→3→4→5→6→7→8,物品大小分别为 68、36、33、49、23、17、30 和 25。若采用 First-Fit 启发式方法对物品按序装箱,即可得到如图 1(a)所示的编码方案,8 个物品分别放置/编码到编号为 A、B、B、C、A、B、C 和 D 的箱子/基因中,4 个箱子所分配的物品分别是 {1,5}、{2,3,6}、{4,7} 和 {8},如图 1(b)所示。由于采用了启发策略辅助编码,因此不会出现超载装箱的情况。

在种群初始化阶段,提出的 EEGGA 算法首先为每个个体都分配一个随机的物品装箱序列,然后,依照此顺序,采用 Next-Fit 启发方法对物品进行装箱操作,并生成对应的基于分组的染色体编码,以确保种群中不会出现不可行的映射方案。

3.2 适应度函数

假设 x 是种群中的一个个体, M_x 表示 x 对应的 VPM 装箱/映射方案,则个体 x 的适应度值根据式(10)计算:

$$f(x) = \frac{\sum_{i=1}^{n_{M_x}} u_i^2}{n_{M_x}} \quad (10)$$

其中, n_{M_x} 是方案 M_x 所需的装箱数量, u_i 是放置到第 i 个箱子中的物品总和与箱子容量的比值,即对应的物理主机的利用率,可通过式(2)计算。

3.3 选择算子

为了提高种群的整体适应性,在 EEGGA 算法迭代的选择阶段,采用了序列选择算子^[20]来选择精英个体。即先对种群中的所有个体根据其适应度大小从高到低的排序,然后优先选择排在前面的适应度较高的个体进入下一代。

同时,算法在其遗传迭代的过程中采用了最佳个体保留策略。如果下一代最佳个体的适应度低于上一代最佳个体的适应度,则用上一代的最佳个体(适应度最高的个体)取代下一代最差的个体(适应度最低的个体),以确保算法在迭代过程中不会出现最优解“退步”的现象。

3.4 交叉算子

在交叉阶段,EEGGA 算法借鉴并改进了文献[18]的交叉方式,设计策略对交叉算子在交叉基因的选取和交叉基因数量的确定两个方面进行了优化。首先,用轮盘赌方法从种群选择两条父染色体,交叉基因的数量由两条父染色体的长度和一个可调节的交叉因子参数 α 确定;交叉算法然后对两条父染色体的基因从大到小进行排序,优先从父染色体中选择较大的基因进行交叉操作,以使子代能获得父代的优质基因、提高子代的适应度。交叉算子的操作步骤如以下算法

所示。

算法 染色体交叉算法

输入:两个父代染色体 x_1 和 x_2 ,交叉因子 α ;

输出:子代染色体 child;

- (1) $l_1 = x_1. genes. length; l_2 = x_2. genes. length;$
- (2) $len = \min((l_1 + l_2) / 2 \times \alpha, \min(l_1, l_2));$
//计算交叉基因数量 len
- (3) $sort(x_1. genes); sort(x_2. genes);$
//将 x_1 和 x_2 的基因按从大到小顺序排序;
- (4) $w_1 = \sum_{i=1}^{len} x_1. genes[i]. weight;$
 $w_2 = \sum_{i=1}^{len} x_2. genes[i]. weight;$
- (5) If ($w_1 > w_2$)
- (6) $x_1 \leftrightarrow x_2;$ //交换染色体 x_1 和 x_2
- (7) Endif
- (8) $child = x_1. clone;$ //从 x_1 克隆子代原型
- (9) $freeItemSet = \varphi;$ //初始化重装物品集合
- (10) For $i = 1$ to len do
//将 x_2 最大的 len 个基因交叉插入到子代
child,并处理子代重复出现的物品
- (11) $tmpGene = x_2. genes[i];$
- (12) Foreach gene in child. genes do
- (13) If $tmpGene \cap gene \neq \varphi$
- (14) $freeItemSet. add(gene - tmpGene);$
- (15) $child. remove(gene);$
- (16) Endif
- (17) EndFor
- (18) $child. add(tmpGene);$
- (19) EndFor
- (20) 将 freeItemSet 中的物品以 FFD 策略重新加入
child 中;
- (21) 返回 child。

