



A review on high performance grid computing QOS applications using cluster computing based on particular weight transforms

*Pasam Venkateshwar Rao, research scholar at faculty of CSE, SSSUTMS-Sehore.
Dr. Niraj Sharma, Associate Professor- Department of CSE, SSSUTMS-Sehore*

Abstract

In growing constant booking figuring's for gathering and cross section processing is to help various kinds of employments. Broadly, computational weights submitted to a pack can be characterized into three sorts: progressive, independently recognizable and discretionarily separable. An abstractly separable remaining Job at hand show is a fair gauge of some certifiable applications, e.g., appropriated search for a model in content, sound, graphical, and information base records; disseminated planning of immense assessment information reports; and various reenactment issues. All parts in such an application frequently demand a vague sort of planning, and regarding the goliath total estimation, the taking care of on each individual segment is nearly nothing. Everything considered applications transform into an essential sort of gathering remaining Jobs at hand and appropriately offering QoS to emotionally particular weights transforms into a gigantic issue for pack based examination processing workplaces.

Keywords: Qos, Performance, Reliability

1. Introduction

The main focus is on The fundamental spotlight is on various components that would give first class registering condition in conveyed systems. Predominant figuring is scarcely portrayed as the improvement and use of the snappiest and most extraordinary processing systems i.e., likely registering. It covers mechanical, political and monetary features of the dispersed figuring undertaking. The huge revelations and proposition with respect to the issue have been summed up in the later fragments of the paper.

To fabricate the profitability of any endeavor, we require a structure that would outfit tip top close by adaptabilities and cost efficiencies for customer. Appropriated registering, as we are generally careful, has ended up being especially notable over the earlier decade. Circulated registering has three significant makes, to be explicit, gathering, organization and cloud. With a particular ultimate objective to develop an unrivaled appropriated system, we need to utilize all the recently referenced three kinds of processing.



In this paper, we will at first have an introduction of the apparent multitude of three kinds of disseminated registering. Therefore analyzing them we will research slants in registering and green reasonable processing to improve the presentation of a dispersed system. Finally demonstrating the future degree, we wrap up the paper proposing an approach to achieve a Green prevalent circulated system using pack, grid and distributed computing.

1.1 Cluster Computing

Up and down the prototype age of computing, ventures endorsing tremendous estimations and abundant handling were subject to purview or sprinkling corporative. Such huge supercomputers and structures were extreme for singleton. Despite the fact that costs of PCs are thumbing down, supercomputers are still beyond anyone's ability to see. Because of which Donald Becker and Thomas Sterling imported Beowulf grouping in 1993 which lit off the counter PCs, constructing a bunch that imitated torpid supercomputers. The principal behind this is to make a computing plan for giving the essential handling power at an ostensible cost. As the hubs are archive of processors, security is totally breezy and thereupon readiness in restricting interconnected systems from outside systems administration. Conceding significant catalyst in computing power, grouping surely has hitches and reluctances as an equivalently unique innovation. Distributed computing controls to assail an expanded circle of bunching by allowing the hubs to win everywhere throughout the world and furthermore be multiuse machines. Distributed computing has a similar to idea as bunching, enabling numerous hubs to take a shot at extensive issues in parallel in the wake of breaking them into littler units. Endlessly work units are distributed a few times to excessively numerous hubs, controlling the probabilities of handling slips and describe for preparing done on monotonous CPUs. The customer regulates the information resurgence and capitulation laps alongside the code basic to arrange the CPU how to schedule the function unit.

1.2 Favorable circumstances of Cluster Computing

There are various favorable circumstances of executing group figuring in the applications. Not many of them to be examined are as per the following:

Cost viability – Even centralized server PCs is by all accounts incredibly steady, group processing is more in usage in view of their cost-adequacy and conservative. Additionally, these frameworks give improved execution than that of centralized server PC organizations.

Handling speed – The bunch registering frameworks offer similar preparing speed as that of centralized server PCs and the speed is likewise equivalent to supercomputers.



Broadened asset accessibility – Computers run over incessant breakdowns, so to wipe out this disappointment, group PCs are accessible with high accessibility. Thus, when one hub gets fizzled, different hubs will be dynamic and will work as an intermediary for the bombed hub. This ensures for improved accessibility.

