International Journal of Engineering Research and Technology.
ISSN 0974-3154 Volume 7, Number 1 (2014), pp. 17-27

http://www.irphouse.com

Fuzzy Bee Colony Optimization for Resource Allocation in Grid Computing

T.Sunil Kumar Reddy, P Venkata Krishna* and P. Chenna Reddy

Abstract

The dynamic resource allocation algorithm determines the optimal resource allocation for the available tasks. Resource allocation scheme do not consider the Grid computing issues such as bandwidth allocation, power utilization which may increase the interference of co-resource and adversely affecting the aggregation of resource across several grids. This paper proposes an adaptive resource allocation scheme using fuzzy bee colony optimization technique**.** It takes the available resource for each task relates with honey bee behaviour provides best throughput and QoS in Grid computing. It investigates the fuzziness of execution time of the task and performs the resource assignment scheme with the objective of solving the above issues like maximizing task allocation and equalizing the task assignment with efficient power utilization to the resources for task**.**

Keywords: Grid scheduling, resource estimation, power reduction, performance analysis

1. Introduction

Resource allocation for tasks in grid is one of the challenging research area to provide sharing, selection, and aggregation of resources. Due to computational requirements there is a need of abstraction for the job in resource management of grid environments. But Virtual Machine (VM) technology, being an efficient support of resource management systems based on lease abstraction [1].

Resources allocation is done based on two types of users such as local user and external user. The local user request is phrased as the request given locally for resources. External user request is termed as the request given externally which means requesting a gateway for accessing the shared resources. The local request is given more priority over the external in cluster [2] but due to external requests, the local requests must not be delay in execution.

The preempting of VMs from local user imposes overheads to the underlying system and degrades the utilization of the resources [1]. In terms of external user's perspective, preemption increases the response time of the requests. The above issues can be solved by choosing the best resource for any specific request in queue at any given time. The available resource at each task can be analyzed to determine the selection of number of VMs used in that task. We can predict the resources offering best performance based on certain features of the task completion on each individual VM at specific space/time. This prediction becomes useful in ranking the available VMs and picking up the best resource suitable to accomplish the task. The decision to allocate the best resources is to contrast the performances achieved by dynamically chosen resource against a statically allocated resource.

Preemptive scheduler by PECT [3] aims scheduling different classes of job in a Grid. It is based on applying coarse-grain time sharing and suspending of task in VMs on disk. The main difference with our work is of scheduling the resource scheduling to minimize the number of VMs preemptions of task. In [4] a prediction method to analyze the unavailable periods in fine grained cycle sharing systems in the mixture of local and global jobs. Margo et al. [5] proposed a scheduling based on priority to increase the utilization of the resources. It determines the priority of each job based on the expansion factor and number of processing elements.

The proposed algorithm implements a mechanism to choose the available resources dynamically by considering the appropriate number of VMs. Its performance is evaluated by deploying in dynamic resource allocation. The Grid with dynamic resource allocation for the task with the prediction of VM available is proposed, implemented and evaluated. The experimental setup ranks all the available resources based on the achievable performance by estimating the best possible set of properties about each resource. Next, we can construct a resource allocation engine that can choose and dynamically allocate the best resource for the existing tasks in Grid with considering the appropriate set of properties in terms of the power utilization.

2. Optimized Fuzzy Bee based Resource Scheduling Architecture

The architecture designed to estimate the resource for the allocation of the tasks has to be scheduled with the available VMs. It consist of three phases , in first phase the available VMs and host from Grid resource were estimated at the middleware of the Grid layers for the allocation of the task.