首先,算法在第 1~2 行根据两个父染色体的长度和交叉因子 α 确定交叉基因的数量 len ;接着,算法第 3—8 行从两条父染色体中选择前 len 个基因总和较小的父染色体 x_1 ,并以此作为模板生成子代染色体 child,其中第 3 行排序所需的时间为 $O(n) * \log(n)$,第 4 行求和所需时间为 $O(n)$;然后,算法第 9~19 行对 child 进行交叉和调整,从父染色体 x_2 中依次选择最大的 len 个基因(即较满载的箱子)插入到子代 child,若将交叉基因数 len 视为常数,则第 10~19 行需要时间为 $O(m)$;算法第 20 行对于 freeItemSet 集合中那些由于新基因的插入而在子代染色体中重复出现的物品,采用 FFD(First-Fit Decreasing) 策略重新装箱,由于 FFD 需要对物品排序,故 20 行所需的时间为 $O(m) * \log(m)$;最后,算法 21 行返回交叉操作的子代染色体 child。若 $m > n$,则算法的时间复杂度为 $O(m) * \log(m)$ 。

由于参与交叉的基因是从染色体中选取出的最大

的基因,因此,相对于传统算法随机选择交叉基因,算法可以更快地提升子代的适应度、从而获得更好的装箱方案。

3.5 变异算子

在文献[21]中,针对分组遗传算法提出了一种变异算子来解决 BPP 问题。其变异操作的思路是从变异染色体中随机选择并删除基因(即装箱问题中的箱子),然后,将被删除的箱子中的那些物品采用 First-Fit 策略重新进行装箱操作,以得到变异后的装箱方案。

为了使变异操作得到更好装箱方案,EEGGA 算法对变异算子进行了优化,即在染色体中确定变异的基因/箱子时并不是随机选择,而是选取所装物品数量最多的箱子/基因进行删除和重装的变异操作。由于箱中的物品数越多,其中每个物品的平均尺寸则越小,从而这些物品将能以更大的概率在不开启新箱子的前提下装入现有箱子中,因而有助于减少装箱的数量和得到更优的装箱方案。

4 实验及结果分析

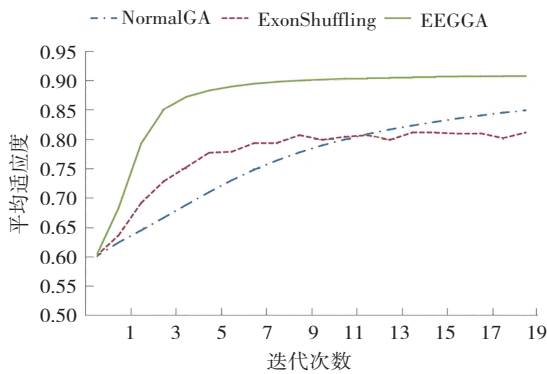
为了验证所提出的 EEGGA 算法的优越性,通过仿真实验对算法的性能进行评估。由于算法是采用改进的遗传算法解决 VMP 问题,以降低数据中心能耗和加快遗传迭代求解的速度,因此,在实验中,将提出的 EEGGA 算法和具有较好求解性能及收敛速度的外显子混洗交叉(记为 ExonShuffling)遗传算法^[22]以及未采取优化策略的传统分组遗传算法(记为 NormalGA)在节能性能和收敛速度方面进行了比较。

实验的测试环境为 Windows 10(64 bit)、Core i7 10810U CPU、32GB RAM、Java 1.8、IntelliJ IDEA 2021.2 Community Edition。遗传算法的参数设置为:交叉概率 50%;变异概率 10%,以提高遗传迭代过程中子代进化的速度;EEGGA 算法中交叉算子中的交叉因子设置为 $\alpha = 0.25$,即父代染色体平均长度 1/4 的基因进行交叉操作;实验评估了各算法在两种问题规模下的性能,即物品/虚拟机数量 m 为 120 和 200,两种规模对应的种群大小分别设置为 60 和 80;最大遗传迭代次数为 20 次,一是为了尽量减少算法执行时间造成的开销,二是评估各算法是否能够在较少的迭代次数收敛;公平起见,实验结果数据均为各个算法运行 100 次的平均值。