Expandability – The following pivotal preferred position of this group processing is its improved adaptability and expandability. As they launch the possibility to consolidate various extra assets or the organizations to the common PC framework.

Adaptability – Cluster registering can be moved up to the predominant particular or stretched out through the option of extra hubs (PC frameworks).

1.3 Applications

There exist various uses of group registering, Few of those are:

Group registering can be actualized in climate demonstrating

Stands as help in-vehicle breakdown and atomic recreations

Utilized in picture handling and in electromagnetic as well

Impeccable to be utilized in the utilizations of astronomy, streamlined features and in information mining

Help to tackle complex computational issues

Holds the adaptability to distribute outstanding Job at hand as little information segments and which is called lattice figuring.

Bunch processing has the ability to work in many web applications, for example, Security,

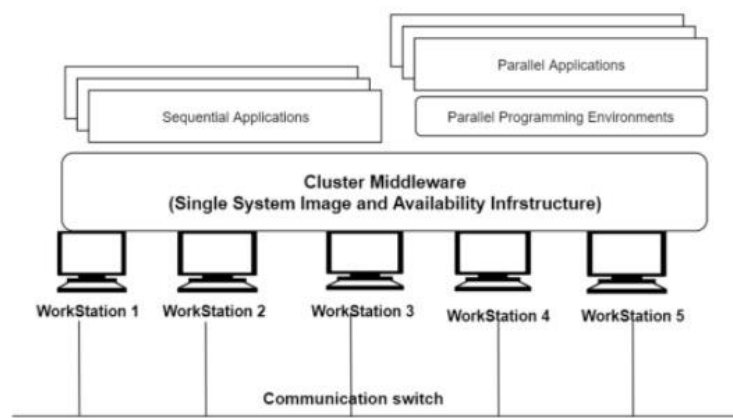


Figure.1. Cluster Computing Architecture



Search Engines, Database workers, web workers, intermediary, and email.

There are life hero applications through this methodology like they can conjecture the event of quakes or twisters.

1.4 Sorts of Cluster Computing

As groups are broadly used in correspondence to the unpredictability of the data, to oversee content and the foreseen working pace. A considerable lot of the applications that foresee high accessibility without a decrease in vacation utilize the situations of group processing. The kinds of group figuring are:

- a. Group load adjusting
- b. High–Availability groups
- c. Elite groups
- d. Group Load Balancing

a. Group load adjusting

Burden adjusting groups are utilized in the circumstances of expanded organization and web use and these bunches proceed as the essential factor. This sort of bunching strategy offers the advantages of expanded organization limit and improved execution. Here the whole hubs remain as strong with all the occurrence where the whole hub objects are totally mindful of the solicitations those are available in the organization. All the hubs won't work in a solitary cycle though they readdress the solicitations separately as they show up contingent upon the scheduler calculation. The other urgent component on the heap adjusting strategy is adaptability where this element is practiced when each worker is completely utilized. Alongside the workers in load adjusting which have a comparative limit in customer response, a considerable lot of the worries are made the same number of the solicitations may be tended to by these workers those may be in disarray. Thus, load adjusting makes balance for the clients and workers. Indeed, even burden adjusting distributes approaching solicitations or traffic for its assets from the associated PCs which work comparable projects and those create as a group. At the point when a hub disappointment happens, at that point the solicitations are again dispensed over the accessible hubs. This methodology is executed in Linux web workers.

2. Literature Review

Marco A. S. Netto and Rajkumar Buyya et.al.[1] Packs of-Jobs (BoTs) are equivalent applications with no between Job correspondence. A collection of issues in a couple of fields, including computational science [18], picture great for censing [23], and colossal endeavors [3], have been shown as



BoT applications. Conversely with the message passing model, BoT applications can be successfully executed on different resource providers to satisfy a customer time limitation or diminish the customer response time. Notwithstanding the way that BoT applications include inde-swinging Jobs, the results made by all Jobs contain the plan of a singular issue. All things considered, customers need the whole game plan of endeavors executed to have the alternative to post-measure or separate the results. Thusly, the upgrade of the ag-gregate set of results is noteworthy, and not the progression of a particular endeavor or assembling of Jobs [4].

