Australian Journal of Basic and Applied Sciences, 9(7) April 2015, Pages: 773-780

ISSN:1991-8178

Australian Journal of Basic and Applied Sciences

Journal home page: www.ajbasweb.com

A Novel Hybrid Algorithm of Onlooker Memetic Artificial Bee Colony and Cuckoo Search Using Global Integer Power Nap Strategy (Gipns) for an Efficient Disk Optimization

Dr.G. RaviKumar

Sri Ramakrishna Institute of Technology, Depatment of Information Technology, Coimbatore, Tamilnadu, India.

ARTICLE INFO

Article history:
Received 22 February 2015
Accepted 20 March 2015
Available online 23 April 2015

Keywords:
Onlooker Memetic artificial bee colony, heuristic cuckoo, disk subsystem, disk scheduling.

ABSTRACT

Background: Disk scheduling energy subsystem is becoming a popular research area in contributing the overall power consumption and becoming primary responsibilities of an operating system. The prior research work focuses the reduction in disk power, concentrating on designing the improved technology of disk performance. **Objective:** The research of this paper work turns the underlying disk power management scheme to be more effective in performing the disk scheduling with an efficient technique resulting in increasing the disk performance and decrease the power consumption. In this paper, a Novel hybrid algorithm of onlooker Memetic artificial bee colony and heuristic cuckoo with Global power nap strategy has been proposed for performing disk scheduling computation and designed to optimize the overall performance of disk drives**. Results:** The proposed technique utilizes Intensive token based disk power approach by using clever algorithm results in reducing the power consumption on disk subsystem, and to improve the storage capability for disks**. Conclusion:** The system produces a greater efficiency in reducing power consumption and produces global efficiency in disk scheduling.

To Cite This Article: Dr.G. RaviKumar., A Novel Hybrid Algorithm of Onlooker Memetic Artificial Bee Colony and Cuckoo Search Using Global Integer Power Nap Strategy (Gipns) for an Efficient Disk Optimization. ***Aust. J. Basic & Appl. Sci.,*** *9(7): 773-780, 2015*

INTRODUCTION

Operating System(OS) provides an interface between user and hardware. One of the major responsibilities of OS is to use the disk drive efficiently and to provide the fast access time. Reducing power consumption is becoming increasingly important for high-end cluster/server based systems and Managing power in the context of single-disk systems such as laptops and desktops has been a topic of extensive research (Mohan.B.C, et.al, 2011). Majority of the present disk provide diverse power modes of operation such as active and idle. Active mode occurs when the disk is servicing a request and idle mode, when it is spinning and not serving a request and one or more low power modes that use less energy than idle. Various researchers have taken up the chance of spinning down the disk during periods of idleness (Qingbo Zhu, 2005) or serving the requests at lower rotational speeds when performance is not a problem. Disk idle periods are considered as a vital factor in those power consumption techniques. To overcome the disk idle process disk power approach gets developed to increase the efficiency of disk process and lower the power consumption.

Motivated by these observations, recent research has focused on reducing power consumption of disk storage (Barroso.L, et.al, 2007) in the context of high-performance systems. The proposed approaches based upon the power consumption on disk subsystem, and disk scheduling process. In comparison, the prior research work describes explicit management of disk power modes and automatic disk layout detection. Irrespective of whether hardware or software based, most prior techniques to disk power management become more effective with long disk idle. Hard disk drives may be a major consumer of power in many computer systems, especially for computer systems such as servers that may utilize a large number of hard disk drives. A hard disk drive may provide one or more low power states of operation. Some of the low power states may turn off the motor that spins the disks as one way of saving power as the motor may be a large portion of the power consumed by the hard disk drive (Karaboga.D, et.al, 2007). To consume the power consumption in hard disk drive the research of this paper produces an Intensive Token based disk

Corresponding Author: Dr.G. RaviKumar, Sri Ramakrishna Institute of Technology, Depatment of Information Technology, Coimbatore, Tamilnadu, India.
E-mail: raviresearch14@gmail.com

power approach by using dynamic programming in parallel systems.

