Cluster Computing (2023) 26:575–586
https://doi.org/10.1007/s10586-022-03608-0

Design of cultural emperor penguin optimizer for energy-efficient resource scheduling in green cloud computing environment

Romany F. Mansour[1] · Hesham Alhumyani[2] · Sayed Abdel Khalek[3] · Rashid A. Saeed[2] · Deepak Gupta[4]

Received: 20 August 2021 / Revised: 23 January 2022 / Accepted: 31 January 2022 / Published online: 17 May 2022

Abstract

In recent times, energy related issues have become challenging with the increasing size of data centers. Energy related issues problems are becoming more and more serious with the growing size of data centers. Green cloud computing (GCC) becomes a recent computing platform which aimed to handle energy utilization in cloud data centers. Load balancing is generally employed to optimize resource usage, throughput, and delay. Aiming at the reduction of energy utilization at the data centers of GCC, this paper designs an energy efficient resource scheduling using Cultural emperor penguin optimizer (CEPO) algorithm, called EERS-CEPO in GCC environment. The proposed model is aimed to distribute work load amongst several data centers or other resources and thereby avoiding overload of individual resources. The CEPO algorithm is designed based on the fusion of cultural algorithm (CA) and emperor penguin optimizer (EPO), which boosts the exploitation capabilities of EPO algorithm using the CA, shows the novelty of the work. The EERS-CEPO algorithm has derived a fitness function to optimally schedule the resources in data centers, minimize the operational and maintenance cost of the GCC, and thereby decrease the energy utilization and heat generation. To ensure the improvised performance of the EERS-CEPO algorithm, a wide range of experiments is performed and the experimental outcomes highlighted the better performance over the recent state of art techniques.

Keywords Green cloud computing · Energy efficiency · Metaheuristics · Load balancing · Scheduling

1 Introduction

Cloud computing (CC), a sub-region of ICT, is the research topic on the environment. There are views and arguments against and for this technology. Excepting the interests shown by the provider of cloud type services and products [1, 2], there is significant pressure from the government organization for reducing negative effects on the environments. The growth of Green Cloud Computing (GCC) is highly associated with the development of green datacentres since the data centers are the essence of the CC environment. As [3] the power expended by the datacentres in 2010 was 1.3% of the overall consumptions. The study reported by GeSI [4], i.e., deliberated "one of the well-

✉ Romany F. Mansour
romanyf@sci.nvu.edu.eg

Hesham Alhumyani
Heshamalhumyani@tu.edu.sa

Sayed Abdel Khalek
sayedquantum@yahoo.co.uk

Rashid A. Saeed
eng_rashid@hotmail.com

Deepak Gupta
deepakgupta@mait.ac.in

[1] Department of Mathematics, Faculty of Science, New Valley University, El-Kharga 72511, Egypt

[2] Department of Computer Engineering, College of Computers and Information Technology, Taif University, P.O. Box 11099, Taif 21944, Saudi Arabia

[3] Mathematics Department, Faculty of Sciences, Taif University, P.O. Box 11099, Taif 21944, Saudi Arabia

[4] Department of Computer Science & Engineering, Maharaja Agrasen Institute of Technology, Delhi, India

recognized and most comprehensive snapshots of the Internet power demands around the world", evaluates a rise in the share of overall carbon dioxide (CO_2) emissions from ICT from 1.3% of global emission in 2002 to 2.3% in 2020. Using power utilization and CC environment, a set of authors at Lawrence Berkeley National Laboratory and North-western University proposed a modelling equipment named the Cloud Energy and Emissions Research Model (CLEER). Such server's farm creates the cloud. The result estimates that the productivity software and Customer Relationship Management software, primary energy footprint of email could be minimized as 87% when each business user in the US moved to CC platform [5]. Also when the model doesn't consider each variable, it could demonstrate beneficial as a result energetic efficacy in the datacentres that belong to Internet Company. It can guarantee a rise in energetic transparency and notify consumers for enabling them to select a better offer. The advantages of CC environment are very important for environmental protection when datacentres are based on green computing principles. Figure 1 depicts the task scheduling (TS) process in GCC.

Datacentre consumes around 1.3% of the overall global electrical energy supplies, i.e., anticipated to rise up to 8% in 2020 [6, 7]. Hence, CO2 significantly increases that directly impacts the environment. Unfortunately, huge amount of electricity is exhausted by the server at the time of lower task [8]. The Quality of Service (QoS) constraints play a significant role among users and mobile cloud service provider. Hence, the major problem is to minimalize energy utilization of mobile cloud datacentres when fulfilling QoS requirement [9]. The hardware virtualization technologies transform conventional hardware into a novel model. Such technologies consolidate tasks, named virtual machine (VM) consolidation, and exploit lower power hardware state. One of the present researches has minimalized the total power consumptions via 2 commonly employed methods like dynamic server provisioning and VM consolidation. Dynamic server provisioning method reduces energy utilization by decreasing the computation resource at the time of lower tasks [10]. These reductions mean turning the redundant server to sleep mode while the task demands decrease. Likewise, while data storage demand and processing increase, this server is reactivate based on the requirement. The servers share its resource amongst many efficiently isolated environments named VM with the help of hypervisor technique.

Resource utilization in cloud is based on the sequence and type of resources and tasks. Workflow technology is employed for handling data-intensive applications, increasing complex data, analysis, and simulations. Also, this technology is employed for scheduling computation tasks on allocated resources, for managing dependencies between tasks and stage datasets to and out of implementation site. This workflow is applied for modelling computation in various fields. Several TS algorithms are presented in this study, i.e., mainly categorized to Level-by-level scheduling, list scheduling algorithm, duplication based scheduling, batch scheduling, batch dependency scheduling technique, dependency scheduling, hybrid algorithm and Genetic Algorithm (GA) based scheduling technique. List scheduling algorithms create a list of tasks when accounting task dependencies. The task in the lists is treated for their existence in the task list. The efficacy of these algorithms is relatively better compared to other classifications of techniques. Level-by-level scheduling algorithm considers task of single level in task graph thus the task deliberated are independent of one another.

This paper presents an energy efficient resource scheduling using Cultural emperor penguin optimizer (CEPO) algorithm, called EERS-CEPO in GCC environment. The CEPO algorithm incorporates the cultural algorithm (CA) into emperor penguin optimizer (EPO) in order to boost the exploitation capabilities of EPO algorithm. The EERS-CEPO algorithm has derived a fitness function for optimal resource scheduling in data centers, minimize the operational and maintenance cost of the GCC, and thereby decreases the energy utilization and heat generation. To ensure the improvised performance of the EERS-CEPO algorithm, a wide range of experiments is performed.

2 Literature review

Zong [11] integrates GA and ACO algorithms for proposing a dynamic fusion TS approach. Thus decreasing the power utilization of computing centres and CC datacentres. The experimental result shows that the presented method could considerably decrease the time and overall power utilization of CC systems. Jena [12] focus on TSCSA method for optimizing processing and energy time. The results attained using TSCSA technique were inspired by a public domain source cloud environment (CloudSim). Lastly, the outcomes are related to the present scheduling algorithm and establish that the presented method offers an optimum balance result for many objectives.