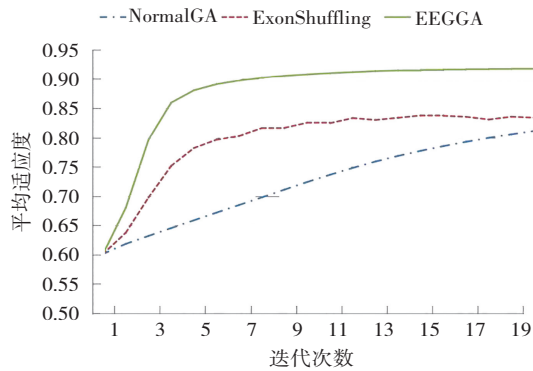
由于节能 VMP 问题可视为 BPP 问题,且能耗与装箱数/物理机数量呈线性关系^[14],因此装箱的数量即能反映节能效果。图 2—图 4 分别对比了 3 种算法的平均适应度、最好适应度以及最优解对应的装箱数/物理机数随着遗传算法迭代次数的变化趋势。

从图 2、图 3 中 3 个算法的对比可以明显看出,在迭代的过程中,提出的 EEGGA 算法具有较好的适应

度。具体而言,在 3 个算法中,NormalGA 算法的适应度提升较缓、其收敛速度相对较慢,这是由于 NormalGA 采用未经优化的传统遗传算法的算子,当物品数量/虚拟机数 m 值为 120 时,在 20 次迭代之内尚可获得比 ExonShuffling 算法更好的适应度;但当 m 值较大 ($m = 200$) 时,其性能反而不如 ExonShuffling 算法,这是由于 ExonShuffling 算法对遗传算子进行了一定的优化。另外,由于 ExonShuffling 算法所采用的变异算子未考虑装箱的可行性^[22],在变异过程中可产生不可行个体而影响整体的适应度,因此,在迭代过程中,其平均适应度存在抖动的现象。



