Intelligent Automation & Soft Computing

Tech Science Press

DOI: 10.32604/iasc.2023.034247
Article

Improved Harris Hawks Optimization Algorithm Based Data Placement Strategy for Integrated Cloud and Edge Computing

V. Nivethitha* and G. Aghila

National Institute of Technology Puducherry, Karaikal, 609609, India
*Corresponding Author: V. Nivethitha. Email: nivethithavphd@gmail.com
Received: 11 July 2022; Accepted: 04 February 2023

Abstract: Cloud computing is considered to facilitate a more cost-effective way to deploy scientific workflows. The individual tasks of a scientific workflow necessitate a diversified number of large states that are spatially located in different datacenters, thereby resulting in huge delays during data transmission. Edge computing minimizes the delays in data transmission and supports the fixed storage strategy for scientific workflow private datasets. Therefore, this fixed storage strategy creates huge amount of bottleneck in its storage capacity. At this juncture, integrating the merits of cloud computing and edge computing during the process of rationalizing the data placement of scientific workflows and optimizing the energy and time incurred in data transmission across different datacentres remains a challenge. In this paper, Adaptive Cooperative Foraging and Dispersed Foraging Strategies-Improved Harris Hawks Optimization Algorithm (ACF-DFS-HHOA) is proposed for optimizing the energy and data transmission time in the event of placing data for a specific scientific workflow. This ACF-DFS-HHOA considered the factors influencing transmission delay and energy consumption of data centers into account during the process of rationalizing the data placement of scientific workflows. The adaptive cooperative and dispersed foraging strategy is included in HHOA to guide the position updates that improve population diversity and effectively prevent the algorithm from being trapped into local optimality points. The experimental results of ACF-DFS-HHOA confirmed its predominance in minimizing energy and data transmission time incurred during workflow execution.

Keywords: Edge computing; cloud computing; scientific workflow; data placement; energy of datacenters; data transmission time

1 Introduction

In general, scientific applications are considered to be data and computation-intensive as they compose hundreds of correlated tasks [1]. The large datasets and complex structure inherent in a scientific workflow necessitate strict requirements over the capacity of storage in the deployment scenario [2]. However, the scientific workflow implemented in the environment, results in the wastage

of more amount of resources. In this context, the cloud computing organizes different virtualized resources available in diversified geographic locations into a pool of resources [3,4]. The potential characteristics of cloud computing such as maximized customizable features, scalability, flexibility and efficiency facilitate a better cost-efficient method for implementing scientific workflows [5]. But, there is a possibility of serious data transmission delays in cloud computing [6]. At this juncture, edge computing resources are generally deployed in very close proximity, which has the feasibility of minimizing the delay in data transmission and introducing more impact on the protection of private datasets [7]. Hence, the merits of cloud computing and edge computing need to be integrated to rationalise the process of data placement associated with a scientific workflow to minimize delays in data transmission in an efficient manner. In the integrated environment, edge computing is specifically responsible for guaranteeing the privacy datasets security for each of the scientific workflow.

In general, Optimization algorithms represent search methods that target determining the solution to an optimization problem. The problem of data placement is an NP-hard optimization problem. The classical data placement strategies contributed to processing scientific workflow mainly utilized evolutionary algorithms [8] to process scientific applications with the process of optimally mapping datasets to data centers. Integrated Genetic algorithm operators and self-adaptive discrete particle swarm optimization algorithm-based data placement policy (GAO-DPSO-DPS) proposed [9] for the data transmission time optimization process incurred in executing workflows. This DPSO scheme inherited the mutation and crossover operations of GA to prevent premature convergence present in the classical PSO algorithm. A differential Evolution-improved DPSO-based data placement strategy was proposed [10] for deriving the merits of edge-cloud computing to achieve better processing of scientific workflows. This DEDPSO-DPS was proposed to handle the process of placing data from the shared datasets into single and multiple workflows situated in geographically varying situations. The Genetic particle swarm optimization (GPSO)-based data placement strategy proposed [11] to utilise the merits of edge and cloud computing that aided in better processing of scientific workflows. This data placement optimized the performance of the model based on the better convergence capability of PSO and exploration capability attributed to the mutation and crossover operations. The simulation experiments of GPSO-DPS conducted using real-world scientific workflow confirmed its superiority in minimizing the data placement costs in an edge-cloud environment.

In this paper, energy and time-driven data placement using Adaptive Cooperative Foraging and Dispersed Foraging Strategies-based HHOA is proposed to reduce the energy utilized during data placement and minimise total data transmission time under the scientific workflow execution.This proposed ACF-DFS-HHOA scheme considered the influential factors of energy consumed by data centers during data placement, delay in data transmission, bandwidth established between dataceneters, the capacity of storage associated with the edge data centers and several edge data centers into account during the event of scientific workflow processing.

2 Problem Definition of ACF-DFS-HHOA-Based Data Placement Strategy

The core objective of this data placement strategy for a specific scientific workflow concentrates on attaining minimum energy utilization and minimum data transmission under the constraints of each datacentre storage capacity and energy threshold of the datacentre.

This problem definition presents an integrated environment that integrates cloud and edge computing, a data placement strategy and a scientific workflow. This integrated environment $DC_{EC} = \{DC_{Edge}, DC_{Cloud}\}$ comprises closely located edge computing and remotely located cloud computing. In this context, the edge computing environment $DC_{Edge} = \{dc_{e(1)}, dc_{e(2)}, \ldots, dc_{e(m)}\}$ and cloud computing

$DC_{Cloud} = \{dc_{c(1)}, dc_{c(2)}, \ldots, dc_{c(n)}\}$ is considered to possess m and n datacenters, respectively. In this implementation environment, only the storage capacity and energy availability of the datacenters are only considered, rather than considered its capacity of computing. At this juncture, every datacenter ($DC_{EC(i)}$) independent of cloud and edge computing is presented in Eq. (1) as,

$$DC_{EC(i)} = \{DC_{EC(Capacity)}, DC_{EC(Type)}, DC_{EC(Energy)}\} \tag{1}$$

where, $DC_{EC(Capacity)}$ represents the capacity of storage associated with each datacentre ($DC_{EC(i)}$) and the datasets stored in any data center should not exceed their capacity level. Further, $DC_{EC(Type)}$ refers to the data center location with respect to the cloud and edge computing. The value $DC_{EC(Type)}$ is set to 0 when the dataset pertains to the cloud environment and is capable of storing only the public datasets. In contrast, the value $DC_{EC(Type)}$ is set to 1 when the datacenter corresponds to edge computing and capable of strengthening both public and private datasets.

Then, the bandwidth between the datacenters of the integrated environment is highlighted in Eqs. (2) and (3)

$$BW_{DC(ij)} = \{Band_{EC(ij)}, DC_{EC(Type(i))}, DC_{EC(Type(j))}\} \tag{2}$$

$$\mathrm{BW_{DC(ij)}} = \begin{bmatrix} \mathrm{BW_{11}} & \mathrm{BW_{12}} & \ldots\ldots & BW_{1m} \\ BW_{21} & \mathrm{BW_{22}} & \ldots\ldots & BW_{2m} \\ \ldots\ldots & \ldots\ldots & \ldots\ldots & \ldots\ldots \\ \mathrm{BW_{q1}} & \mathrm{BW_{q2}} & \ldots\ldots & BW_{qm} \end{bmatrix} \tag{3}$$

where, $BW_{DC(ij)}$ depicts the bandwith established between the datacenters $DC_{EC(i)}$ and $DC_{EC(j)}$, respectively. In addition, the bandwidth established between any two datacenters is considered to be known in advance and does not fluctuate during the implementation process.