Thaman and Singh [13] proposed a strong hybrid planning approach, RHEFT method to bind tasks for VM. The distribution of tasks to VM is depending upon a new task matching approach named Interior Scheduling. Yuan et al. [14] presented an STSRO approach for minimizing the overall cost of their providers by cost-efficiently scheduling each incoming task of heterogeneous application for meeting task' delay-bound limitations. STSRO approach exploits spatial diversity in DGCDC. In every time slot, the

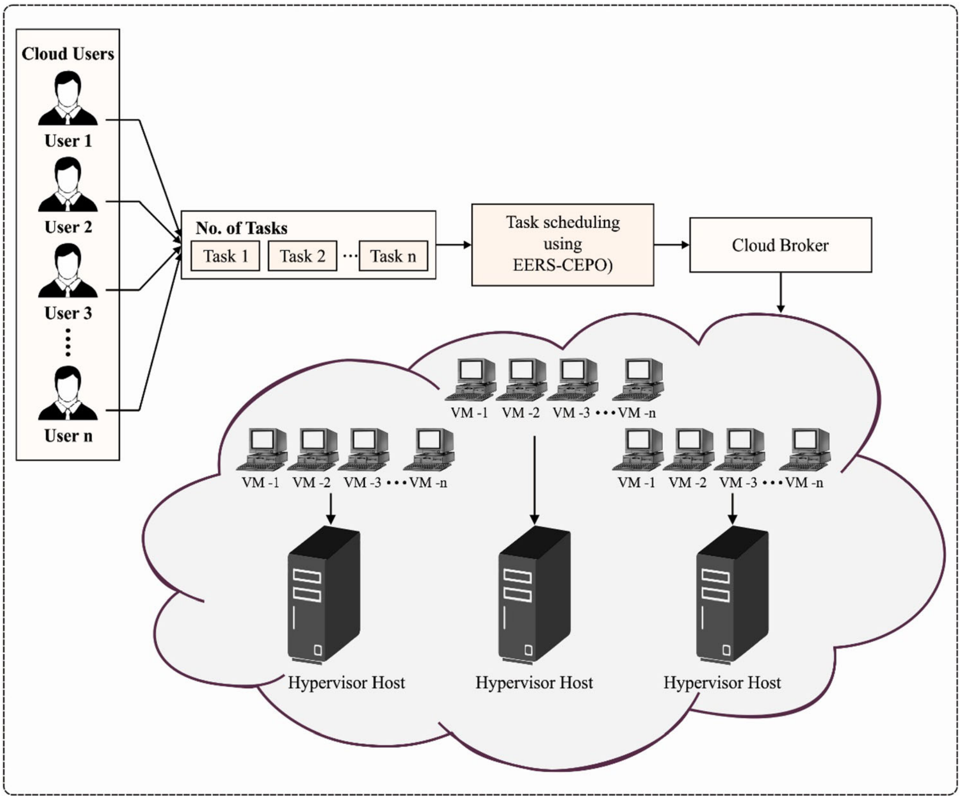

Fig. 1 Task scheduling process in GCC

cost reduction problems for DGCDC are created as a constraint optimization one and resolved by the presented SBA approach.

Ranjan et al. [15] proposed a GCTA method on the basis of enhanced BPSO algorithm. The major contributions of this study are to avoid matrix operation with the help of pipelined amounts for VMs and re-determining the velocity and position of particles. Experimental results show that the presented method has lesser implementation time, and decreases resource utilization consequently. Yuan et al. [16] consider the trade-offs among energy cost minimization and profit maximization for the GCDC provider when encountering the delay constraint of each task. The present TS method fails to take the advantage of spatial variation in various aspects for example amount of electrical energy and accessibility of renewable power production at geographic distributed GCDC locations. Consequently, it can be fail for performing each task of heterogeneous application within their delay bound in a lower power cost and higher revenue manner. In this study, a multi-objective optimization algorithm tackles the disadvantage of the present method is presented.

Mishra et al. [17] examined the power utilization in CC environments depending on variety of services and attained the provision for promoting GCC. This would assist to retain total power consumption of the scheme. Task distribution in the CC environments is a familiar problem and using these problems, could enable GCC. Also, they have presented an adaptive task allocation method for the heterogeneous CC environments. The study employed presented method for minimizing the makespan of the cloud scheme and reduces the power utilization. Yuan et al. [18] adapt a G/G/1 queueing scheme for analyzing the efficiency of the server in DGC. Depending on it, a single objective constrained optimization problems are solved and formulated by a presented SBA approach finding SBA method could minimalize the power costs of DGC providers by optimally assigning task of heterogeneous applications amongst many DGCs, and specify the operating speed of every server and the amount of powered-on servers in every GC when severely meet the response time limit of task of each application.

Abualigah and Diabat [19] projected a new hybrid antlion optimization approach using elite based differential evolution to solve multiobjective TS challenges in CC environment. The antlion optimization approach has been improved with the help of elite-based differential evolution as a local search approach for improving its exploitation capability and prevent from getting trapped in local optimal. In Shu et al. [20], robust agile response TS optimization algorithms are presented based on the peak energy utilization of datacentres and the time span of TS.

Also, agile response optimization technique is adapted. From the perception of task failure rate, the presented method could be employed for investigating the robust agile response optimization method, request a timeout for avoiding network congestions, and probability density function of the task request queue overflow.

Peng et al. [21] presented an effective real time deterministic scheduling (RTDS) model. Primarily, the SSDT can be mathematically formulated as a multi-way flow scheduling issue. Next, the RTDS designed scheduling request pre-processing (SRP) and greedy based multi-channel time slot allocation (GMT) to schedule in an optimal way. Yang et al. [22] presented a smart trust cloud management approach, which includes a trust cloud updating scheme. The experimental outcome demonstrated that the presented model can proficiently resolve the trust uncertainty problem and enhance the detection rate of the malicious devices. Wang et al. [23] presented a delay aware secured non-orthogonal multiple access (NOMA) transmission model with high altitude platform (HAP) and low altitude platform (LAP) collaborated to steadily offer delay sensitive healthcare services. An efficient AP-LAP secure transmission model has been presented for offering NOMA communication services for many hotspots. With the restricted energy as well as spectrum, privacy information delay can be reduced.

Ding et al. [24] developed a generative adversarial network for image restoration among distinct kinds of deprivation. The presented model presented an effective model with an inverse and a reverse module for addressing extra attributes among the image styles. With secondary data, the restoration can be highly accurate. Besides, a loss function can be developed for the stabilization of adversarial training with improved training performance. Tan et al. [25] proposed a blockchain enabled security as well as privacy protection model with traceable and direct revocation for COVID-19 healthcare data. Here, the blockchain is employed for uniform identity authentication and every public key, revocation list, and so on are saved on a blockchain. The system manager server has the responsibility of generate system parameters and published the private key for COVID-19 healthcare professionals.