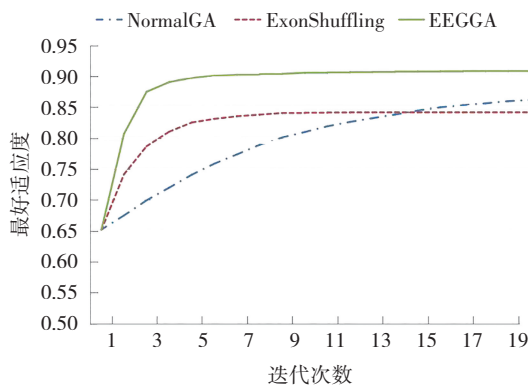
(a) $m = 120$



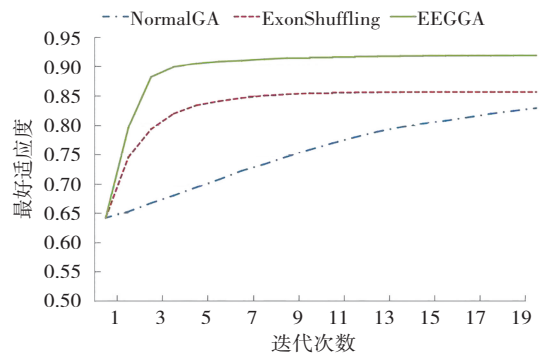
(b) $m = 200$

图 2 算法平均适应度随迭代次数的变化

Fig. 2 Change in average fitness of algorithm with the number of iterations



(a) $m = 120$

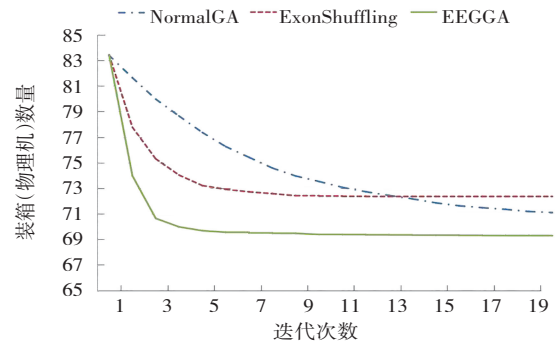


(b) $m = 200$

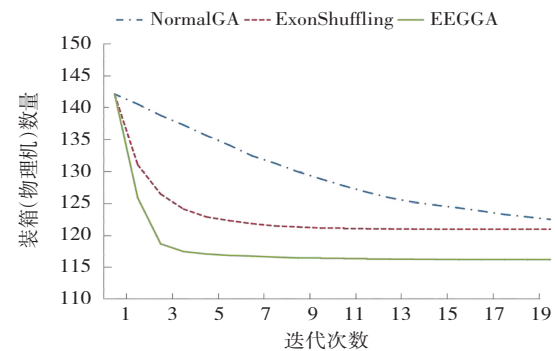
图 3 算法的最好适应度随迭代次数的变化

Fig. 3 The best fitness of algorithms changes with iteration

图 4 比较了 3 个算法在其迭代进化过程中最优解对应的装箱数量/物理主机数随迭代次数的变化趋势。从图 4 中可以看出,提出的 EEGGA 算法的收敛速度和求解性能均明显优于另外两个算法,这是由于其在交叉和变异等阶段采用了启发优化策略。另外,相对 NormalGA 算法而言,EEGGA 和 ExonShuffling 两个算法均具有较好的收敛速度,大约在第 10 次迭代之前即能收敛,但 EEGGA 算法的求解效果优于 ExonShuffling 算法,可减少使用 3~5 个装箱/物理机;而 NormalGA 算法则未能在 20 次迭代之内收敛,其收敛速度和求解性能表现均相对较弱,仅在物品数量较少(问题规模较小)的时候,其性能略优于 ExonShuffling 算法。



(a) $m = 120$



(b) $m = 200$

图 4 算法最优解(箱子/物理机数量)对比

Fig. 4 Comparison of optimal solutions of three algorithms (number of bins/physical machines)

5 结论

云数据中心的巨大能耗是当前亟待解决的一个重要问题。对虚拟机放置 (Virtual Machine Placement, VMP) 进行优化是降低云数据中心能耗和提高其能效的重要技术之一。由于 VMP 优化是一个 NP-hard 问题,目前的相关研究多采用启发式优化算法解决 VMP 节能优化问题。提出了一种基于分组遗传算法的 EEGGA 节能算法以解决数据中心的虚拟机节能放置问题。首先,由于传统的常规编码方式对 VMP 问题进行编码可导致不可行的映射方案等问题,提出的算法对染色体采用基于组的方式并结合启发方法进行编码,以避免产生不可行解的弊端;进而,在算法迭代过程中的交叉和变异等阶段,通过设计启发式优化策略,使种群的个体能够分别通过父代的交叉操作以及自身的变异操作得到更好的基因,从而提高算法的迭代进化速度和节能性能。仿真实验表明:与其他同类遗传算法相比,提出的算法不但能够降低云数据中心的能量消耗,同时还有效地提高了迭代求解的收敛速度。

参考文献(References):