Then, the energy consumed by the datacenters during the process of data placement is derived from [12] and is represented in Eq. (4)

$$DC_{Energy(i)} = C_{stable} * E_{max} + (1 - k) * E_{max} * u \tag{4}$$

The proposed model is that energy consumed by a server grows linearly with the increase of its CPU utilization, including the amount of energy spent in the idle state up to the energy consumed when the server is utilized fully. This linear relationship can be represented as shown in the equation where, $DC_{Energy(i)}$ is the estimated energy consumption of the selected server, C_{stable} is energy wasted by an idle server, E_{max} is the energy consumed by a server when it is fully utilized and is the CPU utilization at time t.

The total energy utilized by a dedicated server (DC_{Edge}) for a time period can be precisely determined as an integral of the power consumption function over a specific time period as shown in Eq. (5)

$$E = \int_{t_0}^{t_1} DC_{Energy}\left(u\left(t\right)\right).dt \tag{5}$$

To simulate the performance of the mentioned linear power model, the total energy consumption for edge network servers (E_{Edge}) metric. It is caused by running the scientific workloads and is calculated as shown in Eq. (6)

$$E_{Edge} = \sum_{j=1}^{n} DC_{Edge(i)} \tag{6}$$

In this integrated environment, a specific scientific workflow $SW = (T, E, DS_{EC})$ is depicted as a directed acyclic graph, where 'T' and 'E' representing the collection of nodes and edges. The set of nodes $T = \{t_1, t_2, \ldots, t_k\}$ representing the scientific workflow, in turn, consists of 'k' tasks. The edges $E = (e_{12}, e_{13}, \ldots, e_{ij})$ represents the data associations existing between each and every pair of tasks. In this context, the datasets $DS_{EC} = \{ds_1, ds_2, \ldots, ds_n\}$ highlights the complete set of datasets residing in a particular scientific workflow. An edge $e_{ij} = (t_i, t_j)$depicts the data correlation between any two tasks t_i and t_j respectively. In this case, the task t_i is considered the direct predecessor of the task t_j. Moreover, a task cannot start its execution unless all of its predecessor tasks have been finished executing in the process of scheduling a scientific workflow. For every task t_i, $IP_{DS(i)}$ and $OP_{DS(i)}$ indicates the input and output datasets corresponding to that t_i task. In addition, the association between the dataset and the set of tasks is determined to be many-to-many since one task may necessitate multiple numbers of datasets, and every single data may utilize the multiple numbers of tasks. Every dataset independent of being input or output datasets comprises of three parameters such as dataset size, task generating datasets and dataset original storage location presented in Eqs. (7) and (8)

$$DS_{GT(i)} = \begin{cases} 0 & ds_{(i)} \in DS_{init} \\ T_{ds(i)} & ds_{(i)} \notin DS_{Init} \end{cases} \tag{7}$$

$$SL_{OR(ds(i))} = \begin{cases} 0 & ds_{(i)} \in DS_{Init} \\ L_{FP(ds(i))} & ds_{(i)} \notin DS_{Init} \end{cases} \tag{8}$$

In this situation, the complete datasets can be partitioned into generated datasets and initial datasets based on their data sources. The generated datasets and the initial datasets are considered the intermediate and input datasets determined during scientific workflow execution. In (7), $T_{DS(i)}$ depicts the task generated from the dataset ds_i. Furthermore, the datasets also can be partitioned into flexible (public datasets) and private datasets (fixed datasets). At this juncture, $SL_{OR(ds(i))}$ and $DC_{GT(i)}$ represents the data center that stores the private dataset into its edge computing environment and private datasets that can be located in edge data centers.

The core objective of the proposed ACF-DFS-HHOA-based data placement strategy completely concentrates on reducing the time incurred in data transmission by satisfying the complete set of requirements essential during the execution of workflows. In a workflow, every individual task under execution needs to meet the following two constraints. i) The input datasets essential for a specified task already exist in a specified data center, and ii) the task need to be allocated to each particular data center. The proposed ACF-DFS-HHOA-based data placement strategy concentrates on reducing the total data transmission time. As a result, the tasks scheduling time associated with data centers is always much lower than the total datasets transmission time from datacenters. In this context, the time incurred in data transition from one datacenter to the other data center is determined based on Eq. (9)

$$DT_{Transfer}(DC_{EC(i)}, DC_{EC(j)}, ds_{size(t)}) = \frac{ds_{size(t)}}{Band_{DC(ij)}} \tag{9}$$

Then, the total data transmission time incurred during the data placement during the process of executing a particular scientific workflow is presented in Eq. (10)

$$DT_{TOTAL} = \sum_{i=1}^{|DS_{EC}|} \sum_{j \neq i}^{|DS_{EC}|} \sum_{k=1}^{|DS_{size(i)}|} DT_{Tranfer}\left(DC_{EC(i)}, DC_{EC_{(j)}} ds_{sixe(t)}\right) \tag{10}$$

Thus, this ACF-DFS-HHOA-based data placement strategy is formulated as the energy and time-driven solution for optimal execution of scientific workflow that integrates the environment of edge and cloud computing as formalized in Eq. (11)

Minimize DT_{Total} and $TE_{EC(util)}$ (11)

Subject to $\forall i, \sum_{i=1}^{|DC_{EC(i)}|} DS_{EC(j)}, v_{ij} \leq DC_{capacity}$ and $\forall i, \sum_{i=1}^{|DC_{EC(i)}|} DC_{E(j)}, e_{ij} \leq TE_{Avail\_Threshold}$

Hence, the core objective of this ACF-DFS-HHOA-based data placement strategy focused on attaining minimum energy utilization in the data center and reduced data transmission time by complying with the constraints of storage capacity constraint associated with each data center.

3 Acf-Dfs-Hhoa-Based Data Placement Strategy

The main objective of the data placement strategy concentrates on the determination of superior mapping from datasets (DS_{SW}) to datacentres (DC_{EC}), such that the minimized data transmission time and the reduced energy consumption is achieved. This process of estimating the superior mapping between DS_{SW} and DC_{EC} is considered as an NP-hard problem. Hence, this ACF-DFS-HHOA-based data placement strategy is proposed from the global dimension point of view. The detailed view of the pre-processing and ACF-DFS-HHOA Algorithm is presented as follows.

3.1 Pre-processing Over a Scientific Workflow

In the pre-processing step of the scientific workflow, each cut-edge dataset is merged into a new dataset. In this context, the cut-edge dataset refers to the dataset in which there exist two neighbourhood ($ds_{set(i)}$ and $ds_{set(j)}$) datasets with at least public dataset and that only shares only one common one between them. Moreover, the in-degree and out-degree of $ds_{set(i)}$ need to be equal to 1, with only one task shared between $ds_{set(i)}$ and $ds_{set(j)}$.

3.2 HHOA Algorithm

The HHOA algorithm was proposed based on the inspiration derived from the foraging and attack behaviour of Harris Hawk [13]. The position of the hawk (search agent) depending on the prey position (PR_{Pos}) is determined based on Eq. (12)

$$\mathrm{HH}\,(\mathrm{t}+1) = \begin{cases} \mathrm{HH_{Rand}}\,(t) - \mathrm{r_1}(\mathrm{HHt} - 2\mathrm{r_2}\ \mathrm{HH(t)}), & \text{if } \mathrm{q_{Rand}} \geq 0.5 \\ (\mathrm{HH_{DBP}}\,(\mathrm{t}) - \mathrm{HH_{Mean}}\,(\mathrm{t})) - \mathrm{r_3}\,(\mathrm{LT} + \mathrm{r_4}\,(\mathrm{UT} - \mathrm{LT}))\,, & \text{If } \mathrm{q_{Rand}} < 0.5 \end{cases} \tag{12}$$

where, $HH_{Rand}(t)$ depicts the random position of the Harris hawk search agents in the current iteration. q_{Rand} is the value of randomness that exists between the range 0 and 1. Moreover, r_1, r_2, r_3 and refers to the four significant random numbers that also ranges between 0 and 1. In this context, LT and UT highlights the lower and upper threshold. This random number aids in better exploration of the hawk agents in the search space of the population. The mean position (HH^t_{Mean}) associated with the complete set of hawk agents are determined based on Eq. (13)

$$HH^t_{Mean} = \frac{1}{P_{Size}} \sum_{i=1}^{P_{Size}} HH^t_{(i)} \tag{13}$$

where, $HH^t_{(i)}$ pertains to the Harris hawk position in the 't' iteration with 'P_{Size}' related to the population size.