Huge degree equivalent applications have been sent in enlisting workplaces, for instance, TeraGrid, DAS-3, Grid'5000, and NAREGI. In any case, the insightful and industry com-munities have moreover been considering the utility enrolling perspective for executing these applications on conditions, for instance, Sun Grid, IBM On-Demand Computing, and Ama-zon Elastic Compute Cloud (EC2) [9]. These conditions offer permission to resources by charging customers according to the application's advantage, which is resolved in contracts called Service Level Agreements (SLAs).

The execution of a BoT application on different utility figuring workplaces is an appealing response for consent to customer time requirements. This is because more Jobs of a lone BoT ap-plication can execute in equivalent and these workplaces need to pass on a particular QoS level, regardless the providers are rebuffed. An authority community containing a metasched-uler is responsible for appropriating the Jobs among re-source providers as shown by their pile and system con-figuration. In any case, allocating resources from different providers is trying because these benefit providers can't reveal a ton of information about their local weight to the metascheduler. Extraordinary job needing to be done is private information that associations don't reveal viably since it may impact the business arrangement of competitors.

Much work has been done on arranging BoTs [7, 15, 16]. Regardless, little effort has been committed to design these applications with cutoff time necessities [6, 14, 24], explicitly considering confined weight information open from resource providers. Thusly, we extend the current plans and add to the assessment field in the follow-ing ways:

We present three methodologies for making offers to design BoT applications (Section 4). Offers are an instrument wherein resource providers reveal their eagerness for executing an entire BoT or simply bit of it without uncovering their close by weight and system capabili-ties. At whatever point providers can't satisfy a period imperative, they produce offers with another conceivable cutoff time. For the offer age inside resource providers (Section 3), we impact the work made by Islam et al. [12,13], however



solidifying proposals for executing an application on various resource providers is inspired by the provisioning model of Singh et al. [22] and the extension measurement of Siddiqui et al. [21]. We investigate the proportion of information resource providers need to open to the metascheduler and its impact on the booking of occupations neighborhood and outside to the advantage providers .

In spite of the way that the essential motivation of this work is scheduling cutoff time constrained BoTs on utility figuring facilities, the methodologies can moreover be used in various circumstances. Therefront, we have avoided esteeming and money related models in this work. Note that our assessment can be used for other applications rather than Bag-of-Jobs. Organizations running on different resource providers moreover require an arranged allocation subject to the stack information access and organizations' solicitations. Another model is the booking of equivalent times of work measures, where a get-together of endeavors need to adjust ish, so as the work cycle can proceed to the accompanying stage.

According to Paul Pop, Petru Eles, Zebo Peng, Viacheslav Izosimov[3] Cycle booking and schedulability assessment have been intensively amassed in the past numerous years. The peruser is suggested [2] for an examination on these subjects. Work in the zone of booking and schedulability assessment improved on a very basic level by considering particular correspondence shows, for instance, Token-Ring [12], Controller Area Network (CAN) [1, 5, 20], ATM [7], and time-division multiple access (TDMA) [21] shows.

An extending number of continuous applications are today implemented using passed on models involving interconnected gatherings of processors. Each such bundle has its own correspondence show and two gatherings give by methods for an entry, a center point connected to them two. Such a models is used in a couple of use areas: vehicles, producing plant systems, networks on chip, etc.

Considering, for example, the vehicle business, the way where handiness has been appropriated on a plan has created after some time. From the outset, every limit was executed on a gave gear section. In any case, in order to use the advantages even more beneficially and abatement costs, a couple of limits have later been inter-ground in one center and, at the same time, certain convenience has been scattered in excess of a couple of center points. Though an application is ordinarily scattered more than one single pack, we begin to see

In spite of the way that the essential motivation of this work is scheduling cutoff time constrained BoTs on utility figuring facilities, the methodologies can moreover be used in various circumstances. Therefront, we have avoided esteeming and money related models in this work. Note that our assessment can



be used for other applications rather than Bag-of-Jobs. Organizations running on different resource providers moreover require an arranged allocation subject to the stack information access and organizations' solicitations. Another model is the booking of equivalent times of work measures, where a get-together of endeavors need to adjust ish, so as the work cycle can proceed to the accompanying stage. According to Paul Pop, Petru Eles, Zebo Peng, Viacheslav Izosimov[3] Cycle booking and schedulability assessment have been intensively amassed in the past numerous years. The peruser is suggested [2] for an examination on these subjects. Work in the zone of booking and schedulability assessment improved on a very basic level by considering particular correspondence shows, for instance, Token-Ring [12], Controller Area Network (CAN) [1, 5, 20], ATM [7], and time-division multiple access (TDMA) [21] shows.