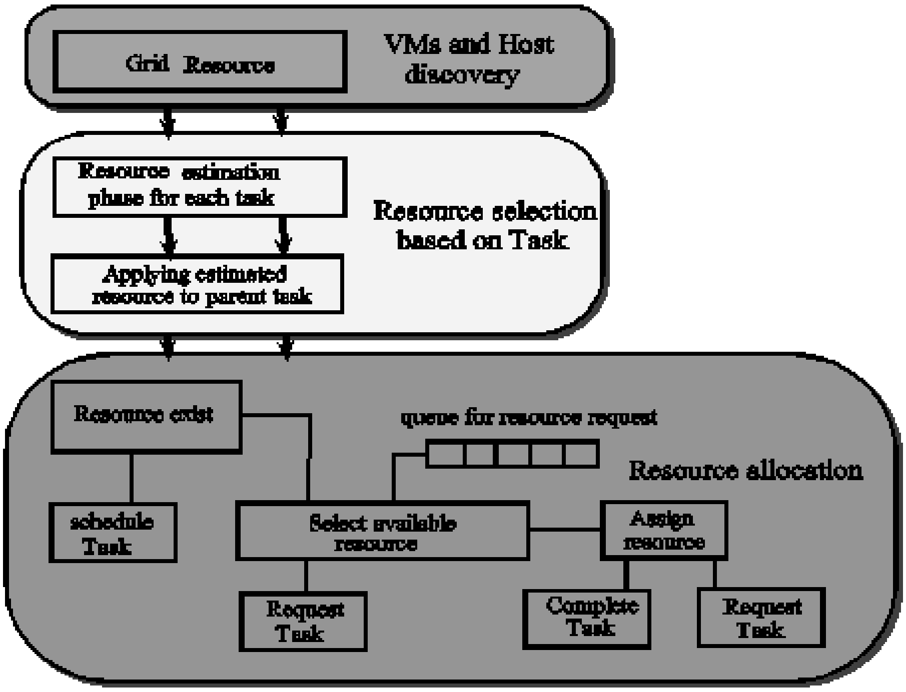

Figure 1. Optimized Fuzzy Bee based Resource Scheduling

Next, the available resource for each task is calculated based on effective utilization of the resource from the available resource. Depending on the size of task and execution time, the task is scheduled to the resource which is best suitable for processing the task by VMs available is selected. Finally resource assignment phase hop by hop resource allocation scheme. If there is already a resource allocated between a task, the same task is used for processing the available resource else it sends a task request message to the other available VMs and assigns a VM with task completion reply. If VMs are available the request is kept in the queue and processed later depending on availability of resource using request task and completion task mechanism.

3. Optimized Fuzzy Bee based Resource Scheduling Algorithm (OFB-R):

The proposed algorithm, resources are allocated to the task in proportion to the requirement of Grid computing. The factors that are considered for resource estimation:

1. Resource available for task
2. Idle time of the resource
3. Task execution time

The resource available for the task can be evaluated

$R_T=R*(T_R/E_T)$

Where R_T is the remaining available resource for task T, R is the available resource, T_R is the idle time of resource and ET is the execution time of the task.

Let $[E_1, E_2]$ be the interval graph between hive to a specific degree of fuzzy nectar that the bee is searching for, then we have Min T: $[E_1, E_2] \rightarrow I$, where I represents unit interval [0, 1]. After getting the required nectar as per the below mathematical model, we need to schedule the web service to specific resources. Our proposed algorithm for searching a specific service in Grid designed with energy saving considerations to satisfy both QoS and resource constraints.

The finishing time of a task ECT and the execution time of a task ET on virtual machine VMs can be denoted as E_{xy} with availability of resource are maintained in the datacenter used by tasks to allocate VMs. Current distance of all available VMs can be calculated based on the information received from the datacenter. The algorithm is a dynamic task resource allocation technique merged with the concept of fuzziness with honey bee behavior is shown in Fig. 2.

$R_T = 0$ when bee does not move from the hive and dies of starvation. The task does not move to any resource and expire without execution.

$$R_T = \frac{[a_{xy}(t)]^{\alpha} - [E_{xy}]^{\beta}}{\sum [a_{xy}(t)]^{\alpha} - [\frac{1}{E_{xy}}]^{\beta}}$$ When bee moves from the hive to the allowable nectars

Where a_{xy} (t) is the arc fitness from the hive x to nectar y at the time t. $R_T = 1$ then bee moves from the hive to the required nectar means, the task is allocated to the resources it required.