In the second stage of HHOA, the transformation from the exploration and exploitation completely depends on the prey escaping energy highlighted in Eq. (14)

$$P_{EE} = P_{EE(0)}\left(1 - \frac{t}{I_{Max}}\right) \tag{14}$$

where, '$P_{EE(0)}$' and 't' depicts the initial prey escaping energy and the maximum number of iterations. At this juncture, $P_{EE(0)}$ is the random value that lies in the range [−1, 1].

In the iteration process, the algorithm is considered to be in the phase of exploration when the value of $|P_{EE}| \geq 1$. Else, the phase of exploitation is determined to be initiated. During the final phase, the search agents adopt four different strategies for determining the prey available in the implementation environment [14]. The strategies of exploitation such as hard besiege, hard besiege with the strategy of progressive rapid dive, soft besiege with the strategy of progressive rapid dive, and soft besiege are adopted depending on the value of escaping probability of target (EP_{Tr}).

This HHOA adopts the strategy of soft besiege under $|P_{EE}| \geq 1$ and $EP_{Tr} \geq 1$ for slowly converging towards the targeted solution. This strategy of soft besiege is presented in Eqs. (15) and (16)

$$HH_i^{t+1} = \Delta HH_i^t - P_{EE}\left|P_{JI}PR_{Pos(i)} - HH_{(i)}^t\right| \tag{15}$$

where,

$$\Delta HH_i^t = \left|PR_{Pos(i)} - HH_{(i)}^t\right| \tag{16}$$

At this juncture, ΔHH_i^t highlights the deviation between the current individual and the target position vector. Is the target prey or target jumping intensity and is considered to possess a random value between [0, 2].

On the other hand, the target cannot escape due to the possessed energy of escape under $|P_{EE}| < 1$ and $EP_{Tr} \geq 1$, and hence adopt the strategy of soft besiege with progressive rapid dive for attaining the required targeted solution in the search domain as presented in Eq. (17)

$$HH_i^{t+1} = PR_{Pos} - P_{EE}\left|\Delta HH_{(i)}^t\right| \tag{17}$$

when the condition $|P_{EE}| \geq 1$ and $EP_{Tr} < 1$ is satisfied, the target solution has the probability of successfully escaping from the search agent as the energy possessed by them is maximized. Hence, the search agents employ soft besiege with progressive rapid dive for determining the target solution as specified in Eq. (18)

$$\mathrm{S}(\mathrm{c}+1) = \begin{cases} \mathrm{Y}_{\mathrm{XP},} \text{ if } \mathrm{F}(\mathrm{Y}_{\mathrm{XP}}) < \mathrm{F}(\mathrm{S}(\mathrm{c})) \\ \mathrm{Z}_{\mathrm{XP}}, \text{ if } \mathrm{F}(\mathrm{Z}_{\mathrm{XP}}) < \mathrm{F}(\mathrm{S}(\mathrm{c})) \end{cases} \tag{18}$$

when the condition $|P_{EE}| \geq 1$ and $EP_{Tr} < 1$ is satisfied, the target solution has the probability of successfully escaping from the search agent as the energy possessed by them is maximized. Hence, the search agents employ soft besiege with progressive rapid dive for determining the target solution as specified in Eq. (19)

$$\mathrm{S}(\mathrm{c}+1) = \begin{cases} \mathrm{Y}_{\mathrm{XP},} \text{ if } \mathrm{F}(\mathrm{Y}_{\mathrm{XP}}) < \mathrm{F}(\mathrm{S}(\mathrm{c})) \\ \mathrm{Z}_{\mathrm{XP}}, \text{ if } \mathrm{F}(\mathrm{Z}_{\mathrm{XP}}) < \mathrm{F}(\mathrm{S}(\mathrm{c})) \end{cases} \tag{19}$$

The primitive HHOA is utilized for solving the problem of continuous optimization. ACF-DFS-HHOA is contributed to solving the issues as mentioned earlier. The ACF-DFS-HHOA-based data placement strategy is explained as follows.

Encoding Problem: In this encoding problem, the candidate solution related to the problem space is encoded as the targeted prey of the HHOA search agent. The candidate solution of the problem space possesses only a single encoded targeted prey. In the problem space, each candidate solution refers to an individual encoded targeted prey In HHOA [15]. In the proposed approach, a discrete encoding methodology was utilized for generating targeted prey candidate solutions of n–dimensions. In this context, a targeted prey candidate solution represents a data placement solution associated with a specific scientific workflow that integrates theud and edge computing. Then, each individual targeted prey candidate solution in the i^{th} iteration is depicted in (20)

$$PR^{t}_{Pos(i)} = (pr^{t}_{i1}, pr^{t}_{i2}, ..., pr^{t}_{in}) \tag{20}$$

where, 'n' represents the datasets count after the step of pre-processing with each individual targeted prey candidate solution is considered as an integer-valued vector of 'k' dimensions. In this case, $PR_{Pos(ik)}$ $(1 \leq k \leq n)$ represents the final placement location of the k^{th} dataset in the i^{th} iteration. This $PR_{Pos(ik)}$ $(1 \leq k \leq n)$ also portrays the number of datacentre $PR_{Pos(ik)} = \{1, 2, 3, \ldots, |DC_{EC}|\})$. Moreover, the private dataset storage location is fixed and never changed during mapping datasets with datacenters.

Fitness Function: In this proposed ACF-DFS-HHOA, the fitness function plays an anchor role in evaluating the merits and limitations of a targeted prey candidate solution. In this case, a targeted prey candidate solution with minimum fitness value is considered to attribute better performance as the complete objective of this proposed scheme concentrates on reducing the energy of datacenters and minimizing the time of transmission during the time of data placement during the processing of a scientific workflow. Thus, the fitness function is equal to the time incurred during data placement strategy during scientific processing. In this proposed scheme, the fitness function is defined depending on the different situations that are feasible during the mapping of datasets to the datacenters.

Case i) When the two compared targeted prey candidate solution are feasible, then the targeted prey candidate solution with smaller energy utilization in data centers and minimizing time of transmission is selected as the best optimal solution. In this setting, the fitness function of the targeted prey candidate solution is defined in Eq. (21)

$$Fit_{Val} = Min\_DC_{Energy}\_DT_{Time}(PR^{t}_{Pos(1)}, PR^{t}_{Pos(2)}) \tag{21}$$

Case ii) When the two compared targeted prey candidate solution are infeasible, then the targeted prey candidate solution with smaller energy utilization in data centers and minimizing time of transmission is again selected as the best optimal solution as highlighted in Eq. (18). However, an infeasible targeted prey candidate solution can get transformed into a feasible targeted prey candidate solution after the application of update operation attained through the strategy of Adaptive Cooperative and Dispersed Foraging. Even after this transformation, the targeted prey candidate solution with smaller energy utilization in data centers and minimizing transmission time is selected as the best optimal solution.