This approach combines the best characteristics of pure software based and the pure compiler based scheme. The approach implemented on run time system support. The proposed run time system receives as token hints, the preferred disk speeds from individual application and considering all token hints which decides the best storage for each disk in the system such that overall power consumption is reduced without affecting performance. The low power transition of the hard disk drive is performed if the token value evaluates to allow the low power transition of the hard disk drive and best storage for each disk in the system gets increased (Mias Nijjim, et.al, 2013). After performing the intensive token based approach in Disk power subsystem, the Disk scheduling can be performed by Novel hybrid algorithm of Onlooker Memetic artificial bee colony and heuristic cuckoo using Global power nap strategy. This technique is used for delay of data transfer in disk scheduling. The Onlooker Memetic artificial bee colony techniques initialize the population nodes of data from the disk and perform the heuristic cuckoo strategy resulting in producing the energy fitness function using Global power nap strategy algorithm. The energy fitness function is used to estimate the energy consumption of each disk drive and resulting in performing the disk scheduling efficiently. The proposed approach aims in reducing the power consumption and effective scheduling process by increasing the speed of disk drive.

Related work:

The author (Babu Lam, et.al, 2011) investigated the work of Automatic Generation of Feasible Independent Paths and Software Test Suite Optimization using Artificial Bee Colony (ABC) based novel search technique. In this approach, ABC combines both global search methods done by scout bees and local search method done by employed bees and onlooker bees. The parallel behavior of these three bees makes generation of feasible independent paths and software test suite optimization faster. This approach has the advantage that it won't create the problem of stuck at local optima but it has limitation in increasing mutation rate of Genetic Algorithm based approach.

In the paper (Sandeep Kumar, et.al, 2014) utilized the paper improve onlooker bee phase with help of a local search strategy inspired by memetic algorithm to balance the diversity and convergence capability of the ABC.It is tested over 12 well known un-biased test problems of diverse complexities and two engineering optimization problems; results show that the anticipated algorithm go better than the basic ABC. It results in improving the steadfastness, efficiency and accuracy of the ABC Algorithm but has lesser impact on Exploitation capability of to memetic algorithm.

In (D. Jeya Mala, et.al, 2009) investigated the work of non-pheromone based test suite optimization approach inspired nodes in the Software under Test (SUT), the artificial bees modify the test cases with time and the bee's aim is to discover the places of nodes with higher Coverage and finally with the highest usage by the given test case. This new approach out performs existing test optimization approach based on Genetic Algorithms (GA) in the task of software test optimization. It yields greater scalability and best in achieving global optimal Solution but uses less iteration to complete the task.

The author (Mias Nijjim, et.al, 2013) dissected the work of adaptive energy-saving scheme or DCAPS in parallel disk systems, adaptability in energy conservation can be achieved through the integration of a dynamic disk scheduling scheme and power management in parallel disk systems. The work carries the adaptability in energy conservation which can be achieved through the integration of a dynamic disk scheduling scheme and power management in parallel disk systems.It Reduces energy consumption of parallel disk systems in a dynamic environment over the same disk systems and results slighter variation in distributing data-intensive applications running in dynamically changing computing environments.

Qingbo Zhu, et.al, 2009) proposed the power-aware storage cache management algorithms that provide more opportunities for the underlying disk power management schemes to save energy. This approach produces off-line energy-optimal cache replacement algorithm using dynamic programming which minimizes the disk energy consumption. The result of this approach yields 22% more disk energy and provides up to 64% better average response time but produces delay in taking a time (a few seconds) for a disk to spin up from a low power mode to the active mode.

In (John Carter, et.al, 2014) which is intended in maintaining a token value based on an allowable number of low power transitions of a hard disk drive without adversely affecting reliability, compared to an actual number of low power transitions of said hard disk drive. The allowable number of low power transitions increases over the hard disk drive's lifetime. Before the hard disk drive performs a low power transition, the token is evaluated to determine if the hard disk drive is allowed to perform a low power transition.

It results in increasing the data storage device into the low power state yet inconsistent in describing about the predetermined value for the token with the elapsed time.