α is a binary variable that turns on or off the arc fitness. β is the parameter that controls the significant level of heuristic distance.

The objective function is

$$MinT = \sum_{T=0}^{n} E_T R_T \tag{1}$$

It must be satisfy the conditions below

$E_T >= 0$

$E_{T1} + E_{T2} >= E_{T2}$

In the algorithm proposed, if a link is established for a specific path, the same resource is used. In case of non-availability of a link, several other tasks might be in wait for the one task called *waited-task* to give up the resource at the same time. When the waited-task forfeits the resource, the important problem is which task first uses this task and resource. The issue can be addressed by defining a queue to store the addresses of task which are waiting for the resource (called *waiting-task*). Thus, waited-task utilizes the queue to store the addresses of waiting-task. After the waited-task surrenders the resource, it can be used by the first task in the queue to route to the waited-task. The waited-task rejects further addresses of other task when the queue is full and will notice the task which wants to store the address. Since the length of the queue cannot be too large, it causes the task to wait for long and impair the performance of the network.

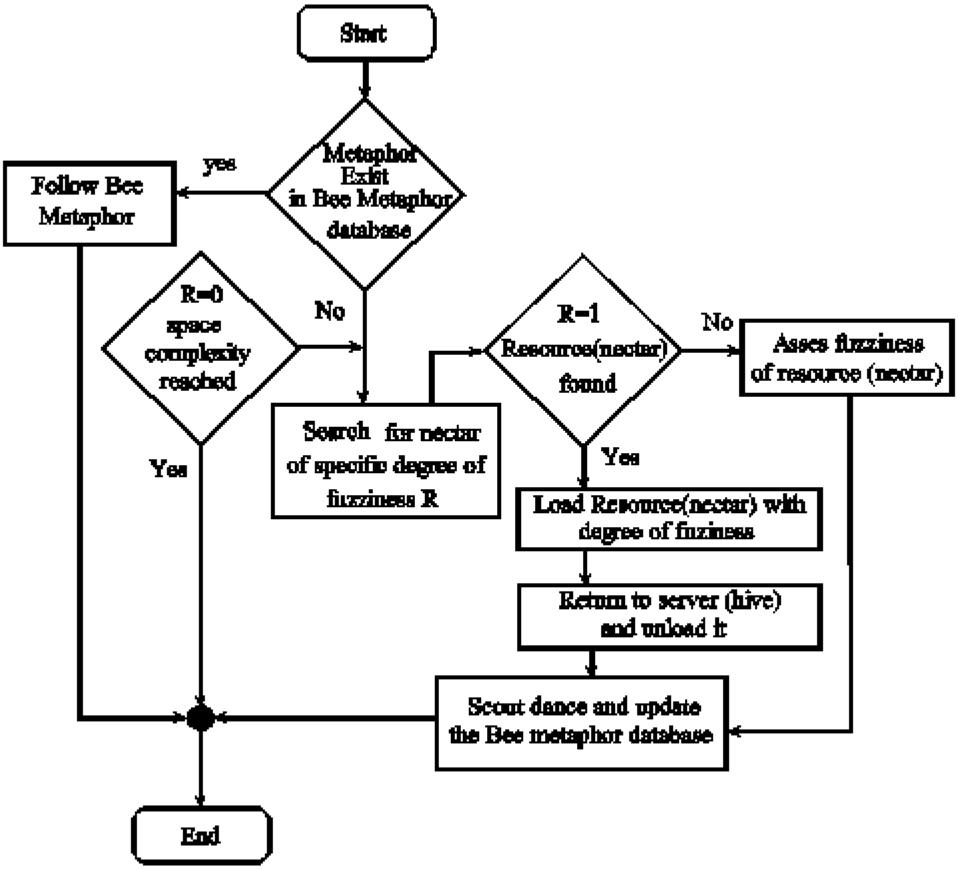

Figure 2 Algorithm flow of Optimized Fuzzy Bee based Resource Scheduling

4. Result and Discussion

The performances of our algorithm based on execution time reduction have been evaluated by a cloud computing simulator. The classes were extended for this simulator to suit the implementation of our algorithm. The makespan and cost of well known scheduling algorithms like Non power aware (NPA), Round Robin (RR) Dynamic Voltage and Frequency Scaling (DVFS) and First come and First serve (FCFS) are evaluated in terms of VMs, power utilization, MIPS and type of host shown in Fig(2). The proposed algorithm addresses minimum completion time with cost to schedule the tasks provides optimal result. The comparisons between the algorithms are shown below in the tables 1 and table2. Considering the random workflow for different number of task.