Case iii) Out of the two compared targeted prey candidate solution, if one solution is feasible and the other solution is infeasible, then the feasible targeted prey candidate is selected, and the fitness function is defined based on the Eq. (22)

$$\text{Fit} = \begin{cases} 0, \text{ if } \forall \text{i}, \sum_{i=1}^{DS} ds, \text{u} < DC_{capacity}, \text{E.Util} < \text{DC}(\text{E}_{\text{Thres}}) \\ 1, \text{ Otherwise} \end{cases} \tag{22}$$

3.3 Adaptive Cooperative and Dispersed Foraging-Based Update Strategy

From the literature, it is evident that primitive HHOA is successful during its application in most of the diversified practical optimization problems. However, the primitive HHOA possess several shortcomings. A new integrated HHOA with Adaptive Cooperative and Dispersed Foraging strategies was determined to be the potential technique that aids in a better tradeoff between the rate of exploitation and exploration. This Adaptive Cooperative and Dispersed Foraging strategies-based HHOA is included for better mapping of datasets to datacenters with minimized datacenters energy and reduced data transmission time.

3.3.1 Strategy of Adaptive Cooperative Foraging

In the traditional HHOA, the position of the targeted prey candidate solution (PR_{Pos}) is updated by the Harris hawk search agents. But, this updating process has the possibility of ignoring an optimal solution in the scope of individuals that possess poor fitness, thereby leading to the possibility of premature convergence. In this Adaptive Cooperative Foraging strategy, three individual search agents are randomly selected as the search agent of guidance in the search space. This strategy included the mean distance computed between the guided search agents and search agents as the factor of search step size for joint support in the search process. The new update formula corresponding to the adaptive cooperative foraging process is presented in Eq. (23)

$$\mathrm{zHH}\,(\mathrm{t}+1) = \begin{cases} \mathrm{HH_{Rand}}\,(t) - \mathrm{r_1}(\mathrm{HHt} - 2\mathrm{r_2}\ \mathrm{HH(t)}, & \text{if } \mathrm{q_{Rand}} \geq 0.5 \\ \left(\mathrm{HH_{(i)}}\,(\mathrm{t})\right) - \mathrm{r_3}\left(\frac{(HH\ Rand1 - HH(t)) + (HH\ Rand2 - HH(t)) + (HH\ Rand3 - HH(t))}{3}\right), & \text{If } \mathrm{q_{Rand}} < 0.5 \end{cases} \tag{23}$$

In the new update formula, the second line represents the new search equation which is completely different from the guidance of a single optimal position inherited in the primitive algorithm. Further, the guiding search agents used in this Adaptive Cooperative Foraging strategy are completely a random vector independent of their good or bad quality. Thus, this randomness offered by the adaptive foraging strategy prevented the limitations of primitive HHOA that ignored the optimized solution that lies near the individual solution with the worst fitness.

3.3.2 Dispersed Foraging Strategies

The unavailability of better-targeted prey candidate solution during searching them by the search agents may induce them to relocate the searching region to explore most potential regions. However, this relocation of search agents may fail in the exploration process. Thus, the Dispersed Foraging strategies are included to prevent the shortcoming as mentioned earlier by utilising a factor of dispersion termed CF_{ε}. In this strategy, the individual targeted prey candidate solution that satisfied the conditions of dispersion is used for improving the position update operations. The equation of position update used in the process of dispersed foraging is presented in Eqs. (24) and (25)

$$PR^{t+1}_{Pos(i)} = PR^{t}_{Pos(i)} + \alpha_{mc} * D^{t}_{SA(i)} * L^{t}_{V(i)} \tag{24}$$

$$D^{t}_{SA(i)} = \left(PR^{t}_{Pos(r1)} - PR^{t}_{Poa(r2)}\right) \tag{25}$$

where, α_{mc} is the migration coefficient that satisfies $\alpha_{mc} \sim N(0.5, 0.12)$ and the value of this parameter is set inconsistency with that in literature. $D^{t}_{SA(i)}$ and $L^{t}_{V(i)}$ represent the distance between the considered two search agents and logical value that is used for identifying the possibility of including dispersed foraging strategy in the classical HHOA as defined in Eq. (26)

$$L^{t}_{V(i)} = \begin{cases} 1 & R_S > \delta \\ 0 & R_S \leq \delta \end{cases} \tag{26}$$

where 'δ' is the factor of dispersion is considered as the parameter that non-linearly decreases with every iteration based on Eq. (27)

$$\delta = \delta_0 \, exp\left(-\frac{t}{I_{Max}}\right) \tag{27}$$

In this proposed scheme, the value of δ is set to a constant value of 0.4. This factor of dispersion is considered to change adaptively with respect to an increase in the number of iterations. It is responsible for selecting only some of the search agents to facilitate the operations of dispersion. The dispersed foraging strategy increases population diversity by preventing the complete set of individual search agents from exploring unknown search regions. Moreover, the value δ is assigned to a relatively large value in the early stages, such that only a limited number of search agents perform the process of scattered foraging. Thus, the dispersed foraging strategies aids in improving the convergence rate in the early stages. But, when the value δ decreases, then the maximized number of search agents is made to perform dispersed foraging operations to prevent the local point of optimality.

3.4 The Process of Mapping Targeted Prey Candidate Solution Towards Data Placement

The process of mapping targeted prey candidate solution towards data placement for a particular scientific workflow comprises inputs that include a scientific workflow $SW = (T, E, DC_{EC})$, the encoded candidate solution and datacenters DC_{EC}. Algorithm 1 presents the mapping process of targeted prey candidate solution towards data placement.

Algorithm 1 Initially, the storage of all the data centres present in the edge-cloud environment currently is set to 0, and the time incurred in data transmission and energy utilized in the data center during data placement is also set to 0 initially. After initialization, the datasets are stored in the respective data centers, and the present storage capacity available with each data center is recorded. In this situation, when any edge datacenters storage capacity is greater than its capacity of storage, then the targeted prey candidate solution is considered as infeasible and delivered as output. Then, based on the task execution sequence, the tasks are placed in a specified data center that satisfied the constraints of minimum energy utilized and reduced transmission time. Moreover, If the cumulative sum of input datasets and output datasets corresponding to a task and the current storage capacity of datacenters is greater than the datacenters' storage capacity, then the targeted prey candidate solution is considered as infeasible and delivered as an output. On the other hand, if the targeted prey candidate solution is identified as feasible, then the calculation of data transmission time is achieved.

Further, the complete set of tasks is scanned sequentially, and the datacenters that stores the input datasets corresponding to the considered tasks are determined. Furthermore, the total transmission time is computed by superimposing the time incurred in transferring an input dataset to data centers and the associated transmission time. Finally, the strategy of data placement and the associated total transmission time are delivered as an output.

4 Simulation Results and Discussion

The experiments of the proposed ACF-DFS-HHOA Algorithm and the benchmarked approaches are conducted in a hybrid environment to simulate a real-world scenario that integrates edge and cloud computing characteristics together. The main assumptions considered in the proposed simulation environment are, i) Each datacentre concentrates on its energy and capacity of storage and completely ignores the capacity of computing, ii) The bandwidth established with the datacenters is kept constant, such that this strategy of data placement is not influenced by the fluctuations in the bandwidth. The experimental evaluation of the proposed ACF-DFS-HHOA Algorithm and the competitive

approaches are conducted using data transmission time, and energy consumption rate with large, moderate and small workflows.