An extending number of continuous applications are today implemented using passed on models involving interconnected gatherings of processors. Each such bundle has its own correspondence show and two gatherings give by methods for an entry, a center point connected to them two. Such a models is used in a couple of use areas: vehicles, producing plant systems, networks on chip, etc.

Considering, for example, the vehicle business, the way where handiness has been appropriated on a plan has created after some time. From the outset, every limit was executed on a gave gear section. In any case, in order to use the advantages even more beneficially and abatement costs, a couple of limits have later been inte-ground in one center and, at the same time, certain convenience has been scattered in excess of a couple of center points. Though an application is ordinarily scattered more than one single pack, we begin to see applications that are appropriated over a few groups. This pattern is driven by the need to additionally lessen costs, improve asset utilization, yet additionally by application limitations like being genuinely near specific sensors and actuators. Besides, not exclusively are these applications appropriated across networks, however their capacities can trade basic data through the door hubs. Such applications are inalienably hard to dissect and plan.

There are two essential methodologies for dealing with undertakings continuously are started at whatever point a specific occasion is noted. In the time-triggered (TT) approach, exercises are started at foreordained focuses in time. There has been a long discussion in the ongoing and installed frameworks networks concerning the upsides of TT and ET approaches [3, 10, 22]. An intriguing examination, from a more mechanical, specifically car, point of view, can be found in [11]. The end there is that one needs to pick the correct methodology relying upon the particularities of the cycles. This implies not just that there



is no single "best" way to deal with be utilized, yet in addition that inside a specific application the two methodologies can be utilized together, a few cycles being TT and others ET.

In [13] we have tended to plan issues for frameworks where the TT and ET exercises share a similar processor and transport. A fundamen-count distinctive engineering way to deal with heterogeneous TT/ET sys-tems is that of heterogeneous multi-groups, where each bunch can be either TT or ET:

In a period set off bunch (TTC) cycles and messages are booked by a static cyclic arrangement, with the transport executing a TDMA convention, for example, the time-set off convention (TTP) [10].

On occasion set off bunches (ETC) the cycles are planned by a need based preemptive methodology, while messages are sent utilizing the need based CAN transport [4]. In this unique situation, in [16] we have proposed a way to deal with schedulability investigation for multi-group conveyed inserted frameworks. Beginning from such an examination, in this paper, we address explicit plan issues for multi-bunch frameworks: dividing an application between the TT and ET groups, and planning the usefulness of the application on the heterogeneous hubs of a group, with the end goal that the circumstance limitations of the last execution are ensured. Our plan space investigation approach depends on a proficient branch-and-bound calculation.

The paper is sorted out in seven segments. The following area pre-sents the application model just as the equipment and programming design of our frameworks. Segment 3 presents the dividing and planning issue we are tending to in this paper, and Section 4 presents the schedulability examination for multi-groups and our favorable to presented apportioning and planning branch-and-bound technique is presented in Section 5. The last two areas present the test results and ends.

Equal PCs are known to have different various models X. Liu and S. Goddard [6]. The Massively Parallel Processor (MPP) is a broad equal structure. Normally, it contains a couple hundred taking care of parts (center points). The center points are related through quick framework and all of them runs an alternate copy of a working structure. MPP executes a "typical nothing" plan. Regularly, every center point involves just of crucial memory and in any event one processors. Additional fragments, like I/O contraptions, could be incorporated. Curiously, the Symmetric Multiprocessor (SMP) structure has "shared-everything" designing. Each center point can moreover use the benefits of substitute centers (memory, I/O contraptions). A lone working structure continues running on all of the centers. Finally, a conveyed equal plan portrays an arrangement of independent machines, generally completed a wide land district. Each center point is an absolutely confine machine with its own working system. Besides, every



center can be of absolutely extraordinary designing. Any blend of MPPs, SMPs or plain PCs could be added to a conveyed structure.