3 Problem formulation

LB is a widely employed approach to maximize throughput, optimize resource utilization, ensure fault-tolerant configuration, and reduce latency. In GCC infrastructure, LB is an approach applied for spreading workloads amongst numerous datacenters or other resource nodes, thus no individual resource is over-loaded. For better describing the scheme of the LB, dynamic, power utilization optimization methods could be provided in the following. Datacenters are usually made up of many resources sites shared in distinct geographical places in GCC. Actually, every subtask t_i $(i = 1, 2, \ldots, m)$ could attain a resource site R_j for meeting the fundamental execution condition, where n signifies the amount of resources for meeting the subtask t_i. The Poisson distributions are appropriate to describe the amount of arbitrary actions arising within a unit time (or space). Hence, the dynamic procedure of TS and RA in CC infrastructure could be deliberated as a Poisson distribution.

Definition 1 Consider that GCC method $GCloud = (D, T, P, G)$, $D = \{D_1, D_2, D_d\}$ is a set of d cloud server datacenters, T represents the correspondence matrix among computing node and random task, T_{ij} signifies the task t_i is performed on node j. Pi denotes the power of node i while it can be idle, and G_i means the peak power of node [26].

Definition 2 Consider that the network bandwidth matrix of all datacenters is B in GCC, b_{ij} represents the broadcast bandwidth of every node in the datacenter.

$$B = \begin{bmatrix} b_{11} & b_{12} & \cdots & b_{1d} \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ b_{d1} & b_{d2} & \cdot & b_{dd} \end{bmatrix} \tag{1}$$

For minimizing the power utilization in the cost constrained energy optimization, is for estimating the cost of power utilization in the procedure of the accomplishment of the CC.

Definition 3 Consider that Q_i signifies the amount of instruction of taskst_i, μ_i denotes the usage rate of node i, f_i indicates the frequency of node i, the runtime of task t_i on node i is Q_i/V_j^i. The energy of server perform the task t_i could be expressed by

$$\begin{cases} E_j^i = \lambda \times u_i \times (Q_i/V_j^i)^{\beta_i - 1} \times e^{f_i} \\ \beta_i = 1 + \dfrac{1}{\alpha_i} \geq 2 \\ \alpha_i \in (0, 1] \end{cases} \tag{2}$$

where, $i = 1, 2, \ldots, m, j = 1, 2, \ldots, n_i$.

Reducing the power utilization of computing nodes could be understood by decreasing the percentage of idle nodes X. The power utilization of GCC environment is provided below

$$E = \sum_{i=1}^{m} \sum_{j=1}^{n} \lambda \times u_i \times (Q_i/V_j^i)^{\beta_i - 1} \times e^{f_i} \tag{3}$$

With the rapid growth of CC, the server scale of cloud datacenter is continually increasing each year that causes enormous energy utilization. Moreover, unreasonable scheduling policy leads to power waste, make the datacenter functioning cost constantly expanding.

4 The proposed resource scheduling technique

EPO is a new optimized technique projected by Dhiman and Kumar in 2018 [27] that is simulated as budding performance of emperor penguins. The hybrid technique is called CEPO was presented for solving the real numbers optimized issues that are with the use of cultural technique. An important idea of CEPO is for obtaining problem solving technique in the budding performance of emperor penguins and create utilize of data for guiding the development of emperor penguins population from returned. Assume that CEPO has been planned to common minimal optimized issue:

$$min\, f(x_i), \tag{4}$$

where $\mathrm{x}_i = (x_{i1}, x_{i2}, \ldots, x_{iD})$ refers the place of i^{th} emperor penguin from D dimension search space,$x_j^{\min} < x_{ij} < x_j^{\max}, (j = 1, 2, \ldots, D)$. $f()$ represents the objective functions, and $f(\mathrm{x}_i)$ indicates the objective value of placex_i. $x_j^{\min}$ and $x_j^{\max}$ implies the lower as well as upper boundary of the place of emperor penguin from the jth dimensional.

A detailed space of emperor penguins' population in t^{th} generation determined as provided by s^t and N_j^t, in which s^t refers the situational knowledge modules. N_j^t refers the normative knowledge that signifies the value space data to all parameters from jth dimensional and in tth generation. N_j^t stands for I, L, U. $\mathrm{I}_j^t = [l_j^t, u_j^t]$, where I_j^t demonstrates the interval of normative knowledge from jth dimensional. The lower boundary l_j^t and upper boundary u_j^t are adjusted based on value range of variables provided as this issue. L_j^t implies the objective value of lower boundary l_j^j of jth parameter, and U_j^t implies the objective value of upper boundary u_j^t of jth parameter. The acceptance function has been utilized for selecting the emperor penguin is directly controls the present confidence space. Figure 2 illustrates the process involved in CEPO technique.

The situational knowledge s^t is upgraded as update function:

$$s^{t+1} = \begin{cases} x_{best}^{t+1}; if f\left(x_{best}^{t+1}\right) < f(s^t), \\ s^t, \text{else}, \end{cases} \tag{5}$$

where $\mathrm{x}_{\mathrm{best}}^{t+1}$ implies the optimum place of emperor penguin population space from $(t+1)$th generation. Considered that, for the qth cultural individuals, an arbitrary variable θ_q lies in the range of 0 and 1 has formed [28]. The qth cultural individuals affect the lower boundary of normative knowledge from jth dimensional if $\theta_q < 0.5$ has been fulfilled. The normative knowledge N_j^t is upgraded by update function:

$$l_j^{t+1} = \begin{cases} x_{qj}^{t+1}, & if\ x_{qj}^{t+1} \le l_j^t\ or\, f\left(x_q^{t+1}\right) < L_j^t, \\ l_j^t, & \text{else}, \end{cases} \tag{6}$$

$$L_j^{t+1} = \begin{cases} f\left(\mathrm{x}_q^{t+1}\right), & if\ x_{qj}^{t+1} \le l_j^t\ or\, f\left(\mathrm{x}_q^{t+1}\right) < L_j^t, \\ L_j^t, & \text{else}. \end{cases}$$

The qth cultural differences affect the upper boundary of normative knowledge from jth dimensional if $\theta_q \ge 0.5$ is fulfilled:

$$u_j^{t+1} = \begin{cases} x_{qj}^{t+1}, & ifx_{qj}^{t+1} \le u_j^t orf\left(\mathrm{x}_q^{t+1}\right) < U_j^t, \\ u_j^t, & \text{else}, \end{cases} \tag{7}$$

$$U_j^{t+1} = \begin{cases} f\left(\mathrm{x}_q^{t+1}\right), & ifx_{qj}^{t+1} \le u_j^t orf\left(\mathrm{x}_q^{t+1}\right) < U_j^t, \\ U_j^t, & \text{else}. \end{cases}$$