5 Experimental Setup

The proposed ACF-DFS-HHOA Algorithm simulation experiments and the benchmarked approaches are conducted on the Win8 64-bit Operating System platform with an i5-7500U 2.90 GHz, Intel ®, Core ™ process of 8 GB of RAM. The population size and the maximum number of iterations are set to 100 and 1000, respectively. The experiments are achieved using partly scientific workflows such as laser interferometer gravitational-wave observatory (LIGO), Epigenomics, sRNA identification protocol using high-throughput technology (SIPHT), Montage and Cybershake. The complete details of the mentioned partly synthetic workflows are presented in [16]. An XML file is used for recording the complete information about the input, output datasets and the dependence structure associated with each type of workflows. The complete experiments of the proposed ACF-DFS-HHOA Algorithm and the benchmarked approaches are conducted using large, moderate and small scientific workflows. Moreover, the large, moderate and small scientific datasets comprise of 100 tasks, 50 tasks and 30 tasks, respectively. The implemented hybrid environment considered for experimenting consists of four data centers.

5.1 Performance Evaluation of Proposed ACF-DFS-HHOA Algorithm Based on Data Transmission Time with Large, Moderate and Small Workflows

In this experiment, the proposed ACF-DFS-HHOA data placement strategy and the benchmarked GAO-DPSO-DPS, DEDPSO-DPS and GPSO-DPS schemes are compared based on data transmission time (data transmission time is estimated as the mean of 100 repeated experiments) with large, moderate and small workflows. Fig. 1 presents the time incurred in data transmission of the proposed ACF-DFS-HHOA data placement strategy and the benchmarked algorithms. The proposed ACF-DFS-HHOA data placement strategy exhibits better performance as it is capable of the deviation existing between the pre-assigned and the actual data placement. The proposed scheme attained a reduced average data transmission time of 5.28%, 6.79% and 7.56%, superior to the benchmarked GAO-DPSO-DPS DEDPSO-DPS and GPSO-DPS schemes. Fig. 2 demonstrates the time incurred in data transmission of the proposed ACF-DFS-HHOA data placement strategy and the benchmarked GAO-DPSO-DPS, DEDPSO-DPS and GPSO-DPS schemes with moderate scientific workflows. The proposed ACF-DFS-HHOA data placement strategy reduced the average data transmission time by 6.54%, 7.42% and 8.96%, superior to the benchmarked GAO-DPSO-DPS, DEDPSO-DPS and GPSO-DPS scheme.

Algorithm 1: Mapping targeted prey candidate solution towards data placement

Procedue Data Placement ($SW, DC_{EC}, PR_{Pos(i)}$)

1: **Begin**

2: Initialize$DC_{Cap_used} \leftarrow 0$, $DT_{Time} \leftarrow 0$ and $DC_{Eneregy\_Util} \leftarrow 0$.

3: **for** each DS_{SW} of $DS_{Initial}$

4: Identify the possibility of a datacenter that is overloaded during the placement of initial datasets

5: ($DC_{Cap\_used}(PR_{Pos(i)} += DS_{Size})$) place the dataset $DS_{SW(i)}$ in the datacenter $DC_{EC}(PR_{Pos(i)})$

6: **if** ($DC_{Cap\_used} > PR_{Pos(i)}.capacity$) **then**

7: return the targeted prey candidate solution is not feasible

(Continued)

Algorithm 1: Continued

end if
end for
for each task $1 \leq j \leq |T|$ // Identify whether there exists any overloaded datacenter during the execution of tasks//
Place the task $T_{(j)}$ in datacenter $DC_{EC(k)}$ based on energy utilized and minimal data transmission time
if $(DC_{Cap\_used}(j) + Sum(IP_{DS(j)}) + Sum(OP_{DS(j)}) > T_{Cap(j)})$ **then**
return the targeted prey candidate solution is infeasible
end if
Update the current storage after placing the output datasets $OP_{DS(j)}$ related to task $T_{(j)}$ in the corresponding data center.
end for
for each task $1 \leq j \leq |T|$ // Compute the total energy consumed and the total transmission time incurred during data placement//
Determine the datacenters $DC_{EC(k)}$ that stores the input datasets $IP_{DS(j)}$ related to the task $T_{(j)}$
Estimate the time incurred in transmission from $IP_{DS(j)}$ to $DC_{EC(k)}$ and energy-based on Eqs. (3) and (4)
$(DC_{Energy} + DT_{Time)}) += Transfer(T_j)$
end for
Output the strategy of data placement and the associated energy utilized with total transmission time DT_{Time} .
End procedure.

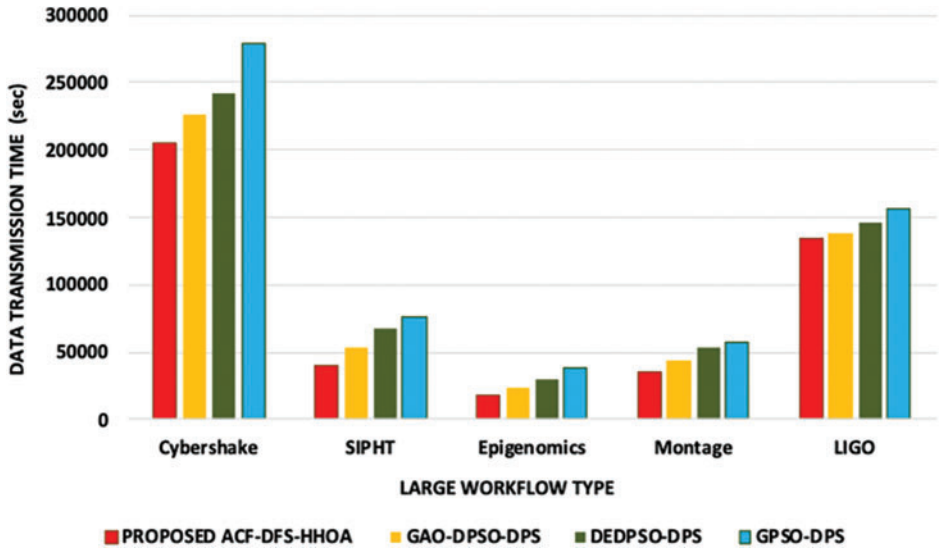

Figure 1: Proposed ACF-DFS-HHOA-Data transmission time with large workflows

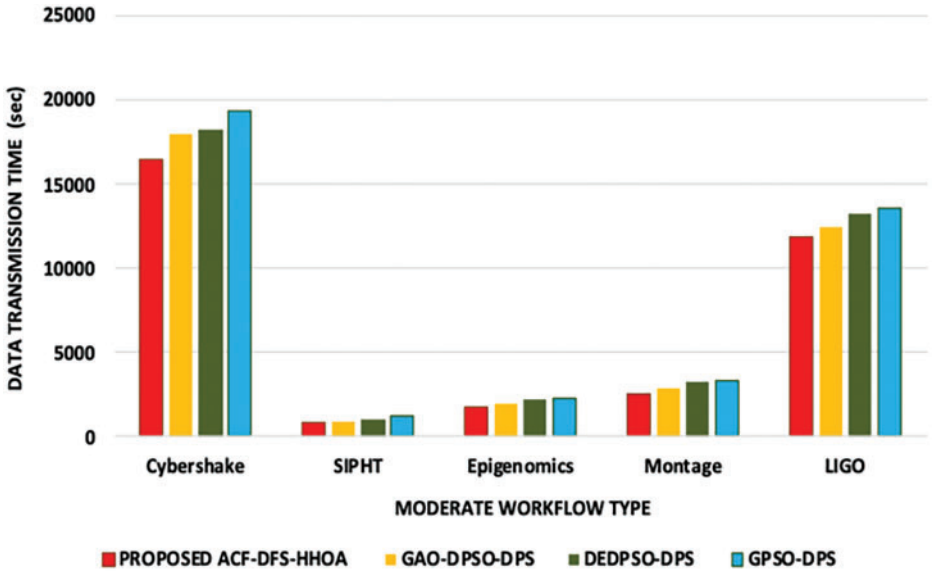

Figure 2: Proposed ACF-DFS-HHOA Data transmission time with moderate workflows

Fig. 1 presents the time incurred in data transmission of the proposed ACF-DFS-HHOA data placement strategy and the benchmarked algorithms. Fig. 2 demonstrates the time incurred in data transmission of the proposed ACF-DFS-HHOA data placement strategy and the benchmarked GAO-DPSO-DPS, DEDPSO-DPS and GPSO-DPS schemes with moderate scientific workflows. Fig. 3 portrays the proposed ACF-DFS-HHOA with small workflows reduced the average data transmission time by 7.21%, 8.59% and 9.38%, superior to the benchmarked GAO-DPSO-DPS, DEDPSO-DPS and GPSO-DPS schemes.