X. Liu, X. Zhu, P. Pradeep, Z. Wang, and S. Sharad et.al. (2007) [7] describes "A gathering is such an equal or conveyed taking care of structure which includes a social event of interconnected stay lone PCs collaborating as a lone, joined figuring resource". A PC center can be a lone or multiprocessor system (PC, workstation, or SMP) with memory, I/O workplaces, and a working structure. Unmistakable terms rise to describe assorted bundle "flavors": Networks of Workstations (NOW) or Cluster of Workstations (COW), gatherings of PCs (CoPs) or Piles of PCs (PoPs).

Starting late, various QoS-delicate applications, typi-cally sight and sound applications, have drawn out various QoS-maintained execution models [2, 12, 16, 20] and schedulers [4, 25, 11, 29]. Regardless, simply static QoS maintain isn't adequate. Eventually it is hard to choose the execution rate. Various applications have variable execution rates. For test ple, a different goal various sensor radar worldwide situating system gives more thought to brisk moving concentrations than to slow tar-gets. Also, the stages moreover impact the execution rate. Ob-viously, a program runs snappier on a 800MHz P3 processor than on a 400MHz P2 processor. On normal working structures, a colossal number of applica-tions have been worked under time-sharing schedulers. We call those applications legacy applications. Various legacy ap-plications, especially sight and sound applications, starting at now have QoS demands. In any case, normal time-sharing systems don't give any QoS maintain. Contemplating the countless legacy applications, it is truly infeasible to recreate those applications upon an absolutely new stage. A probable plan is to help legacy applications at the twofold level. The Variable-Rate Execution (VRE) model [10] was ini-tially expected to handle the above issues. The VRE model is an extension of the Rate-Based Execution (RBE) model [12]. While the RBE model plans Jobs at a fixed av-erage rate, the VRE model sponsorships variable execution rates by allowing variable period and execution time limits during runtime. It in like manner licenses Jobs to join and leave the system at optional events. Work like the VRE model is the rate-based most reliable cutoff time (RBED) scheduler presented in [6], which was in-restrictively and simultaneously made. In spite of the way that the theoretical model isn't the central point of convergence of this paper, we include the difference between the RBED scheduler and the VRE model in Section 2.



This paper in a general sense bases on dealing with three issues: (I) showing legacy applications under the VRE model; (ii) how to alter the execution rates during runtime; (iii) the implementation and programming interface.

To be feasible with the time-sharing schedulers, we split the execution of a legacy application into progressive time cuts. The qualification in our execution model is that each endeavor is depicted by two limits, period (p) and size of time cut (c), and each time cut is named a deadline subject to the period. The Earliest Deadline First (EDF) scheduler guarantees that the endeavor will get a period cut of size c in each season of p time units. We change the assessments of c and p to control the execution movement of a legacy application. The rate controller fragment is familiar with deal with issue (ii). Since different applications can have contrast ent execution plans, it is difficult to give a comprehensive rate controller to all applications. Therefore, we give a ton of interfaces to customers to re-try their specific rate controllers. The improvement of specific rate controllers is past the degree of this work. In any case, we arranged a fundamental rate controller as a default rate controller for legacy multi-media applications.

This work, including the VRE scheduler and the rate controller framework, was executed in Linux. We barely modified the Linux segment to have the alternative to capably stack customer revamp schedulers. The loadable scheduler framework separates specific booking courses of action from the arranging segment. Customers can stack their specific schedulers without rebooting the system as long as the schedulers are hidden consistence with the scheduler interface presentation duced in Section 4.1. Furthermore, customers can change the execution case of a VRE Job by joining a specific rate controller. The rest of this paper is figured out as follows. Zone 2 analyzes related work. Fragment 3 presents the planning model acknowledged in this work. Zone 4 gives an audit on the programming interface. Territory 5 presents the experiments and results..

Starting late, various QoS-sensitive applications, Sharad Singhal* et.al.typically sight and sound applications, have drawn out various QoS-maintained execution models [2, 12, 16, 20] and schedulers [4, 25, 11, 29]. In any case, simply static QoS maintain isn't adequate. Before long it is hard to choose the execution rate. Various applications have variable execution rates. For test ple, a different target various sensor radar worldwide situating system gives more thought to brisk moving concentrations than to slow tar-gets. Likewise, the stages moreover impact the execution rate. Obviously, a program runs speedier on a 800MHz P3 processor than on a 400MHz P2 processor.