The situational and normative knowledge is utilized for guiding emperor penguin population development by control functions. In CEPO algorithm, a selective operator β has been created for selecting one of two approaches for controlling the of development emperor penguin populations:

$$\beta = \frac{\mathrm{Max}_{\mathrm{iteration}} - t}{\mathrm{Max}_{\mathrm{iteration}}}, \tag{8}$$

where $\mathrm{Max}_{\mathrm{iteration}}$ refers the maximal number of iterations. Considered that for ith emperor penguins, an arbitrary variable λ_i that lies from the range of 0 and 1 has been created. An initial method is for updating the place of emperor penguin with altering the search size and way of difference with confidence space that is executed if fulfilled $\lambda_i \le \beta$. The place of emperor penguin from jth dimensional is upgraded as:

$$x_{ij}^{t+1} = \begin{cases} x_{ij}^t + \left|\mathrm{size}\left(\mathrm{I}_j^t\right) \cdot N(0,1)\right|, & ifx_{ij}^t < s_j^t, \\ x_{ij}^t - \left|\mathrm{size}\left(\mathrm{I}_j^t\right) \cdot N(0,1)\right|, & ifx_{ij}^t > s_j^t, \\ x_{ij}^t + \eta \cdot \mathrm{size}\left(\mathrm{I}_j^t\right) \cdot N(0,1), & \text{else}, \end{cases} \tag{9}$$

where $N(0,1)$ implies the arbitrary number exposing to typical normal distribution. size $\left(\mathrm{I}_j^t\right)$ represents the length of adaptable interval of jth parameter from confidence

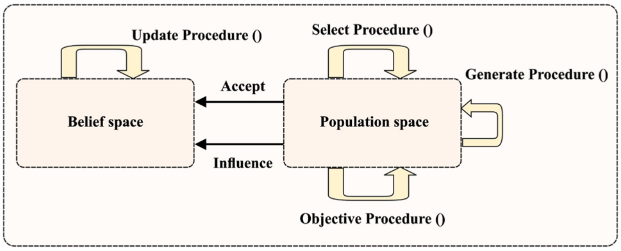

Fig. 2 Process involved in CEPO algorithm

space from tth generation. η fixed to be in the range of 0.01 and 0.61. Another manner is a series of steps from EPO that are huddle boundary generation, temperature profile on the huddle calculating, the distance computation amongst emperor penguin, and the place upgrade of emperor penguin that is implemented if the achieved $\lambda_i > \beta$. The particular steps are demonstrated as:

$$T' = T - \frac{t - \mathrm{Max}_{\mathrm{iteration}}}{\mathrm{Max}_{\mathrm{iteration}}}, \tag{10}$$

$$T = \begin{cases} 0, & \mathrm{if} R \geq 0.5, \\ 1, & \mathrm{if} R < 0.5, \end{cases}$$

where T' implies the temperature profile about the huddle, T refers the time to determine optimum solution, and R indicates the arbitrary variable that lies from the range of 0 and 1.

$$\mathrm{D}_{\mathrm{ep}}^{t} = S_{\mathrm{ep}}(A^{t}) \cdot \mathrm{x}_{\mathrm{best}}^{t} - B^{t} \cdot \mathrm{x}_{i}^{t}, \tag{11}$$

where $\mathrm{D}_{\mathrm{ep}}^{t}$ stands for the distance amongst emperor penguin as well as optimum solution, $\mathrm{x}_{\mathrm{best}}^{t}$ refers the present optimum solution initiate in emperor penguins population space from tth generation, S_{ep} implies the social force of emperor penguins which is responsible for convergence nearby the optimum solution, A^{t} and B^{t} are utilized for avoiding the collision amongst adjoining emperor penguin, and B^{t} indicates the arbitrary variable that lies from range of 0 and 1. A^{t} is calculated as:

$$A^{t} = (M \times (T' + P_{\mathrm{grid}}^{t}(\mathrm{Accuracy})) \times rand()) - T',$$

$$P_{\mathrm{grid}}^{t}(\mathrm{Accuracy}) = \left|\mathrm{x}_{\mathrm{best}}^{t} - \mathrm{x}_{i}^{t}\right|, \tag{12}$$

where M refers the effort parameter that hold the gap amongst emperor penguin to collision avoidance and P_{grid}^{t} (Accuracy) determines the absolute variance with relating the variance amongst emperor penguin. $S_{\mathrm{ep}}(A^{t})$ in Eq. (28) was calculated as:

$$S_{\mathrm{ep}}(A^{t}) = \sqrt{\varepsilon \cdot e^{-t/\rho} - e^{-t}}, \tag{13}$$

where e demonstrated the base of natural logarithm. ε and ρ are 2 influence parameters to optimum exploration as well as exploitation that is in the range of [1.5, 2] and [2, 3]. Eventually, the place of emperor penguins is upgraded as:

$$\mathrm{x}_{i}^{t+1} = \mathrm{x}_{\mathrm{best}}^{t} - A^{t} \cdot \mathrm{D}_{\mathrm{ep}}^{t}. \tag{14}$$

The EERS-CEPO algorithm is derived to reduce energy usage via optimal allocation of resources in GCC. Power utilization of the resource utilization and computing devices have a stronger relationship with the power wattage equation as follows.

$$P = E/T \tag{15}$$

$$E = P * T \tag{16}$$

In which the Power is estimated using P, number of times calculated using T, and energy estimated by E. The difference among energy and power is very important because decrease in energy consumption doesn't end in time to cut the expended power [29]. For storing the number of energy expended by the cloud resource is determined using the green cloud environment. The cloud resource has several classifications of servers, nodes, network topology, process units, storage facility, and power contribution unit the taken as full energy consumption at the cloud environment could be provided below

$$\mathrm{ECR} = \mathrm{EN} + \mathrm{ENET} + \mathrm{ES} + \mathrm{ESD} + \mathrm{EOE} \tag{17}$$

EN-Energy Consume from node, ECR-Cloud Resources energy, ES-Energy consumes from the server, EN-Energy Consume form network, EO-Energy consumes ES-Energy consumes from storage device, from another electrical equipment's. The overall energy exploitations of physical resources include 2 components as $\mathrm{E}_{\mathrm{VM}}$and Estatic. The Estatic is the secured energy of a server despite working VM or not and E_{VM} is the dynamic energy i.e., determined by VM working on it. Assume that there are nVM

$$Etotphy = Estatic + \sum Evm = Estatic + \sum\nolimits_{i}^{n} Evmi \tag{18}$$

$\mathrm{E}_{\mathrm{VMi}}$ could be further rotten keen on the power utilization of devices like memory, IO, and CPU represented as VMI, Ecpu, EIO VMI, Emem VMI. The EIO VMI consists of the general power cost of each device which involves I/O Operations like network data transfer and disk. The power utilization of VMI is

$$Evmi = E_{vmi}^{cpu} + E_{vmi}^{mem} + E_{vmi}^{IO} \tag{19}$$

The task of cloud assets such as disk storage, network interface, and CPU builds depiction on shift node on/ off to decrease the common power utilization. The cloud networks switch units of measurements using dynamism reconfigured for changing the action of the server when needed. The power utilization is decreased by shifting inactive to energy saving modes (hibernation, sleep).

5 Performance validation

For examining the improved efficacy of the EERS-CEPO technique, a sample GCC environment is considered with 6physcial machines (PMs) and every machine holds eight CPUs with the capacity power of 10,000 MIPS. The EERS-CEPO technique is simulated using CloudSim tool with job dispatcher, resource planner, cloud, and VM instances.