Energy consumption of Disk power:

Minimizing energy consumption of disk power is a vital issue for data centers. Storage is one of the main consumers of energy among various components of a data center. Earlier researches have revealed that the average idle period for a server disk in a data center is very little compared to the time taken to spin down and spin up. This greatly limits the effectiveness of disk power management schemes (Mohan.B.C, et.al, 2011).In analyzing drive power consumption, it is important to have a detailed understanding of how drives operate and their internal make up. Modern hard drives are composed of (i) magnetic platters, on which data is stored; (ii) a platter spindle motor, responsible for turning the spindle attached to the platters; (iii) read and write heads, which retrieve and commit data to the platters; (iv) read and write head actuator arms, which suspend the read and write heads above the platters;(v) a voice coil actuator, which positions the actuator arms over the correct location on the platters; (vi) and printed circuit board electronics, which consists of a buffer cache, a motor driver, and a main controller.The research of this paper insists in reducing the power consumption on disk Drive by reducing the Disk power seek time, and rotational speed and primarily focuses on reducing the storage space for power consumption on disk drives. The reduction in storage space and power consumption is reduced by the proposed system of Intensive token based disk power approach by using dynamic programming. The energy consumption on Disk is shown in Fig 1.

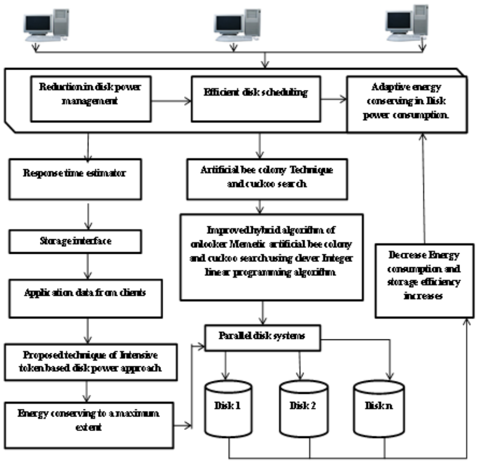

Fig. 1: Energy consumption on disk.

The efficient energy consumption on disk power is obtained by the proposed technique of

- Intensive token based disk power approach by using clever algorithm.
- Novel hybrid algorithm of onlooker Memetic artificial bee colony and heuristic cuckoo using Global power nap strategy.

Intensive token based disk power approach by using clever algorithm:

Many computer systems utilize hard disk drives for data storage. one of the methods of saving power in a computer system may be to spin down the hard disk drive. But the hard disk drive may be designed to handle only a certain number of spin-up/spin-down cycles over the lifetime of the motor without compromising reliability. Managing maximum allowable number of spin-up/spin-down cycles is not exceeded way of improving the reliability of the computer system (Suguna, et.al, 2011).A hard disk drive may be designed to accommodate a limited number of times of parking the heads become another way of improving the reliability of the computer system.

The Research of this paper leads to Intensive token based disk power approach for managing power consumed by a hard disk drive shown in Fig 2. It includes in maintaining a token value based on an allowable number of low power transitions of a hard disk drive determining that hard disk drive should be put into a low power state to conserve power. The maintaining of the token value comprises incrementing a first field of token value at intervals of time and incrementing a second field of said token value at a time associated with performing the low power transition of hard disk drive. The disk controller comprises the token storage element, the token value in a token storage element receiving a request to put the disk drive into a low power state making a determination that value stored in the token storage element is not greater than a predetermined value and delaying the disk drive from being entered into the low power state in response to the determination that value stored in the token storage element is not greater than the predetermined value.

The Energy Power consumption Model can be estimated as the sum of the CPU and disk power consumption.

$$power_{total} = \sum_{i=1}^{n}\left(power_{cpu,i}\right) + \sum_{i=1}^{n}\left(power_{disk,i}\right) + power_{idle} \quad (1)$$

The accountable power based on token predetermined value is given by

$$power_i = power_{cpu,i} + power_{idle,i} + \frac{power_{idle}}{n} \quad (2)$$

Then the CPU power consumption is given by

$$power_i = x_i \times u_{cpu,i} + y_i \times u_{cpu,i} \times predeter\min\ dedvalue_i \times d_i \quad (3)$$

Where $u_{cpu,i}$ is the utilization power for processor and $predeter\min\ dedvalue_i$ is defined by the token value at a time associated with performing the low power transition of hard disk drive, where X_i ,Y_i are the coefficient value for the cpu.

The disk power is characterized as

$$power_{disk,i} = z_i \times u_{disk,i} + e_i \quad (4)$$

The Total power equation is given by

$$power_{total} = \sum_{i=1}^{n}(x_i \times u_{cpu,i}) + \sum_{i=1}^{n}(y_i \times u_{cpu,i}) \times predeter\min\ edvalue_i) + \sum_{i=1}^{n}(c_i \times u_{disk,i}) + power_{idle} \quad (5)$$

The token value based on an allowable number of low power transition of hard disk drive can be given by

$$\min \left\|(D \times T + power_{idle} - power_{total}\right\|^2 subjectto : T > 0 \quad (6)$$

D is the matrix of cpu utilization and T is the token value data that need to be trained, since power is proportional to cpu and disk usage, all coefficients should be positive.