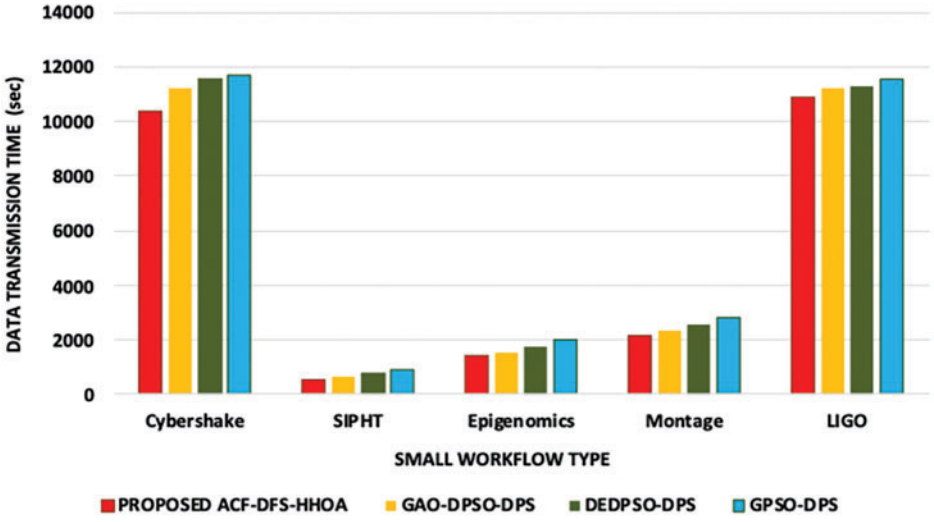

Figure 3: Proposed ACF-DFS-HHOA Data transmission time with small workflows

5.2 Performance Evaluation of Proposed ACF-DFS-HHOA Algorithm Based on Different Edge Datacenters with Moderate Workflows

In this experiment, the proposed ACF-DFS-HHOA data placement strategy and the benchmarked GAO-DPSO-DPS, DEDPSO-DPS and GPSO-DPS schemes are compared based on data transmission time with different edge datacenters and moderate number of scientific workflows. Figs. 4 and 5 present the data transmission time incurred with respect to the scientific workflows of LIGO and Cybershake under a different number of edge data centers. Figs. 6 and 7 presents the data transmission time incurred by the proposed ACF-DFS-HHOA data placement strategy and the benchmarked GAO-DPSO-DPS, DEDPSO-DPS and GPSO-DPS schemes with respect to the scientific workflows of Epigenomics and SIPHT under a different number of edge data centers.

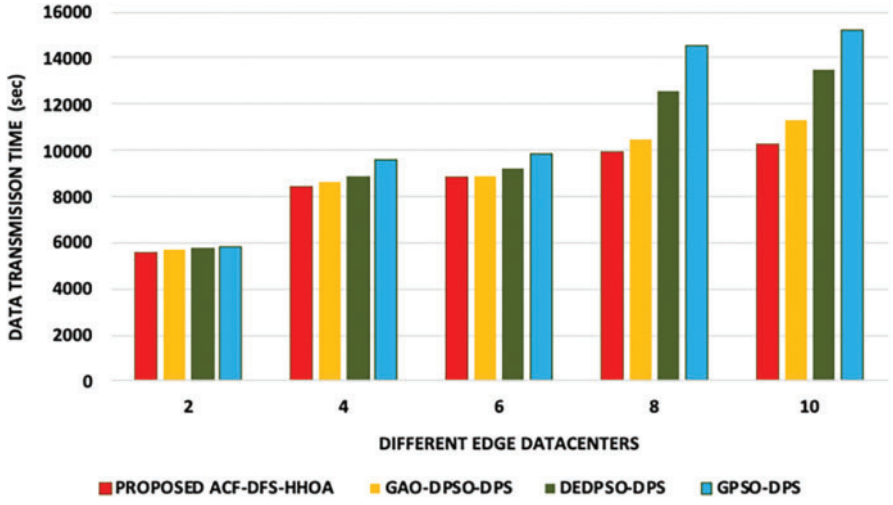

Figure 4: Proposed ACF-DFS-HHOA-Data transmission time with different edge datacentres under moderate workflows (LIGO)

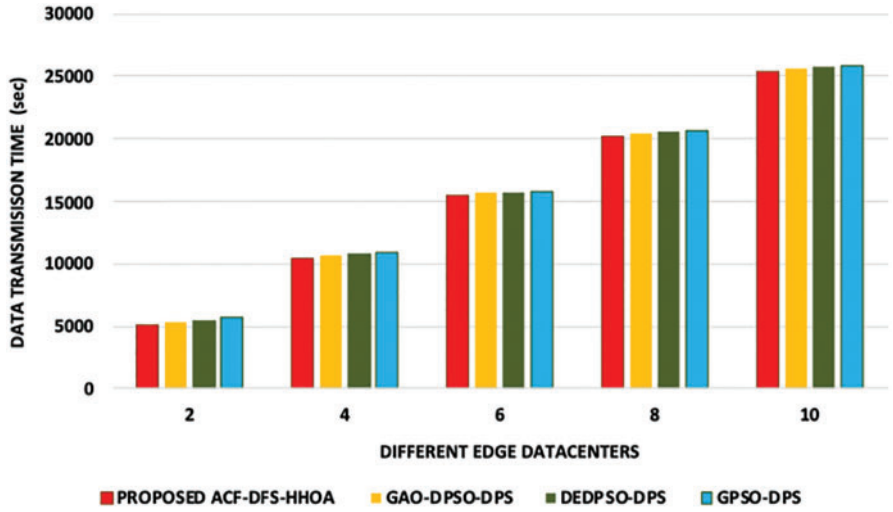

Figure 5: Proposed ACF-DFS-HHOA-Data transmission time with different edge datacentres under moderate workflows (Cybershake)

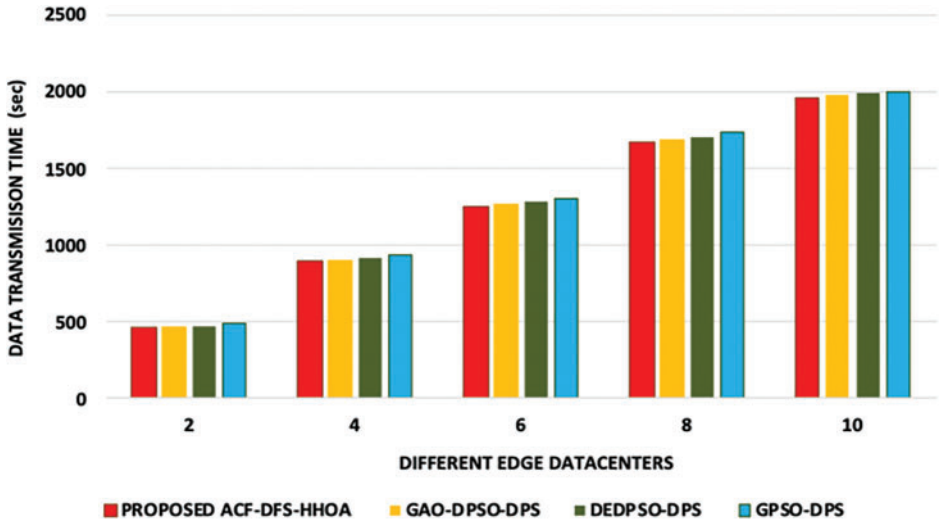

Figure 6: Proposed ACF-DFS-HHOA-Data transmission time with different edge datacentres under moderate workflows (Epigenomics)