On common working structures, countless applications have been worked under time-sharing schedulers. We call those applications legacy applications. Various legacy applications, especially sight and sound applications, starting at now have QoS demands. Nevertheless, standard time-sharing structures don't give any QoS maintain. Contemplating the an enormous number of legacy applications, it is truly infeasible to remake those applications upon an absolutely new stage. A probable game plan is to help legacy applications at the twofold level. The Variable-Rate Execution (VRE) model [10] was initially planned to handle the above issues. The VRE model is an extension of the Rate-Based Execution (RBE) model [12]. While the RBE model schedules Jobs at a fixed average rate, the VRE model support variable execution rates by allowing variable period and execution time limits during runtime. It moreover allows Jobs to join and leave the structure at optional events.

Work like the VRE model is the rate-based most reliable cutoff time (RBED) scheduler presented in [6], which was in-restrictively and simultaneously made. In spite of the way that the theoretical model isn't the crucial point of convergence of this paper, we highlight the differentiation between the RBED scheduler and the VRE model in this Section .

This paper essentially revolves around dealing with three issues: (i) showing legacy applications under the VRE model; (ii) how to adjust the execution rates during runtime; (iii) the implementation and programming interface.

To be reasonable with the time-sharing schedulers, we split the execution of a legacy application into progressive time cuts. The differentiation in our execution model is that each endeavor is depicted by two limits, period (p) and size of time cut (c), and each time cut is delegated a deadline reliant on the period. The Earliest Deadline First (EDF) scheduler guarantees that the endeavor will get a period cut of size c in each season of p time units. We alter the assessments of c and p to control the execution movement of a legacy application. The rate controller section is familiar with deal with issue (ii). Since different applications can have contrast ent execution plans, it is difficult to give a comprehensive rate controller to all applications. Hence, we give a great deal of interfaces to customers to re-try their specific rate controllers. The improvement of specific rate controllers is past the degree of this work. In any case, we arranged an essential rate controller as a default rate controller for legacy multi-media applications.

This work, including the VRE scheduler and the rate controller framework, was executed in Linux. We imperceptibly modified the Linux segment to have the alternative to effectively stack customer revamp schedulers. The loadable scheduler framework segregates specific booking courses of action from the arranging segment. Customers can stack their specific schedulers without rebooting the structure as long



as the schedulers are hidden consistence with the scheduler interface presentation duced in Section 4.1. In addition, customers can change the ex-ecution case of a VRE Job by joining a specific rate controller.

High-openness (moreover called failover) packs execute the possibility of overabundance C. Lu, T. F. Abdelzaher, J. A. Stankovic, and S. H. Youngster (2001) [8]. They are used for mission-fundamental applications. A representation is a web worker at an association that must not crash and burn. HA is refined by having various discretionary workers that are right duplicates of a basic worker. Persistently, they screen made by the fundamental worker holding up to expect control if it crashes and burns. In this principal shape, simply a single machine (worker) is in unique use while the remaining ones are in stay by mode.

Burden changing packs give better execution by scattering outstanding Job at hand among center points in a gathering Abdelzaher, J. A. Stankovic, and S. H. Kid (2006) [9]. Consider a web worker. In case store changing is executed, unmistakable inquiries are given to different centers for taking care of. Synchronous treatment of requests achieves speedier overall response season of the worker. LB is capable by different systems. A fundamental cooperative estimation or more incredible computations that rely upon analysis from the individual machines can sort out which machine is generally fitting for dealing with the accompanying Job.

Proposed System Model

A bunch comprises of a head center, meant by P0, related by methods for a change to N planning centers, implied by P1, P2. . .PN. We acknowledge that all dealing with centers have similar computational force and all associations from the change to the planning centers have a comparable information move limit. The system show acknowledges an ordinary bunch condition in which the head center doesn't partake in estimation. The aspect of the head center is to recognize or excuse moving toward endeavors, execute the arranging count, segment the outstanding Job at hand and pass on information protuberances to the getting ready center points. Since different centers measure various information pieces, the head center progressively sends every datum irregularity to relating dealing with center through the switch.