Clever algorithm is given by

for i=0 to n-1 do
for T=0 to i do
for all k such that k is a set of b distinct disk blocks do
if $x_i \in k$^$T = 0$*then*
x(k,T,i)-> $\max_{T',k'}\left(x(k',T,i-1')k \in repl(k',x_i)^{\wedge b} x_i \notin k' \right)$
else if $x_{i \in} k$ ^ $T \neq 0$ ^ $T>m$ *then*
X (k, T, i) -> X (k, T-1, i-1) + 1
else if Xi^ T≠ 0 ^ T≤ m then X (k, T, i) -> X (k, T-1, i-1)
else X (k, T, i) -> ∞
end if
end for
end for
maximum low power state achieved -> max $_{k,T}$ (X (k, T, n-1))
end for

If T=0, then to maximize overall possible token value of data in cache configuration k' such that k ϵ repl (k', xi) and T' ≤ i-1 where the number of different k' values are the number of possible different data blocks in the cache .we get the time complexity as $\sum_i y^k \times (y+1)$ which is O (y^{k+1} n^2). The above algorithm can be extended to work with multiple power modes and produces efficient adaptive saving disk power consumption.

Novel hybrid algorithm of Onlooker Memetic artificial bee colony (OMABC) and cuckoo search (CS) using Global integer power nap strategy (GIPNS):

Disk scheduling is a type of combinatorial optimization problem because as much as energy is wasted in idle time (John Carter, et.al, 2014). After performing the Intensive token based disk power approach, the major problem in Disk scheduling is that many scheduling do not fit into a common description model. Hence for scheduling problems it is too difficult to define a common frame work. For solving optimization problem, an efficient algorithm is necessary. The research of this paper proposed Energy optimized hybrid approach of onlooker Memetic artificial bee colony and cuckoo search using Global power nap strategy for performing efficient energy optimization scheduling process in minimizing the time taken for performing job scheduling in Disk and lower the power consumption of disk. Here $u_{disk,i}$ is the

Onlooker Memetic artificial bee colony:

Onlooker memetic artificial Bee Colony is used for service rescheduling. There are many nodes that are under-loaded and some are overloaded and the nodes that are idle (Sandeep Kumar, et.al, 2014). This technique uniformly distributes the load among the entire node. There are two types of bees: One is forger bees, which collect the nectar and move out for searching the nectar (food). They move randomly in any direction. After finding the nectar, they come back on the hive and start dancing on the dance floor. The duration of this dance is closely correlated with the search time experienced by the dancing bees. There are two types of dance, waggle dance which implies poor quality of nectar and tremble dance (round dance) which implies good quality of nectar. If the dance is tremble dance, then the new born bee agents fly to collect the good quality nectar and store them in the hive. After this operation, the old bee agents die and the new born bee agents start to fly with the good quality nectar stored in the hive, and finally mix them with those sources which are holding poor quality nectar. This process of distribution goes on until there is a uniform quality of nectar in all the sources. Similarly, in disk scheduling environment, it observe that some CPUs are overloaded for processing consumer's services, some are under loaded and some are totally idle. Onlooker memetic artificial Bee Colony Save the consumption of energy by turning this idle CPUs OFF and rescheduling services from overloaded CPUs to under loaded CPUs.

Heuristic Cuckoo:

In parallel machine scheduling problem with step-deteriorating jobs and sequence-dependent setup times, the objective is to minimize the total tardiness by determining the allocation and sequence of jobs on identical parallel machines. The cuckoo search strategy technique describes the concept of each egg in a nest represents a solution, and a cuckoo egg represents a new solution. The aim is to use the new and potentially better solutions (cuckoos) to replace a not-so-good solution in the nests (Choudhar. K, et.al, 2011). In the simplest form, each nest has one egg.

The algorithm can be extended to more complicated cases in which each nest has multiple eggs representing a set of solutions. In this problem, the processing time of each job is a step function dependent upon its starting time. An individual extended time is penalized when the starting time of a job is later than a specific deterioration date. The possibility of deterioration of a job makes the parallel machine scheduling problem more challenging than ordinary ones and provides the efficient optimization of Disk scheduling.