Table 1 and Fig. 3 demonstrate the response time analysis of the EERS-CEPO technique with existing techniques under distinct tasks. The results portrayed that the EERS-CEPO technique has accomplished effective outcomes with least response time under all tasks. For instance, with 50 tasks, the EERS-CEPO technique has offered a reduced response time of 236 ms whereas the CCS, ICSA, and CSRSA techniques have obtained a raised response time of 249 ms, 241 ms, and 287 ms respectively. Similarly, with 200 tasks, the EERS-CEPO technique has provided a lower response time of 597 ms whereas the CCS, ICSA, and CSRSA techniques have attained a higher response time of 969 ms, 807 ms, and 1031 ms respectively.

Meanwhile, with 400 tasks, the EERS-CEPO technique has achieved a decreased response time of 12646 ms whereas the CCS, ICSA, and CSRSA techniques have resulted in an increased response time of 1713 ms, 1558 ms, and 1457 ms respectively. Furthermore, with 600 tasks, the EERS-CEPO technique has exhibited a minimum response time of 1860 ms whereas the CCS, ICSA, and CSRSA techniques have demonstrated a maximum response time of 2604 ms, 2287 ms, and 2124 ms respectively.

Table 2 and Fig. 4 illustrate the response time analysis of the EERS-CEPO approach with recent methods under different iterations. The outcomes showcased that the EERS-CEPO approach has accomplished effective results with least response time under all iterations. For instance, with 25 iterations, the EERS-CEPO manner has accessible a reduced response time of 1867 ms whereas the CCS, ICSA, and CSRSA techniques have obtained a raised response time of 2503 ms, 2564 ms, and 2121 ms correspondingly. In line with, 75 iterations, the EERS-CEPO manner has provided a lesser response time of 1236 ms whereas the CCS, ICSA, and CSRSA algorithms have attained a superior response time of 1979 ms, 2025 ms, and 1419 ms respectively. Meanwhile, with 175 iterations, the EERS-CEPO method has reached a minimum response time of 641 ms whereas the CCS, ICSA, and CSRSA manners have resulted in an improved response time of 962 ms, 839 ms, and 753 ms respectively. Eventually, with 250 iterations, the EERS-CEPO technique has outperformed a minimal response time of 654 ms whereas the CCS, ICSA, and CSRSA techniques have portrayed an increased response time of 845 ms, 824 ms, and 789 ms correspondingly.

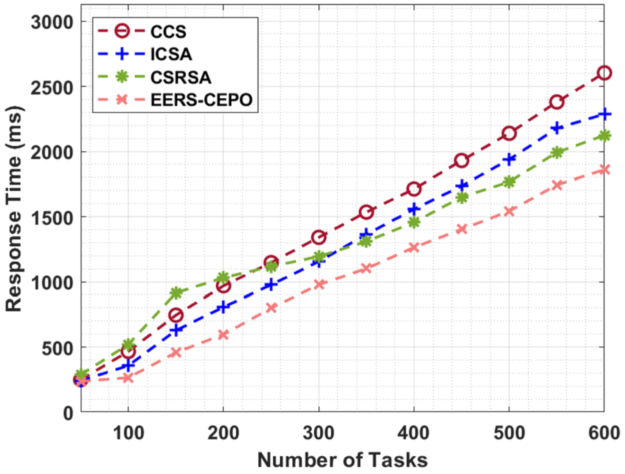

Fig. 3 Response time analysis of EERS-CEPO model with different tasks

Table 3 and Fig. 5 illustrate the execution time analysis of the EERS-CEPO method with state-of-art techniques under distinct tasks. The results portrayed that the EERS-CEPO manner has accomplished effective outcomes with least execution time under all tasks. For sample, with 50

Table 1 Response time analysis of EERS-CEPO technique

Response time (ms)				
Number of tasks	CCS	ICSA	CSRSA	EERS-CEPO
50	249	241	287	236
100	466	357	520	264
150	745	628	915	458
200	969	807	1031	597
250	1147	977	1117	799
300	1341	1155	1194	977
350	1535	1364	1310	1101
400	1713	1558	1457	1264
450	1930	1736	1651	1403
500	2139	1938	1767	1543
550	2380	2178	1992	1744
600	2604	2287	2124	1860

Table 2 Response time analysis of EERS-CEPO technique

Response time (ms)				
Number of iterations	CCS	ICSA	CSRSA	EERS-CEPO
25	2503	2564	2121	1867
50	2198	2187	1816	1608
75	1979	2025	1419	1236
100	1653	1608	1104	951
125	1384	1323	763	672
150	1094	911	773	651
175	962	839	753	641
200	860	829	758	638
225	834	829	768	656
250	845	824	789	654

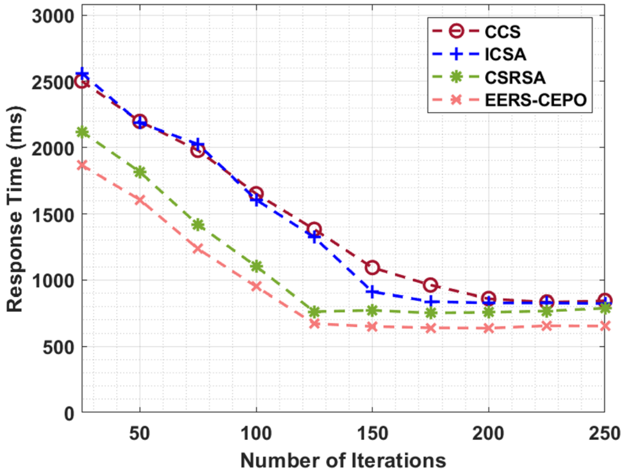

Fig. 4 Response time analysis of EERS-CEPO model with different iterations

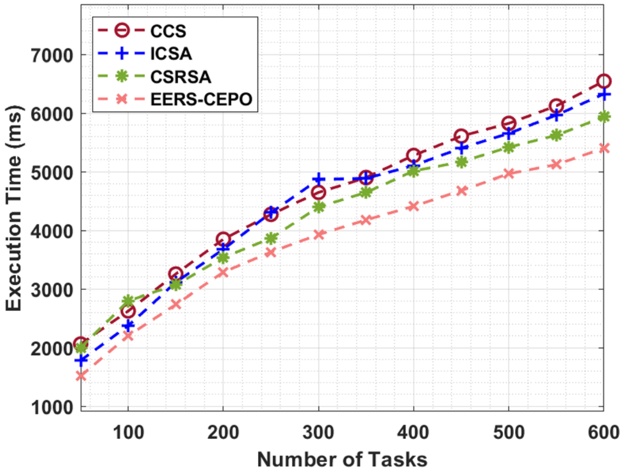

Fig. 5 Execution time analysis of EERS-CEPO model with different tasks

tasks, the EERS-CEPO technique has offered a decreased execution time of 1515 ms whereas the CCS, ICSA, and CSRSA techniques have reached an increased execution time of 2064 ms, 1782 ms, and 2001 ms correspondingly. Also, with 200 tasks, the EERS-CEPO technique has given a lower execution time of 3287 ms whereas the CCS, ICSA, and CSRSA systems have attained a higher execution time of 3851 ms, 3678 ms, and 3537 ms respectively. In the meantime, with 400 tasks, the EERS-CEPO method has reached a decreased execution time of 4415 ms whereas the CCS, ICSA, and CSRSA techniques have resulted in an increased execution time of 5277 ms, 5105 ms, and 5011 ms respectively. Furthermore, with 600 tasks, the EERS-CEPO technique has displayed a minimum execution time of 5403 ms whereas the CCS, ICSA, and CSRSA methodologies have outperformed a higher execution time of 6547 ms, 6328 ms, and 5936 ms correspondingly.