- [1] SHEHABI A, SMITH S J, MASANET E, et al. Data center growth in the United States: decoupling the demand for services from electricity use [J]. *Environmental Research Letters*, 2018, 13(12): 124030.
- [2] JONES N. How to stop data centres from gobbling up the world's electricity[J]. *Nature*, 2018, 561: 163—166.
- [3] 肖枝洪, 王明浩. 中国碳排放量的组合模型及预测[J]. *重庆工商大学学报(自然科学版)*, 2016, 33(1): 9—15.
XIAO Zhi-hong, WANG Ming-hao. Prediction of carbon emission quantity of China based on combined model [J]. *Journal of Chongqing Technology and Business University (Natural Science Edition)*, 2016, 33(1): 9—15.
- [4] DINESH REDDY V, GANGADHARAN G R, RAO G S V R K. Energy-aware virtual machine allocation and selection in cloud data centers [J]. *Soft Computing*, 2019, 23 (6): 1917—1932.
- [5] ALBOANEEN D A, TIANFIELD H, ZHANG Y. Metaheuristic approaches to virtual machine placement in cloud computing: a review [C]//*Proceedings of the 15th International Symposium on Parallel and Distributed Computing*, 2016: 214—221.
- [6] LU J, ZHAO W, ZHU H, et al. Optimal machine placement based on improved genetic algorithm in cloud computing[J]. *The Journal of Supercomputing*, 2022, 78(3): 3448—3476.
- [7] ABDESSAMIA F, ZHANG W Z, TIAN Y C. Energy-efficiency virtual machine placement based on binary gravitational search algorithm[J]. *Cluster Computing*, 2020, 23(3): 1577—1588.
- [8] ABOHAMAMA A S, HAMOUDA E. A hybrid energy-Aware virtual machine placement algorithm for cloud environments [J]. *Expert Systems with Applications*, 2020, 150: 113306.
- [9] AL-MOALMI A, LUO J, SALAH A, et al. Optimal virtual machine placement based on grey wolf optimization [J]. *Electronics*, 2019, 8(3): 283—292.
- [10] USMAN M J, ISMAIL A S, CHIZARI H, et al. Energy-efficient virtual machine allocation technique using flower pollination algorithm in cloud datacenter: a panacea to green computing[J]. *Journal of Bionic Engineering*, 2019, 16(2): 354—366.
- [11] 王静莲, 龚斌, 刘弘, 等. 软硬件节能原理深度融合之绿色异构调度算法[J]. *软件学报*, 2021, 32(12): 3768—3781.
WANG Jing-lian, GONG Bin, LIU Hong, et al. Green heterogeneous scheduling algorithm through deep integration of hardware and software energy saving principles[J]. *Journal of Software*, 2021, 32(12): 3768—3781.
- [12] ZHANG B, WANG X, WANG H. Virtual machine placement strategy using cluster-based genetic algorithm[J]. *Neurocomputing*, 2021, 428: 310—316.
- [13] 王继业, 周碧玉, 张法, 等. 数据中心能耗模型及能效算法综述[J]. *计算机研究与发展*, 2019, 56(8): 1587—1603.
WANG Ji-ye, ZHOU Bi-yu, ZHANG Fa, et al. Data center energy consumption models and energy efficient algorithms [J]. *Journal of Computer Research and Development*, 2019, 56(8): 1587—1603.
- [14] FAN X, WEBER W D, BARROSO L A. Power provisioning for a warehouse-sized computer [C]//*Proceedings of the 34th annual international symposium on Computer architecture*, 2007: 13—23.
- [15] TANG M, PAN S. A hybrid genetic algorithm for the energy-efficient virtual machine placement problem in data centers [J]. *Neural Processing Letters*, 2015, 41(2): 211—221.
- [16] BELOGLAZOV A, ABAWAJY J, BUYYA R. Energy-aware resource allocation heuristics for efficient management of data centers for Cloud computing[J]. *Future Generation Computer Systems*, 2012, 28(5): 755—768.
- [17] WU Q, ISHIKAWA F. Heterogeneous virtual machine consolidation using an improved grouping genetic algorithm [C]//*Proceedings of the IEEE 17th International Conference on High Performance Computing and Communications*, 2015 IEEE 7th International Symposium on Cyberspace Safety and Security, and 2015 IEEE 12th International Conference on Embedded Software and Systems, 2015: 397—404.
- [18] FALKENAUER E. A hybrid grouping genetic algorithm for Bin packing[J]. *Journal of Heuristics*, 1996, 2(1): 5—30.
- [19] EPSTEIN L, JOHNSON D S, LEVIN A. Min-sum Bin packing[J]. *Journal of Combinatorial Optimization*, 2018, 36 (2): 508—531.
- [20] KAYA M. The effects of a new selection operator on the performance of a genetic algorithm [J]. *Applied Mathematics and Computation*, 2011, 217(19): 7669—7678.
- [21] FALKENAUER E, DELCHAMBRE A. A genetic algorithm for Bin packing and line balancing [C]//*Proceedings of the Proceedings 1992 IEEE International Conference on Robotics and Automation*, 2002: 1186—1192.
- [22] ROHLFSHAGEN P, BULLINARIA J A. A genetic algorithm with exon shuffling crossover for hard Bin packing problems [C]//*Proceedings of the 9th annual conference on Genetic and evolutionary computation*, 2007: 1365—1371.

责任编辑:田 静