Table 1: Intensive Token based Disk power approach.

Cache memory	Cpu utilization	Energy of Disk
1	T5	E5
2	T5	E5
3	T5	E5
4	T5	E5
5	T5	E5
Cache memory		
1		
2		
3		
4		
5		
Token		
100		
200		
300		

Table 2: Clever algorithm results in minimizing energy consumption of disk power.

Cache memory	Cpu utilization	Energy of Disk
1	T6	E5+E(T6-T5)
2	T6	E5+E(T6-T5)
3	T6	E5+E(T6-T5)
4	T5	E5
5	T5	E5
Cache memory		
4		
1		
2		
3		
5		
Token		
400		
100		
200		

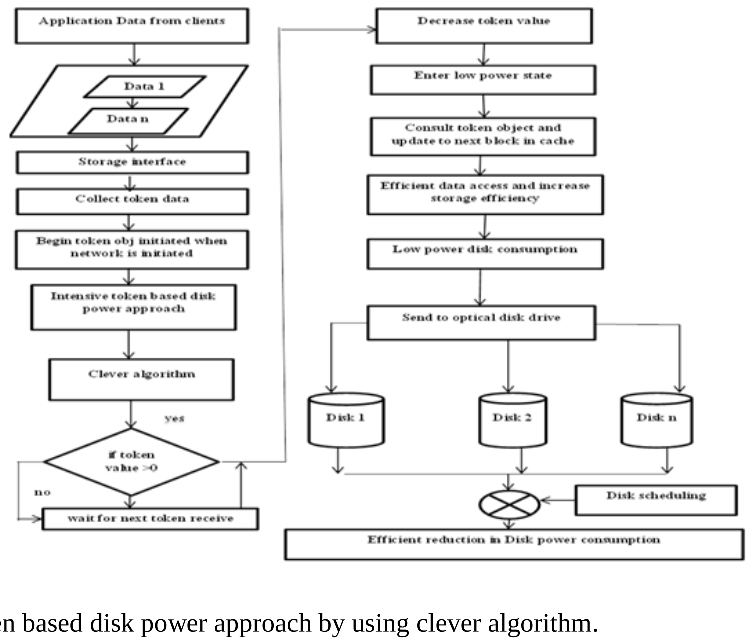

Fig. 2: Intensive token based disk power approach by using clever algorithm.

Novel hybrid algorithm:

The novel approach comprises of bee colony system in which the Disk scheduling divide into two parts, the first part, which looks after the proper management of overloaded and under loaded CPUs (service rescheduling) and the second part, which helps to manage the idle CPUs. The proposed bee colony algorithm has been performed for service rescheduling and Global integer power nap strategy for power consumption management.Thus for service rescheduling the tracker as hive which consist of bee agents, from where agents start to forge for nectar. Here nectar implies the threshold value of CPU. Poor quality of nectar implies lower threshold value of CPU. The dance floor represents the service scheduling table, where the information about the status of the CPUs that the bee agents have visited is stored. Again, if the dance of the bee agents is waggle dance then hive tracker indicates the service acceptor to accept new services from consumer. After this step, a Cuckoo search is derived for optimal solution and applies Global integer power nap strategy to find the idle CPUs and then turning them OFF in order to minimize the consumption of power and hence, lower the operational cost. The parameters are creation time C_t, Task creation time T_{ct}, Time to save disk information T_{diskf}. Task creation time can be defined as the time taken to create and save the disk information. Thus the Disk scheduler saves the task information in its database and it can be given as,

$$T_{ct} = C_t + t_{diskf} \qquad (7)$$

The Energy fitness Function of Disk power consumption is given by

$$Energy_{disk} = power_{ctr} + power_s + power_c \qquad (8)$$

$power_s$ represents the energy consumption by multi-channel memory, $power_c$ represents the energy consumption by the computing nodes and $power_{ctr}$ represents the energy consumption by the controlling nodes.

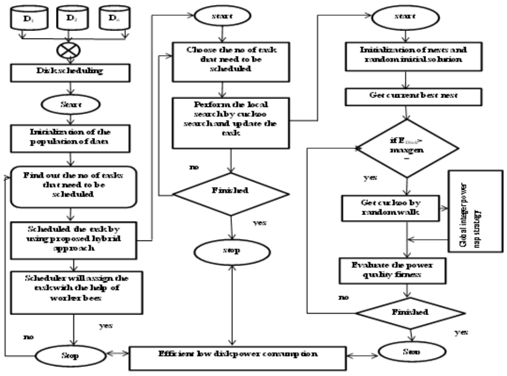

Fig. 3: Disk scheduling flow chart.