Table 4 and Fig. 6 depicts the energy consumption analysis of the EERS-CEPO technique with recent algorithms under different t cycle. The results portrayed that the EERS-CEPO technique has accomplished effective outcomes with least energy consumption under all cycles. For instance, with 1 cycle, the EERS-CEPO technique has offered a decreased energy consumption of 1.76 KWh whereas the CCS, ICSA, and CSRSA manners have attained an increased energy consumption of 2.32 KWh, 2.21 KWh, and 1.92 KWh correspondingly. Likewise, with 3 cycles, the EERS-CEPO manner has given a lesser energy consumption of 1.79 KWh whereas the CCS, ICSA, and CSRSA algorithms have attained a higher energy consumption of 2.51 KWh, 2.58 KWh, and 1.95 KWh correspondingly. Followed by, with 5 cycles, the EERS-CEPO technique has gained a minimal energy consumption of 1.59 KWh whereas the CCS, ICSA, and CSRSA approaches have resulted in improved energy consumption

Table 3 Execution time analysis of EERS-CEPO technique

Execution time (ms)				
Number of tasks	CCS	ICSA	CSRSA	EERS-CEPO
50	2064	1782	2001	1515
100	2628	2377	2801	2205
150	3255	3114	3067	2738
200	3851	3678	3537	3287
250	4274	4305	3867	3631
300	4650	4870	4400	3929
350	4901	4885	4650	4180
400	5277	5105	5011	4415
450	5607	5403	5168	4682
500	5826	5654	5418	4964
550	6124	5967	5622	5121
600	6547	6328	5936	5403

Table 4 Energy consumption analysis of EERS-CEPO technique

Energy consumption (KWh)				
Scheduling cycle	CCS	ICSA	CSRSA	EERS-CEPO
1	2.32	2.21	1.92	1.76
2	1.94	2.00	1.75	1.62
3	2.51	2.58	1.95	1.79
4	2.23	2.16	2.25	1.97
5	1.88	1.90	1.74	1.59
6	2.67	2.78	2.51	2.32
7	1.89	1.88	1.82	1.69
8	2.91	2.74	2.58	2.45

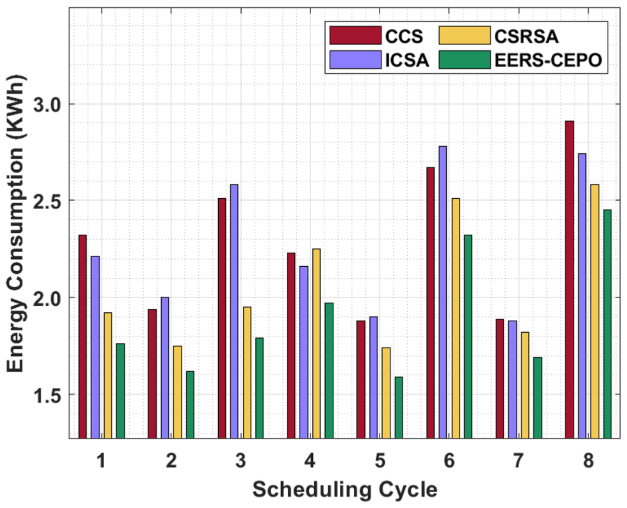

Fig. 6 Energy consumption analysis of EERS-CEPO model

of 1.88 KWh, 1.90 KWh, and 1.74 KWh respectively. Eventually, with 8 cycles, the EERS-CEPO technique has exhibited a minimum energy consumption of 2.45 KWh whereas the CCS, ICSA, and CSRSA techniques have outperformed a superior energy consumption of 2.91 KWh, 2.74 KWh, and 2.58 KWh correspondingly.

Table 5 and Fig. 7 showcases the average energy consumption (AEC) analysis of the EERS-CEPO manner with existing approaches under varying tasks. The results showcased that the EERS-CEPO technique has accomplished effective outcomes with least AEC under all tasks. For instance, with 50 tasks, the EERS-CEPO manner has presented a minimum AEC of 1.53 kJ whereas the CCS, ICSA, and CSRSA techniques have achieved a maximum AEC of 2.04 kJ, 1.85 kJ, and 1.85 kJ respectively. Similarly, with 200 tasks, the EERS-CEPO technique has provided a lower AEC of 2.89 kJ whereas the CCS, ICSA, and CSRSA approaches have attained a higher AEC of 3.61 kJ, 3.42 kJ, and 3.22 kJ respectively. Meanwhile, with 400 tasks, the EERS-CEPO system has reached a lower AEC of 5.08 kJ whereas the CCS, ICSA, and CSRSA techniques have resulted in an increased AEC of 6.93 kJ, 6.65 kJ, and 5.59 kJ correspondingly. Also, with 600 tasks, the EERS-CEPO manner has exhibited a minimum AEC of 7.69 kJ whereas the CCS, ICSA, and CSRSA algorithms have demonstrated a maximum AEC of 9.88 kJ, 9.49 kJ, and 8.57 kJ respectively.

Table 5 Average energy consumption analysis of EERS-CEPO technique

Average energy consumption (kJ)				
Number of tasks	CCS	ICSA	CSRSA	EERS-CEPO
50	2.04	1.85	1.85	1.53
100	2.55	2.45	2.39	1.95
150	3.03	2.94	2.75	2.45
200	3.61	3.42	3.22	2.89
250	4.23	4.09	3.82	3.49
300	4.92	4.74	4.44	4.07
350	5.89	5.75	5.02	4.65
400	6.93	6.65	5.59	5.08
450	7.88	7.46	6.31	5.80
500	8.73	8.41	6.95	6.54
550	8.91	8.68	7.88	7.07
600	9.88	9.49	8.57	7.69

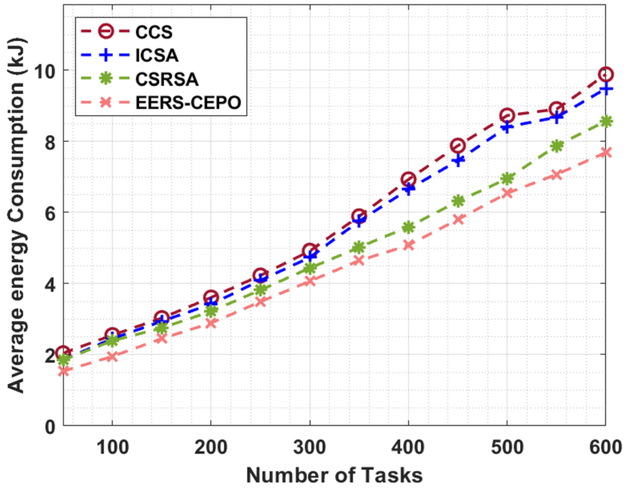

Fig. 7 AEC analysis of EERS-CEPO model with different tasks

Table 6 and Fig. 8 portray the average executive power (AEP) analysis of the EERS-CEPO method with existing techniques under distinct tasks. The results portrayed that the EERS-CEPO technique has accomplished effective outcomes with least AEP under all tasks. For instance, with 50 tasks, the EERS-CEPO technique has offered a reduced AEP of 1231 W whereas the CCS, ICSA, and CSRSA techniques have obtained a raised AEP of 1553 W, 1400 W, and 1676 W respectively.