The simulation setup to evaluate the performance of OFB-R is shown in table 3. The average utilization of power and VMs evaluated shown better result compare to other approaches as shown in Fig (2). The number task executed with and without estimation of the resource shown less time execution with effective utilization of the resources. The power utilization of with different no of host with fixed VMs is evaluated for different techniques is shown from Fig (4) - Fig (6)

Table 3- The Simulation Setup

Host	10-50
VMs	50-300
Core	1-4
RAM	512-24576
Bandwidth	1000-1000000
Scheduling	FCFS, DVFS, RR

Table 2- Virtual Machines Setup

Resource/ VM	VM Type1	VM Type2	VM Type3	VM Type4
MiPs	750	1000	1500	2000
Cores	1	1	1	1
RAM	512	512	1024	1024
Bandwidth	1000	1000	1000	1000
Storage	25000	25000	25000	25000

Table 3- Host Setup

Resource/ Host	Host type0	Host type1	Host type2	Host type3	Host type4
MiPs	1500	2000	2500	4200	6000
Cores	1	1	2	4	4
RAM	24576	24576	24576	24576	24576
Bandwidth	100000	100000	100000	100000	100000
Storage	100000000	100000000	1000000000	100000000	100000000

Table 4 Task scheduling without estimation of VMs

Task(Length)	Time	VMs	Overall
T1(10), T3(20),T4(30),	60s	M1	60s
T2(40),T3(50), T4(60), T5(70)	M1=40s, M2=50s, M3=60s+70s	M1={T2}, M2={T3}, M3={T4,T5}	130s
T6(80),T7(90), T8(100)	M1=80s, M2=90s+100s	M1=T6,M2={T7,T8}	190s

Table 5 Task scheduling with estimation of VMs

Task(Length)	VMs	Time	Overall
T1(10), T3(20),T4(30),	M1	60s	60s
T2(40),T3(50),T4(60),T5(70)	M1={T2,T5}M2={T3},M3={T4}	M1=40s+70s, M2=50s, M3=60s	120s
T6(80),T7(90), T8(100)	M1={T6,T8}, M2={T7}	M1=80+100s, M2=90s	180