Proposed novel hybrid algorithm:

Disk power turn decider ()
Begin
{
$A_{rs} = A_0$, $H_S = H_0$
build $T^{k(t)}$
choose the cpu with probability μ_k

$$u_k(x,y) = \left\{ \frac{[A(x,y)].[n(x,y)^{\beta}.\mu(y)}{\sum [A(x,y)][n(x,y)]^{\beta}} \right. \qquad (9)$$

Find the H_s of s^{th} cpu

$$f(x) = \{1, turnoff, if H_s < H \qquad (10)$$
$$< 1, movetonextnode$$

}
place n bees at the nest repeat till the final path is decided
for t=1 to t_{max} do
Schedule the task using hybrid approach and assign the task T to the scheduler.
While (E_{disk}> maxgen)
Get the cuckoo value by random walk, if not replace it by energy fitness function.

$$E_{disk}{}^{t+1} = E_i^t + H_s(\mu(x,y) - q) + T_{ct} - Energy_{disk} \qquad (11)$$

Evaluate the quality and randomly choose nest among i, say j,
if (E_i^t> E_j)
Replace j value by new solution.

$$E^*_{disk} = E_i^t + Q\left(E_{disk}{}^{t+1}\right) \qquad (12)$$

Where(r, s) = represent the corresponding nodes in bees nest
β = deciding parameter
H_s= power of s^{th} cpu
(x, y)= set of all neighbours of corresponding nodes in cache.
H = probability of S^{th} cpu
maxgen= maximum generation of nodes.
q= random no
E^*_{disk}= global disk power consumption
E_j= new solution obtained by heuristic cuckoo approach.
Retain the best solution and the disk storage efficiency increases according to the task creation time.

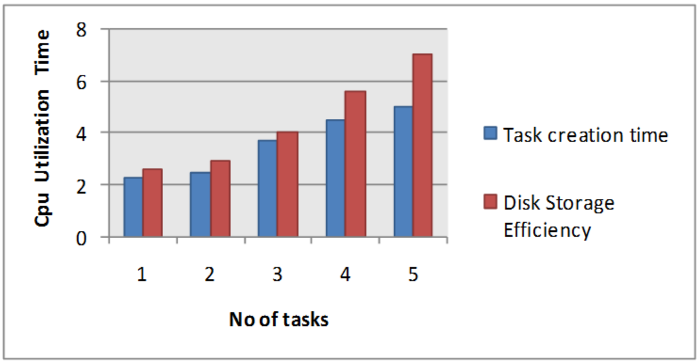

Fig. 4: Task Creation Time and Disk Storage Efficiency in Novel Hybrid Algorithm.

Simulation setup:

A simulation for Novel hybrid algorithm of onlooker memetic artificial bee colony optimization has been set up on a dual-core CPU, 512MB RAM, 120 GB hard-disk and with disksim simulator. It is determined by externally provided disk I/O request traces, which are constructed by the trace generator. The disk energy consumption comprises of all the energy consumptions in both active and idle periods, considering all the states of disks during the whole execution. Also, the performance numbers include all conflicts in accessing the parallel disk system. In this simulation, have assigned a number of nodes having different energy consumption level.

The heuristics compared in this case are ACO (ant colony optimization), SA simulated annealing algorithm, GA (genetic algorithm) in which the proposed Novel hybrid approach of Onlooker memetic artificial bee colony + Heuristic cuckoo + Global integer power nap strategy (OMABC+HC+GIPNS) is run for 1,250 iterations using 20 artificial bee colonies. OMABC+HC+GIPNS results are averaged over 15 trials. This practical has been performed on Simulator.