Similarly, with 200 tasks, the EERS-CEPO technique has provided a lower AEP of 1998 W whereas the CCS, ICSA, and CSRSA techniques have obtained an increased AEP of 2366 W, 2244 W, and 2213 W respectively. At the same time, with 400 tasks, the EERS-CEPO manner has reached a lower AEP of 3348 W whereas the CCS, ICSA, and CSRSA algorithms have resulted in an increased AEP of 3916 W, 3839 W, and 3655 W respectively.

Finally, with 600 tasks, the EERS-CEPO methodology has exhibited a minimal AEP of 4376 W whereas the CCS, ICSA, and CSRSA methods have demonstrated a maximal AEP of 6170 W, 5787 W, and 4729 W respectively. The simulation results demonstrated that the EERS-CEPO

Table 6 Average executive power analysis of EERS-CEPO technique

Average executive power (W)				
Number of tasks	CCS	ICSA	CSRSA	EERS-CEPO
50	1553	1400	1676	1231
100	1707	1599	1737	1431
150	2014	1860	2075	1722
200	2366	2244	2213	1998
250	2673	2750	2566	2397
300	3087	2995	2965	2719
350	3532	3409	3317	3072
400	3916	3839	3655	3348
450	4468	4299	3824	3547
500	5051	4621	4222	3824
550	5634	5112	4422	4115
600	6170	5787	4729	4376

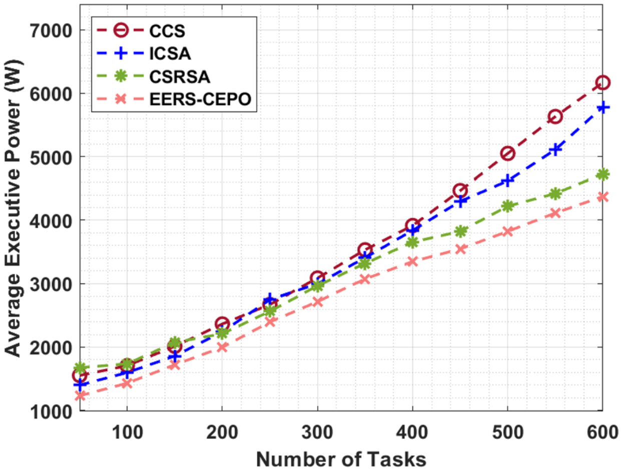

Fig. 8 AEP analysis of EERS-CEPO model with different tasks

technique has resulted in increased performance over the recent state of art schedulers in the GCC environment. The improved performance is due to the hybridization of EPO algorithm with CA.

6 Conclusion

This study has designed a new EERS-CEPO algorithm to reduce the energy utilization at the data centers of GCC. The proposed model has effectually allotted the load between distinct data centers or other resources and thereby avoiding overload of individual resources. The CEPO algorithm is designed by the integration of the EPO algorithm with the CA. The EERS-CEPO algorithm has derived a fitness function to optimally schedule the resources in data centers, thereby minimize the operational cost, energy utilization, heat generation, and maintenance cost of the GCC. An extensive simulation analysis is carried out to ensure the betterment of the EERS-CEPO algorithm. The simulation results demonstrated that the EERS-CEPO technique has resulted in increased performance over the recent state of art schedulers in the GCC environment. In future, VM migration techniques and fault tolerant approaches can be designed to maximize the efficiency of the GCC.

Acknowledgements Taif University Researchers Supporting Project Number (TURSP-2020/216), Taif University, Taif, Saudi Arabia.

Funding Funding provided by The Science, Technology & Innovation Funding Authority (STDF) in cooperation with The Egyptian Knowledge Bank (EKB)

Data Availability Data sharing not applicable to this article as no datasets were generated during the current study.

Declarations

Conflict of interest The authors declare that they have no conflict of interest. The manuscript was written through contributions of all authors. All authors have given approval to the final version of the manuscript.

Ethical approval This article does not contain any studies with human participants or animals performed by any of the authors.

References

1. Radu, L.D.: Green cloud computing: a literature survey. Symmetry **9**(12), 295 (2017)
2. Kumar, A.S., Venkatesan, M.: Task scheduling in a cloud computing environment using HGPSO algorithm. Clust. Comput. **22**(1), 2179–2185 (2019)
3. Ghafari, R., Kabutarkhani, F.H., Mansouri, N.: Task scheduling algorithms for energy optimization in cloud environment: a comprehensive review. Cluster Comput (2022). https://doi.org/10.1007/s10586-021-03512-z
4. Rahimikhanghah, A., Tajkey, M., Rezazadeh, B., Rahmani, A.M.: Resource scheduling methods in cloud and fog computing environments: a systematic literature review. Cluster Comput (2021). https://doi.org/10.1007/s10586-021-03467-1
5. Masanet, E., Shehabi, A., Ramakrishnan, L., Liang, J., Ma, X., Walker, B., Hendrix, V., Mantha, P.: The energy efficiency potential of cloud-based software: a U.S. case study. (2013)
6. Yadav, R., Zhang, W.: MeReg: managing energy-SLA tradeoff for green mobile cloud computing. Wireless Commun. Mob. Comput. (2017). https://doi.org/10.1155/2017/6741972
7. Lambert, S., Van Heddeghem, W., Vereecken, W., Lannoo, B., Colle, D., Pickavet, M.: Worldwide electricity consumption of communication networks. Opt. Express **20**(26), B513–B524 (2012)
8. Barroso, L.A., Holzle, U.: The case for energy-proportional computing. Comput. J. **40**(12), 33–37 (2007)
9. Corradi, A., Fanelli, M., Foschini, L.: VM consolidation: a real case based on OpenStack Cloud. Futur. Gener. Comput. Syst. **32**(1), 118–127 (2014)