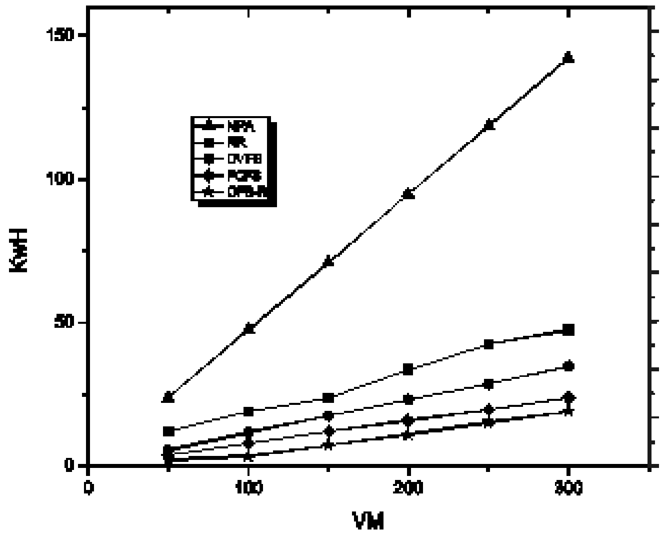

Figure 2 Power utilization

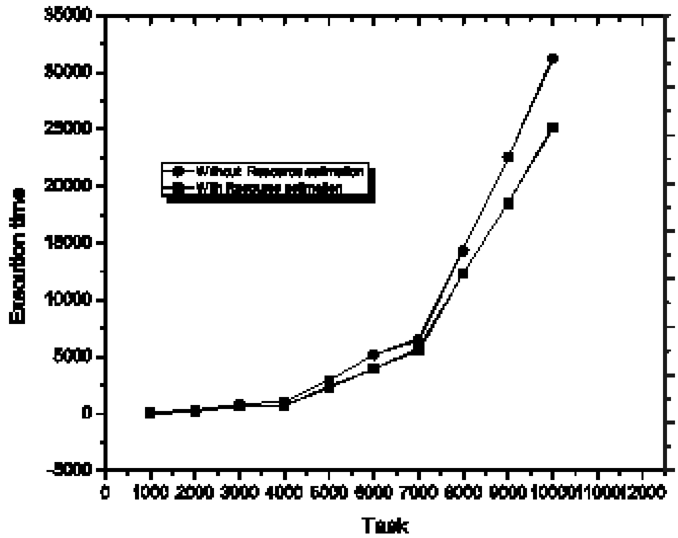

Fig. 3. Execution time based on number of task

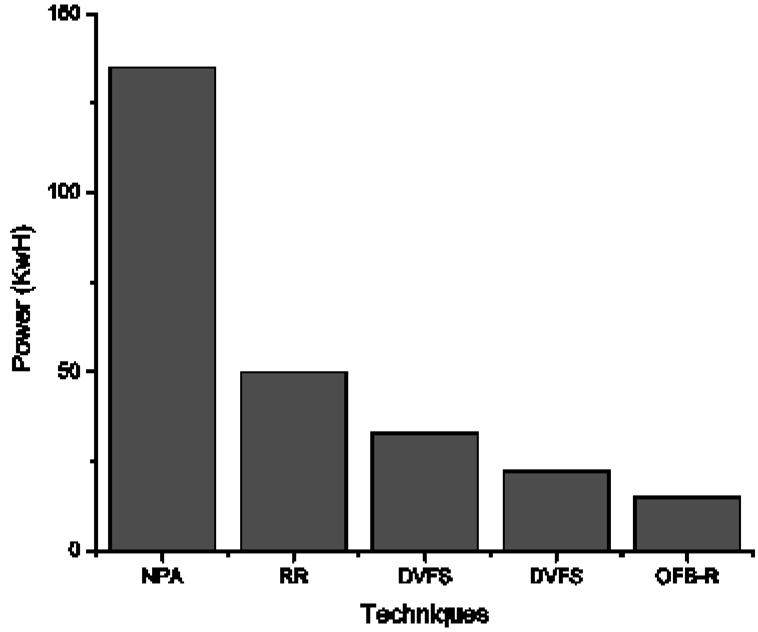

Fig. 4. Power utilization performance

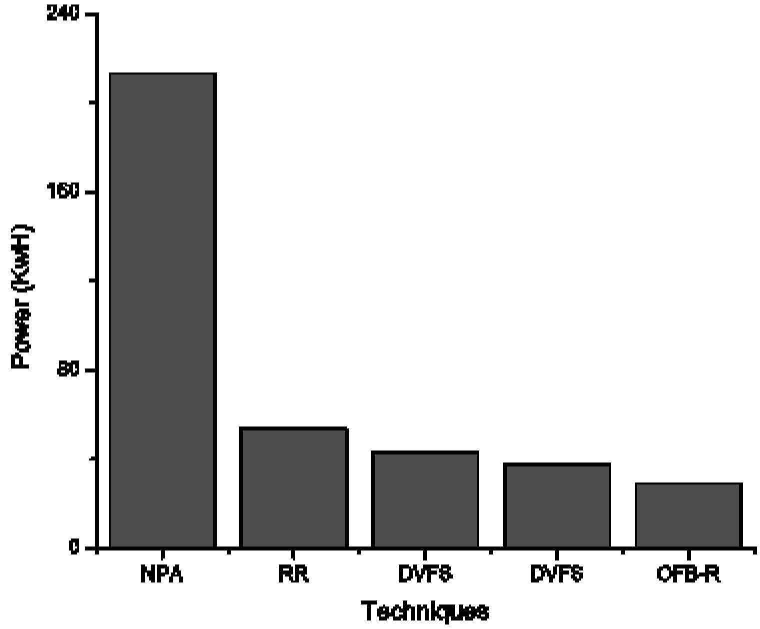

Fig 5: Power utilization performance for the 150 Host with 800 VMs

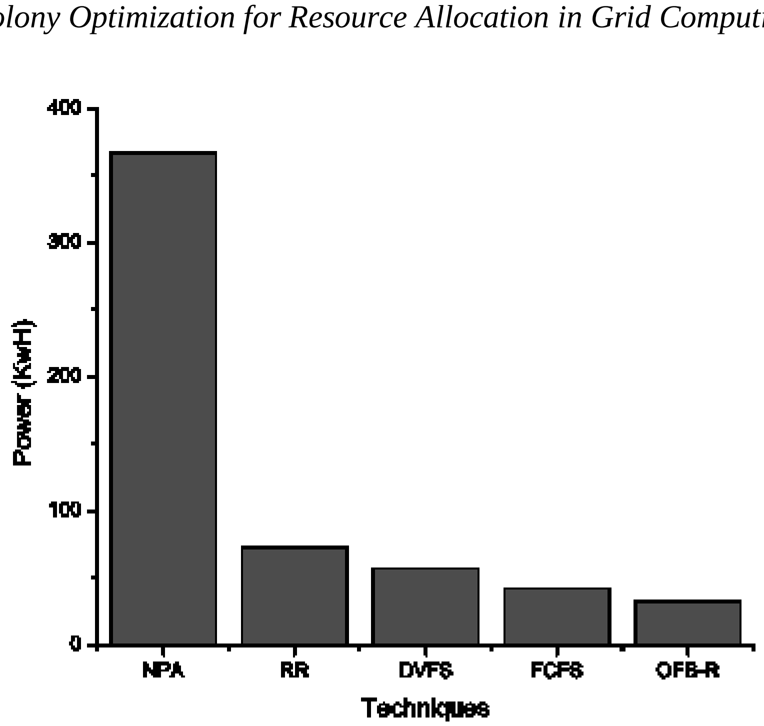

Fig 6: Power utilization performance for the 350 Host with 800 VMs

5. Conclusions

The static resource assignments are insufficient for the requirements of the task available in the Grid resulting in traffic-load. The dynamic strategy of resource allocation improves grid computing system performance in the context of overload of the task. The proposed adaptive resource allocation mechanism for Grid computing system is optimized using fuzzy bee colony optimization technique depending on the resource reveals a appreciable improvement of data throughput performance with less utilization of power and effective utilization of resource.