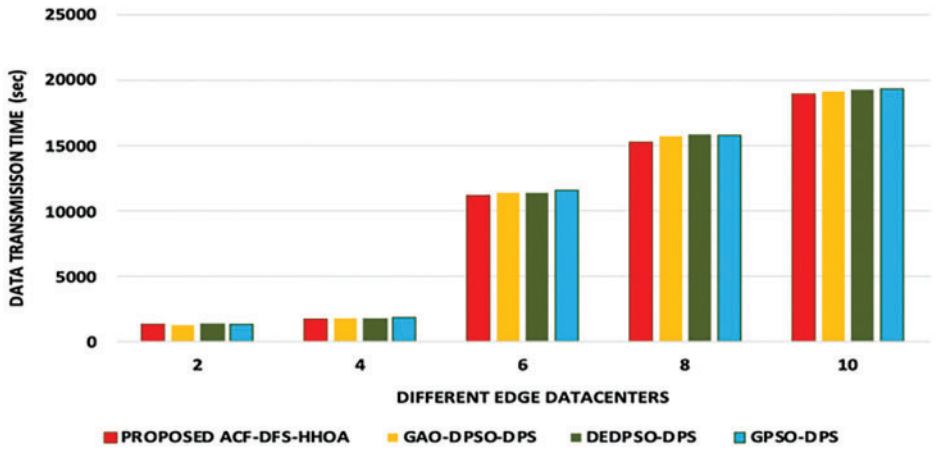

Figure 7: Proposed ACF-DFS-HHOA-Data transmission time with different edge datacentres under moderate workflows (SIPHT)

In addition, Fig. 8 highlights the data transmission time incurred by the proposed ACF-DFS-HHOA data placement strategy and the benchmarked GAO-DPSO-DPS, DEDPSO-DPS and GPSO-DPS schemes with respect to the scientific workflows of Montage under a different number of edge data centers.

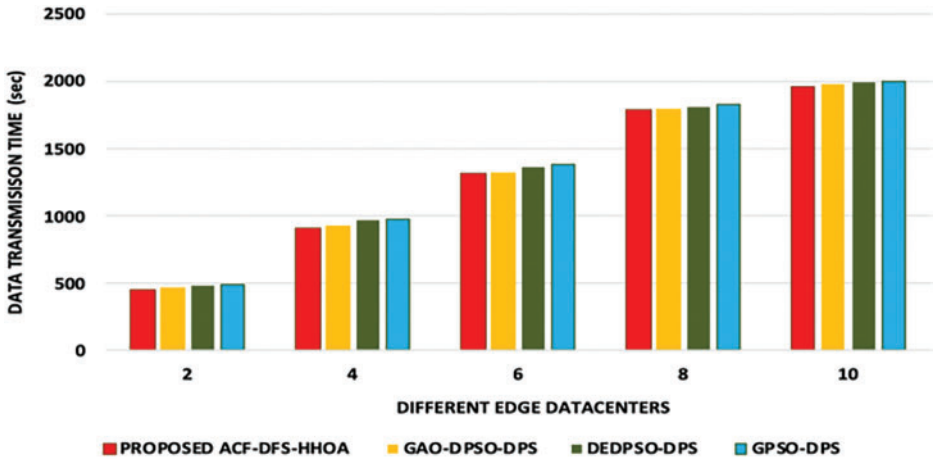

Figure 8: Proposed ACF-DFS-HHOA-Data transmission time with different edge datacentres under moderate workflows (Montage)

5.3 Performance Evaluation of Proposed ACF-DFS-HHOA Algorithm Based on Baseline Storage Capacity Multipliers with SIPHT and Epigenomics Workflows

In this part of the analysis, Figs. 9 and 10 demonstrate the proposed ACF-DFS-HHOA scheme with large workflows minimized the average energy consumption ratio by 5.68%, 6.84% and 10.32%, better than the schemes used for comparison. Moreover, the proposed ACF-DFS-HHOA scheme with moderate workflows is also confirmed to minimize the average energy consumption ratio by 6.74%, 8.68% and 11.92% benchmarked GAO-DPSO-DPS, DEDPSO-DPS and GPSO-DPS schemes.

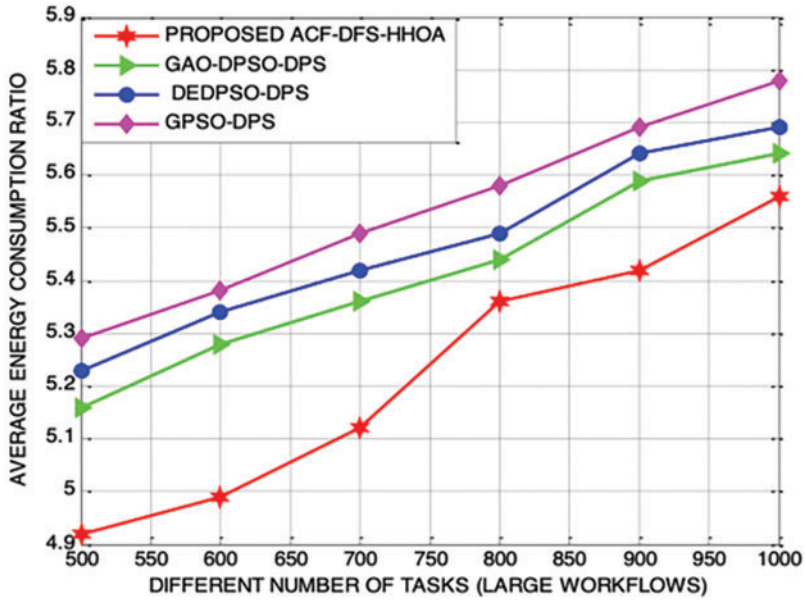

Figure 9: Proposed ACF-DFS-HHOA-Average energy consumption ratio with large workflows

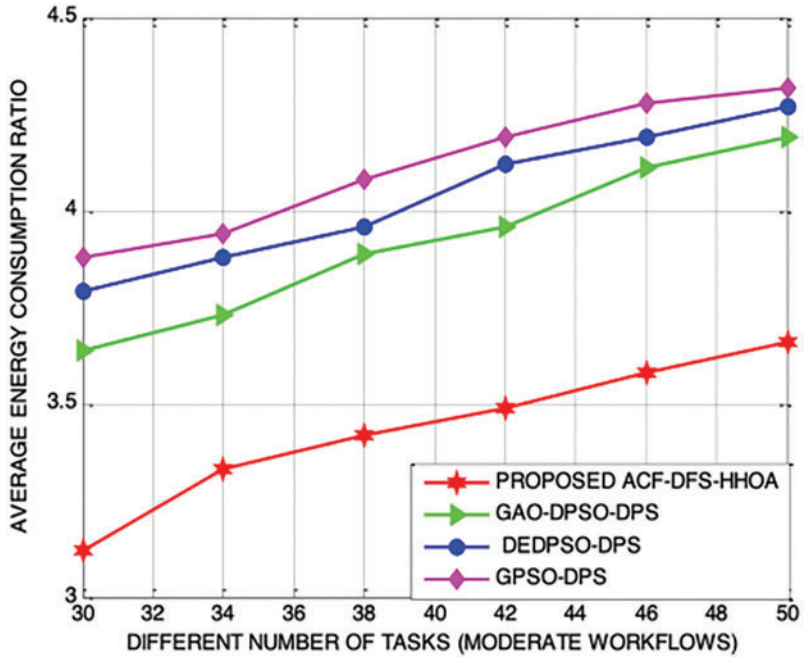

Figure 10: Proposed ACF-DFS-HHOA-Average energy consumption ratio with moderate workflows

In addition, Fig.11 depicts the performance of the proposed ACF-DFS-HHOA and the benchmarked schemes with different number of tasks associated with the small workflows the Average Energy Consumption Ratio by 5.62%, 6.94% and 8.69%.