10. Dashti, S.E., Rahmani, A.M.: Dynamic VMs placement for energy efficiency by PSO in cloud computing. J. Exp. Theor. Artif. Intell. **28**(1–2), 97–112 (2016)
11. Zong, Z.: An improvement of task scheduling algorithms for green cloud computing. In: 2020 15th International Conference on Computer Science & Education (ICCSE), August (pp. 654–657). IEEE (2020)
12. Jena, R.K.: Energy efficient task scheduling in cloud environment. Energy Procedia **141**, 222–227 (2017)
13. Thaman, J., Singh, M.: Green cloud environment by using robust planning algorithm. Egypt. Informat. J. **18**(3), 205–214 (2017)
14. Yuan, H., Bi, J., Zhou, M.: Spatial task scheduling for cost minimization in distributed green cloud data centers. IEEE Trans. Autom. Sci. Eng. **16**(2), 729–740 (2018)
15. Ranjan, A., Kaur, I., Bhardwaj, A., Sharma, V.: A Review of PSO-Based Task and Workflow Scheduling Analysis in Cloud Environment. SSRN J. (2021). https://doi.org/10.2139/ssrn.3842717
16. Yuan, H., Liu, H., Bi, J., Zhou, M.: Revenue and energy cost-optimized biobjective task scheduling for green cloud data centers. IEEE Trans. Autom. Sci. Eng. **18**(2), 817–830 (2020)
17. Mishra, S.K., Puthal, D., Sahoo, B., Jena, S.K., Obaidat, M.S.: An adaptive task allocation technique for green cloud computing. J. Supercomput. **74**(1), 370–385 (2018)
18. Yuan, H., Zhou, M., Liu, Q., Abusorrah, A.: Fine-grained resource provisioning and task scheduling for heterogeneous applications in distributed green clouds. IEEE/CAA J. Automat. Sin. **7**(5), 1380–1393 (2020)
19. Abualigah, L.M., Diabat, A.: A novel hybrid antlion optimization algorithm for multi-objective task scheduling problems in cloud computing environments. Clust. Comput. **24**(1), 205–223 (2021)

20 Shu, W., Cai, K., Xiong, N.N.: Research on strong agile response task scheduling optimization enhancement with optimal resource usage in green cloud computing. Future Generat. Comput. Syst. **124**, 12–20 (2021)

21. Peng, Y., Jolfaei, A., Yu, K.: A novel real-time deterministic scheduling mechanism in industrial cyber-physical systems for energy internet. IEEE Trans. Industr. Inf. (2021). https://doi.org/10.1109/TII.2021.3139357
22. Yang, L., Yu, K., Yang, S.X., Chakraborty, C., Liu, Y., Guo, T.: An intelligent trust cloud management method for secure clustering in 5G enabled internet of medical things. IEEE Trans. Industr. Inf. (2021). https://doi.org/10.1109/TII.2021.3128954
23. Wang, D., He, Y., Yu, K., Srivastava, G., Nie, L., Zhang, R.: Delay sensitive secure NOMA transmission for hierarchical HAP-LAP medical-care IoT networks. IEEE Trans. Industr. Inf. (2021). https://doi.org/10.1109/TII.2021.3117263

24 Ding, F., Yu, K., Gu, Z., Li, X., Shi, Y.: Perceptual enhancement for autonomous vehicles: restoring visually degraded images for context prediction via adversarial training. IEEE Trans. Intell. Transport. Syst. (2021). https://doi.org/10.1109/TITS.2021.3120075

25 Tan, L., Yu, K., Shi, N., Yang, C., Wei, W., Lu, H.: Towards secure and privacy-preserving data sharing for COVID-19 medical records: a Blockchain-empowered approach. IEEE Trans. Netw. Sci. Eng. (2021). https://doi.org/10.1109/TNSE.2021.3101842

26. Lu, Y., Sun, N.: An effective task scheduling algorithm based on dynamic energy management and efficient resource utilization in green cloud computing environment. Clust. Comput. **22**(1), 513–520 (2019)
27. Dhiman, G., Kumar, V.: Emperor penguin optimizer: a bio-inspired algorithm for engineering problems. Knowl. Based Syst. **159**, 20–50 (2018)

28 Yang, J., Gao, H.: Cultural emperor penguin optimizer and its application for face recognition. Math. Problems Eng. (2020). https://doi.org/10.1155/2020/9579538

29 Karuppasamy, M., Balakannan, S.P.: Energy saving from cloud resources for a sustainable green cloud computing environment. J. Cyber Secur. Mobil. **7**, 95–108 (2018)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Romany F. Mansour Received the B.Sc. degree in Computer Science in 1998 and the M.Sc. in computer science in 2006, from Assiut University, Egypt. And the Ph.D. degree from the University of Assiut in 2009, He is currently working as an Associate Professor at Department of Mathematics, Faculty of science, New Valley University, Egypt. His research interests include Pattern Recognition, Computer Vision, Computer Networks, Soft Computing, Image Processing, Evolutionary Computation and Machine Learning.

Hesham Alhumyani is working in Department of Computer Engineering in College of Computers and Information Technology. Taif University at Taif Saudi Arabia. He has appointed as faculty Dean in 2019. He has obtained his Ph.D. Degree from University of Connecticut Storrs, USA. His research interests are Wireless Sensing, IoT (Internet of Things) and Cloud computing. Dr. Hesham Alhumyani has published many research papers in distinctive journals and conferences.

Sayed Abdel Khalek received Ph.D. degree in Computer Science in 2016 from Azhar University. His research interests include different directions in quantum information and computer sciences. He is the author of several articles published in different international scientific journals and is a member of different working groups. He is a Full Professor, of Applied Mathematics, Mathematics Department, Faculty of Science, Sohag University, Egypt. Also, he is an Associate Professor, of Applied Mathematics, Mathematics Department, Faculty of Science, Taif University, Taif, Saudi Arabia.

Rashid A. Saeed (M'02–SM'07) received his Ph.D. in Communications and Network Engineering, University Putra Malaysia (UPM). Currently he is a professor in Computer Engineering Department, Taif University. He was also working in Electronics Department, Sudan University of Science and Technology (SUST) since 1999. He was senior researcher in Telekom Malaysia^{TM} Research and Development (TMRND) and MIMOS. Rashid published more than 150 research papers, books and book chapters on wireless communications and networking in peer-reviewed academic journals and conferences. His areas of research interest include computer network, cognitive computing, computer engineering, wireless broadband, WiMAX Femtocell. He is successfully awarded 3 U.S patents in these areas. He supervised more 50 MSc/PhD students. Rashid is a Senior member IEEE, Member in IEM (I.E.M), SigmaXi, and SEC.

Deepak Gupta received a B.Tech. degree in 2006 from the Guru Gobind Singh Indraprastha University, Delhi, India. He received M.E. degree in 2010 from Delhi Technological University, India and Ph.D. degree in 2017 from Dr. APJ Abdul Kalam Technical University (AKTU), Lucknow, India. He has completed his Post-Doc from National Institute of Telecommunications (Inatel), Brazil in 2018. He has co-authored more than 145 journal articles including 109 SCI papers and 45 conference articles. He has authored/edited 53 books, published by IEEE-Wiley, Elsevier, Springer, Wiley, CRC Press, DeGruyter and Katsons. He has filled four Indian patents. He is convener of ICICC, ICDAM & DoSCI Springer conferences series. Currently he is Associate Editor of Expert Systems (Wiley), and Intelligent Decision Technologies (IOS Press). He is the recipient of 2021 IEEE System Council Best Paper Award. He has been featured in the list of top 2% scientist/researcher database in the world [Table-S7-singleyr-2019]. He is also working towards promoting Startups and also serving as a Startup Consultant. He is also a series editor of "Elsevier Biomedical Engineering" at Academic Press, Elsevier, "Intelligent Biomedical Data Analysis" at De Gruyter, Germany, "Explainable AI (XAI) for Engineering Applications" at CRC Press. He is appointed as Consulting Editor at Elsevier.