Simulation result:

Table 3: Comparison of disk power consumption of various algorithms ACO (ant colony optimization), SA (simulated annealing algorithm, GA (genetic algorithm with the proposed Technique OMABC+HC+GIPNS (Novel hybrid approach of Onlooker Memetic artificial bee colony + Heuristic cuckoo + Global integer power nap strategy)

Table 3:

Power Energy consumption on Disk				
No of tasks	OMABC+HC+GIPNS	ACO	SAA	GA
1	3.70	4.88	4.98	5.06
2	3.00	4.01	4.03	5.25
3	3.47	4.65	4.70	4.83
4	3.50	4.81	4.86	4.87
5	4.00	4.40	4.49	4.70

The table 3 illustrates the Disk power consumption in the proposed system yields low power energy consumption on disk when compared to other algorithms. It improves the efficiency of Disk storage and Disk scheduling and thereby results in decreasing energy consumption and the experimental result is shown in Fig 5.

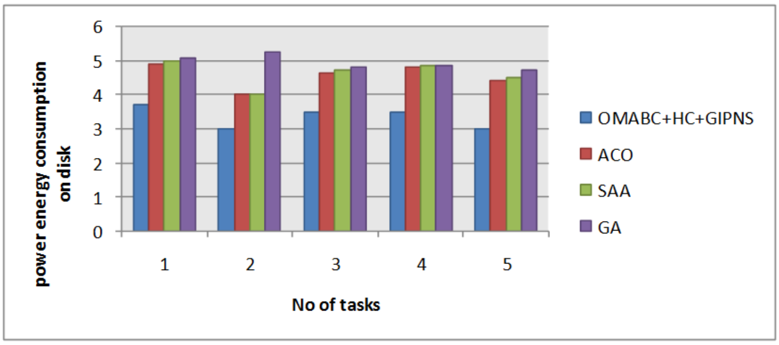

Fig. 5: Experimental result of disk power energy consumption.

Conclusion:

In this paper, the proposed system of Intensive token based disk power approach by using clever algorithm help the underlying disk energy management scheme save more energy and provides efficient CPU utilization. To perform the disk scheduling process more efficiently, the research of this paper produces the Novel hybrid approach of Onlooker memetic artificial bee colony and Heuristic cuckoo using Global integer power nap strategy results in minimizing the Disk power consumption to a maximum extent. The Novel research of this paper aims in minimizing the Disk power to a greater extent and increase the storage efficiency of disk.

Conflict of Interests:

The authors declared no potential conflict of interests with respect to the authorship or publication of this paper.

REFERENCES

Barroso, L. and U. Holzle, 2007. The case for energy proportional computing , IEEE Computer.

Benini, L., G. Paleologo, A. Bogliolo and G. Micheli, 2010. “Policy optimizationfor dynamic power management,” IEEE Trans. Computer Aided Design, 18: 813–833.

Corelli, S. and D. Grunwald, 2002. Massive Arrays of Idle Disks forStorage Archives, Proc. 2002 ACM/IEEE Conf. Supercomputing, pp: 1-11.

Choudhar, K., G.N. Purohit, 2011. A new testing approach using cuckoo search to achieve multi-objective genetic algorithm, J. compute textbf, 3(4): 117–119.

Karaboga, D., B. Basturk, 2007. A Powerful and Efficient Algorithm for Numerical Function Optimization Artificial Bee Colony (ABC) Algorithm, Journal of Global Optimization, 39(3): 459–471.

Mohan, B.C., R. Baskaran, 2011. Energy aware and energy efficient routing protocol for adhoc network using restructured artificial bee colony system”, Communications in Computer and Information Science, 169(3): 473-484.

Mohan, C.B. and R. Baskaran, 2011. Survey on recent research and implementation of ant colony optimization in various engineering applications”, Inter. J. Computer. Intel. Sys, 4: 566-582.

Mukherjee, K. and G. Sahoo, 2009. Mathematical Model of Cloud computing framework using Fuzzy Bee Colony optimization Technique, International Conference on Advances in Computing, Control and Telecommunication Technologies, IEEE Explore.

Qingbo Zhu, 2005. power aware storage cache management”, IEEE Transaction on computers, 54-5.

Ramanathan, D. and R. Gupta, 2000. System level online power management algorithms,” Proc. Design, Automation, Test Eur., 606–611.

Suguna, N., 2011. An independent rough set approach hybrid with artificial bee colony algorithm for dimensionality reduction, Applied Sci., 8: 261-266. DOI: 10.3844/ajassp, 261-266.

Venkaiah Ch, D.M. Vinod Kumar, 2011. Fuzzy adaptive bacterial foragingcongestion management using sensitivity based optimal active power reschedulingof generators, Applied Soft Computing, 11: 4921–4930.