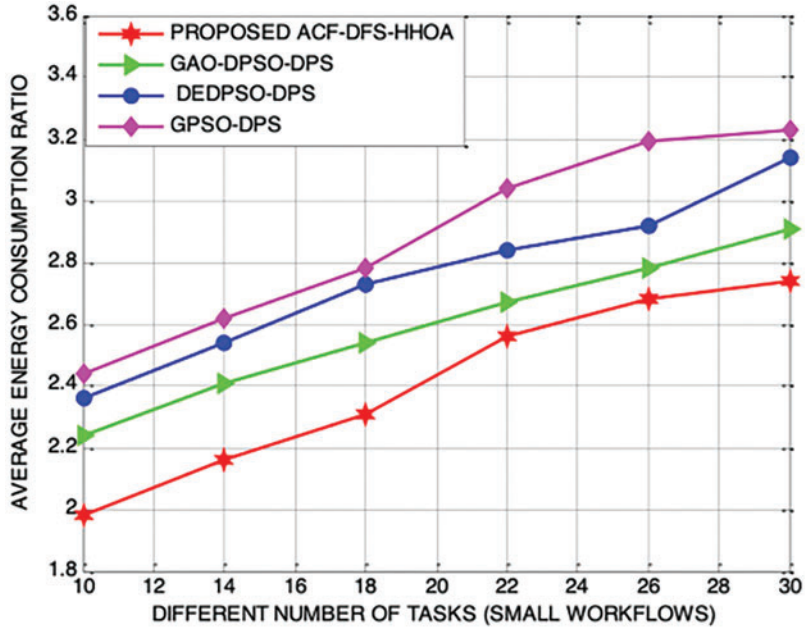

Figure 11: Proposed ACF-DFS-HHOA-Average energy consumption ratio with small workflows

6 Conclusions

In this paper, ACF-DFS-HHOA was proposed with the core objective of attaining optimal data placement strategy for a specific scientific workflow by reducing data transmission time and the energy utilization aunder the constraints of each datacentre storage capacity and energy threshold of the datacentre. It introduced the strategies of an adaptive cooperative and dispersed foraging into HHOA for superior guidance of the position updates that improve population diversity and effectively prevent the algorithm from being trapped into local optimality point. It also included a randomly shrinking exponential function for establishing a potential tradeoff between exploitation and exploration. As a part of the future plan, it is also planned to formulate a wingsuit optimization algorithm-based data placement strategy and compare it with the proposed ACF-DFS-HHOA to determine the degree of exploitation and exploration attributed by them during the searching process.

Funding Statement: The authors has no specific funding for this study.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] G. Juve, A. Chervenak, E. Deelman, S. Bharathi, G. Mehta *et al.,* "Characterizing and profiling scientific workflows," *Future Generation Computer Systems*, vol. 29, no. 3, pp. 682–692, 2013.

[2] J. Yu, R. Buyya and K. Ramamohanarao, Workflow scheduling algorithms for grid computing. In: *Metaheuristics for Scheduling in Distributed Computing Environments*. Berlin: Springer, pp. 173–214, 2008.

[3] J. Shuja, S. A. Madani, K. Bilal, K. Hayat, S. U. Khan *et al.,* "Energy-efficient data centers," *Computing*, vol. 94, no. 12, pp. 973–994, 2012.

[4] S. Didi, K. Bahhous, M. Zerfaoui, Z. Aboulbanine, H. Ouhadda *et al.,* "Experimental validation of a linac head geant4 model under a grid computing environment," *Biomedical Physics & Engineering Express*, vol. 8, no. 2, pp. 025007, 2022.

[5] A. Beloglazov, R. Buyya, Y. C. Lee and A. Zomaya, "A taxonomy and survey of energy-efficient data centers and cloud computing systems," *Advances in Computers*, vol. 82, pp. 47–111, 2011.

[6] R. Kleminski, P. Kazienko and T. Kajdanowicz, "Analysis of direct citation, co-citation and bibliographic coupling in scientific topic identification," *Journal of Information Science*, vol. 48, no. 3, pp. 349–373, 2022.

[7] J. Sun, L. Yin, M. Zou, Y. Zhang, T. Zhang *et al.,* "Makespan-minimization workflow scheduling for complex networks with social groups in edge computing," *Journal of Systems Architecture*, vol. 108, pp. 101799, 2020.

[8] L. Cui, J. Zhang, L. Yue, Y. Shi, H. Li *et al.,* "A genetic algorithm based data replica placement strategy for scientific applications in clouds," *IEEE Transactions on Services Computing*, vol. 11, no. 4, pp. 727–739, 2018.

[9] M. K. Hasan, M. Akhtaruzzaman, S. R. Kabir, T. R. Gadekallu, S. Islam *et al.,* "Evolution of industry and blockchain era: Monitoring price hike and corruption using BIoT for smart government and industry 4.0," *IEEE Transactions on Industrial Informatics*, vol. 18, no. 12, pp. 9153–9161, 2022.

[10] B. Liang, X. Dong, Y. Wang and X. Zhang, "Memory-aware resource management algorithm for low-energy cloud data centers," *Future Generation Computer Systems*, vol. 113, pp. 329–342, 2020.

[11] Z. Chen, J. Hu, G. Min and X. Chen, "Effective data placement for scientific workflows in mobile edge computing using genetic particle swarm optimization," *Concurrency and Computation: Practice and Experience*, vol. 33, no. 8, pp. e5413, 2021.

[12] V. D. Reddy, B. Setz, G. S. V. R. K. Rao, G. R. Gangadharan and M. Aiello, "Metrics for sustainable data centers," *IEEE Transactions on Sustainable Computing*, vol. 2, no. 3, pp. 290–303, 2017.

[13] A. A. Heidari, S. Mirjalili, H. Faris, I. Aljarah, M. Mafarja *et al.,* "Harris hawks optimization: Algorithm and applications," *Future Generation Computer Systems*, vol. 97, pp. 849–872, 2019.
[14] A. S. Menesy, H. M. Sultan, A. Selim, M. G. Ashmawy and S. Kamel, "Developing and applying chaotic harris hawks optimization technique for extracting parameters of several proton exchange membrane fuel cell stacks," *IEEE Access*, vol. 8, pp. 1146–1159, 2020.
[15] X. Zhang, K. Zhao and Y. Niu, "Improved harris hawks optimization based on adaptive cooperative foraging and dispersed foraging strategies," *IEEE Access*, vol. 8, pp. 160297–160314, 2020.
[16] M. Vasudevan, Y. C. Tian, M. Tang and E. Kozan, "Profile-based application assignment for greener and more energy-efficient data centers," *Future Generation Computer Systems*, vol. 67, pp. 94–108, 2017.