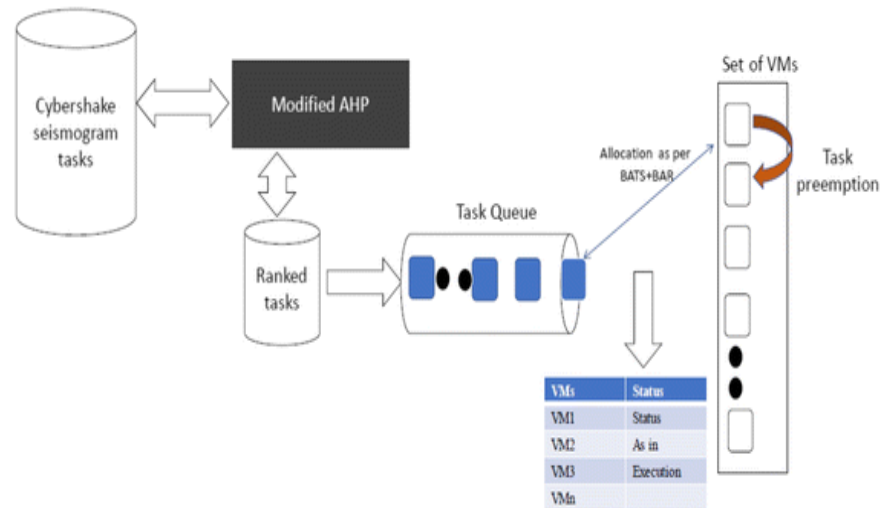


Fig Proposed system architecture

We expect that information transmission doesn't happen in equal. For abstractly separable weights, Jobs and sub Jobs are independent. Thusly, when executing such applications planning center points don't talk with one another.

Conclusion

This research proposed the heuristic calculation that performs Job planning and allots assets proficiently in distributed computing conditions. We utilize genuine Cybershake and Epigenomics logical work processes as information assignments for the framework. At the point when we contrast our proposed heuristic methodology and the current BATS and IDEA systems regarding turnaround time and reaction time, we find that our methodology gives improved outcomes. Then again, from the perspective of asset usage, the proposed heuristic methodology proficiently distributes assets with high utility. We got the greatest use result for figuring assets, for example, CPU, memory and transfer speed. Most existing frameworks think about just two assets, CPU and memory, in assessing their presentation the proposed framework includes data transfer capacity as an asset. Future work will zero in on more viable planning calculations in which turnaround time and reaction time will be improved.



References

- [1] M. A. S. Netto and R. Buyya. Offer-based scheduling of deadline-constrained bag-of-Jobs applications for utility computing systems. In Proceedings of the 18th International Heterogeneity in Computing Workshop, in conjunction with the 23rd IEEE International Parallel and Distributed Processing Symposium (IPDPS'09), Roma, Italy, May 2009.
- [2] S. Parekh, J. Hellerstein, T. S. Jayram, N. Gandhi, D. Tilbury, and J. Bigus. Using control theory to achieve service level objectives in performance management, 2001.
- [3] P. Pop, P. Eles, Z. Peng, and V. Izosimov. Schedulability-driven partitioning and mapping for multi-cluster real-time systems. In Proc. of 16th Euromicro Conference on Real-Time Systems, pages 91–100, July 2004.
- [4] X. Qin and H. Jiang. Dynamic, reliability-driven scheduling of parallel real-time jobs in heterogeneous systems. In Proc. of 30th International Conference on Parallel Processing, pages 113–122, Valencia, Spain, September 2001.
- [5] K. Ramamritham, J. A. Stankovic, and P. fei Shiah. Efficient scheduling algorithms for real-time multiprocessor systems. IEEE Trans. on Parallel and Distributed Systems, 1(2):184–194, April 1990.
- [6] X. Liu and S. Goddard. Supporting dynamic qos in linux. In Proceedings of the 10th IEEE Real-Time and Embedded Technology and Applications Symposium, pages 246–254, May 2004. Toronto, Canada.
- [7] X. Liu, X. Zhu, P. Pradeep, Z. Wang, and S. Sharad. Optimal multivariate control for differentiated services on a shared hosting platform. In IEEE Conference on Decision and Control, pages 3792–3799, 2007.
- [8] C. Lu, T. F. Abdelzaher, J. A. Stankovic, and S. H. Son. A feedback control approach for guaranteeing relative delays in web servers. In In IEEE Real-Time Technology and Applications Symposium, pages 51–62, 2001.
- [9] C. Lu, Y. Lu, T. F. Abdelzaher, J. A. Stankovic, and S. H. Son. Feedback control architecture and design methodology for service delay guarantees in web servers. IEEE Transactions on Parallel Distributed Systems (TPDS), 17(9):1014–1027, 2006.