References

1. B. Sotomayor, K. Keahey, I. Foster, Combining batch execution and leasing using virtual machines, in: Proceedings of the 17th International Symposium on High Performance Distributed Computing, ACM, New York, NY, USA, 2008, pp. 87–96.
2. J.S. Chase, D.E. Irwin, L.E. Grit, J.D. Moore, S.E. Sprenkle, Dynamic virtual clusters in a Grid site manager, in: Proceedings of the 12th IEEE International Symposium on High Performance Distributed Computing, Washington, DC, USA, 2003, pp. 90–98.
3. A. Sodan, Service control with the preemptive parallel job scheduler Scojo-PECT, Journal of Cluster Computing (2010) 1–18.
4. X. Ren, S. Lee, R. Eigenmann, S. Bagchi, Prediction of resource availability in

fine-grained cycle sharing systems empirical evaluation, Journal of Grid Computing 5 (2) (2007) 173–195.

5. M. Margo, K. Yoshimoto, P. Kovatch, Preemption and priority calculation in production job data, in: Proceedings of the 9th LCI International Conference on High-Performance Clustered Computing, Urbana, Illinois, USA, 2008.
6. 6. Sudip Misra, P. Venkata Krishna and V. Saritha, "*LACAV: an energy-efficient channel assignment mechanism for vehicular ad hoc networks*" The Journal of Supercomputing, Springer Netherlands, DOI: 10.1007/s11227-011-0552-1.
7. Sudip Misra, P. Venkata Krishna and V. Saritha, "*An efficient approach for distributed channel allocation with learning automata-based reservation in cellular networks*" SIMULATION , 2011, doi: 10.1177/0037549711420175.
8. Sudip Misra, P. Venkata Krishna and Kiran Issac Abraham, "*Learning Automata Solution for Medium Access with Channel Reservation in Wireless Networks*" Wireless Personal Communications (WPS), Vol. 56, Issue 1, pp. 55-72, (2011), Springer.
9. P. Venkata Krishna, Sudip Misra, Mohhamed S. Obaidat and V. Saritha, "*An efficient approach for distributed dynamic channel allocation with queues for real time and non realtime traffic in cellular networks"* Journal of Systems & Software, pp1112-1124, 82(2009), (Elsevier) DOI: 10.1016/j.jss.2009.01.043.

BIOGRAPHY

▲ **Name:** Sunil Kumar Reddy.T
Address: Global college of Engineering & Technology, Kadapa
Education & Work experience: Completed Bachelor of Technology in Information Technology from Satyabama University, Chennai and Masters in Networking from V.I.T University, and pursuing PhD in JNTU University Anatapur.Currently working as Associate. Professor & Head Department of CSE in Sir Vishvehwaraiah Institute of Science and Technology,,Madanapall..He published several papers in National and International journals. His area of interest includes Grid computing, mobile computing, computer networks, Information security etc.
Tel:09966950000
E-mail:sunil_reddy1982@yahoo.co.in

▲**Dr. P. Venkata Krishna** is a Professor in the School of Computing Science and Engineering, VIT University, Vellore, India. He received his B. Tech in Electronics and Communication Engineering from Sri Venkateswara University, Tirupathi, India, M. Tech in Computer Science & Engineering from REC, Calicut, India and he received his Ph.D. from VIT University, Vellore, India. His research interests include Mobile and wireless systems, Grid Computing and Distributed Systems. He has authored few books on Computer Networks and Programming Languages. He has delivered several key note addresses and chaired sessions in reputed conferences. He is serving as the reviewer for many reputed journals like IEEE Trans. on Mobile Computing, IET Journal on Communications, International Journal of Security and Networks (IJSN) etc. He is serving as editor or member of Boards in various reputed Journals

▲ **Name:** P. Chenna Reddy
Address: ,JNTUcollegofEngneering ,Pulivendula,kadapa
Education & Work experience: Dr. P. Chenna Reddy did his B.Tech from S.V. University College of Engineering, Tirupati, M.Tech & Ph.D from JNTU, Hyderabad. He has 15 years of Teaching experience. His areas of interest are Computer Networks and related fields. is currently working on Bio inspired networking. He is currently working Professor in Department of CSE at JNTUA College of Engineering, Pulivendula. He has published several papers in reputed journals and conferences