- [10] C. Lu, J. A. Stankovic, G. Tao, and S. H. Son. Design and evaluation of a feedback control EDF scheduling algorithm. In Proceedings of the IEEE Real-Time Systems Symposium, pages 56–67, 2007.
- [11] C. Lu, X. Wang, and X. Koutsoukos. Feedback utilization control in distributed realtime systems with end-to-end Jobs. IEEE Transactions on Parallel and Distributed Systems, 16:550–561, 2005.
- [12] Y. Lu, T. Abdelzaher, C. Lu, and G. Tao. An adaptive control framework and its application to differentiated caching services. In International Workshop on Quality of Service (IWQoS), Miami Beach, FL, May 2009.
- [13] J. MacLaren. Advance reservations: State of the art. <http://www.fz-juelich.de/zam/RD/coop/ggf/graap/graap-wg.html>. 143
- [14] A. Mamat, Y. Lu, J. Deogun, and S. Goddard. Real-time divisible load scheduling with advance reservations. In 20th Euromicro Conference on Real-Time Systems, pages 37–46, July 2008.
- [15] G. Manimaran and C. S. R. Murthy. An efficient dynamic scheduling algorithm for multiprocessor real-time systems. IEEE Trans. on Parallel and Distributed Systems, 9(3):312–319, 2010.
- [16] M. W. Margo, K. Yoshimoto, P. Kovatch, and P. Andrews. Impact of reservations on production job scheduling. In 13th Workshop on Job Scheduling Strategies for Parallel Processing, June 2007.
- [17] P. Mohapatra. Dynamic real-time Job scheduling on hypercubes. J. of Parallel and Distributed Computing, 46(1):91–100, 2011.
- [18] T. F. Abdelzaher and V. Sharma. A synthetic utilization bound for aperiodic Jobs with resource requirements. In Proc. of 15th Euromicro Conference on Real-Time Systems, pages 141–150, Porto, Portugal, July 2003
- [19] M. A. S. Netto and R. Buyya. Offer-based scheduling of deadline-constrained bag-of-Jobs applications for utility computing systems. In Proceedings of the 18th International Heterogeneity in Computing Workshop, in conjunction with the 23rd IEEE International Parallel and Distributed Processing Symposium (IPDPS'09), Roma, Italy, May 2009
- [20] W. Y. Lee, S. J. Hong, and J. Kim. On-line scheduling of scalable real-time Jobs on multiprocessor systems. Journal of Parallel and Distributed Computing, 63(12):1315–1324, 2003



- [21] L. He, S. Jarvis, D. Spooner, X. Chen, and G. Nudd. Hybrid performance-oriented scheduling of moldable jobs with qos demands in multiclusters and grids. In Proc. of the 3rd International Conference on Grid and Cooperative Computing, pages 217–224, Wuhan, China, October 2004
- [22] X. Lin, Y. Lu, J. Deogun, and S. Goddard. Enhanced real-time divisible load scheduling with different processor available times. In 14th International Conference on High Performance Computing, December 2007
- [23] R. A. Ammar and A. Alhamdan. Scheduling real time parallel structure on cluster computing. In Proc. of 7th IEEE International Symposium on Computers and Communications, pages 69–74, Taormina, Italy, July 2002
- [24] M. Degermark, T. Kohler, S. Pink, and O. Schelen. Advance reservations for predictive service in the internet. *Multimedia Systems*, 5(3):177–186, 1997.
- [25] Y. Diao, C. W. Wu, J. L. Hellerstein, A. J. Storm, M. Surendra, S. Lightstone, S. Parekh, C. Garcia-Arellano, M. Carroll, L. Chu, and J. Colaco. Comparative studies of load balancing with control and optimization techniques. In American Control Conference (ACC), June 2005
