Tuijin Jishu/Journal of Propulsion Technology
ISSN: 1001-4055
Vol. 47 No. 03 (2026)

Benchmarking Concurrency Performance Across Hypervisor Architectures

K. Radhakrishnan

Associate Professor, PG & Research Department of Computer Science, Dr. Ambedkar Government Arts College, Chennai.

Abstract: Hypervisor helps to interact with the various machines to communicate on the various platforms. There source allocation on the hypervisor model has important role to govern the transactions. Resources are allocated by the host machine according to the various types of tasks but in the hypervisor its quite different from the existing models. If the resources are allocating properly the transaction may successful in all the times if it is in fail there will be collision among processes. The host machine owns the responsibility for the maintenance of concurrency. The concurrency is determined by the multi threading by allocation proper resource units. Each process must need there source and memory unit to complete the task. The hypervisor model has brought some differences than the traditional machines. But in the security wise features the hypervisor has given less performance son the resource allocation and the scalability units. The more existing model has been initialized on this problem but in the security wise that may not full fill the performances. Resource allocation on the hypervisor models must be analysed and propose some new methodology to enhance the concurrency among the hypervisor transactions. The proper resource units among the hypervisor technology brings better innovation to interact multiple machines in through virtual model with better performances. The importance of resource units and the possibility of the maximum resource allocation on the hypervisor model may be analyses for the modern hypervisor technology.

Keywords: Hypervisor, Concurrency, Resource allocation, Network Security, Cloud hypervisor, Distributed Allocations.

Introduction

The resource allocation among the multiple process much helpful to fulfill the concurrency speed[1].The existing model introduces the DCM model to enhance some concurrency with respect to the scalability properties. It's the act of managing and running multiple computation at the same time but not simultaneously. In contrast with the parallelism, concurrency has done their jobs in the single instance. But however, there are lot of problems over the concurrency with in the same instance which may lead to the severe degrade performances in the process. There are lot of improper synchronization affects the latency problems [1]. To enhance the latency, issue the proper resource allocation is must among the process. The resource allocation among the multiple process much helpful to fulfill the concurrency speed [1]. The existing model introduces the DCM model to enhance on concurrency with respect to the scalability properties. Concurrency has helped to maintain the multiple computations at the same time in the same instant. In the modern programming the concurrency is helping to increase the degree of multi programming.

There source allocation has the main factors behind the sync and the concurrency parts. There source utilization may have more percentage to complete the concurrent process in the sync method [2].So, the resource allocation among the all nodes in the synchronization in the timely manner by some clocks. There are so many existing algorithms proposed many methodologies over the synchronization.

The services on the hypervisors satisfied by the underlying operating systems helped to maintain the concurrent process. The synchronization on the hypervisor is implemented on the cloud architecture which may implemented in the hots system or sometimes in the guest machines.

Tuijin Jishu/Journal of Propulsion Technology
ISSN: 1001-4055
Vol. 47 No. 03 (2026)

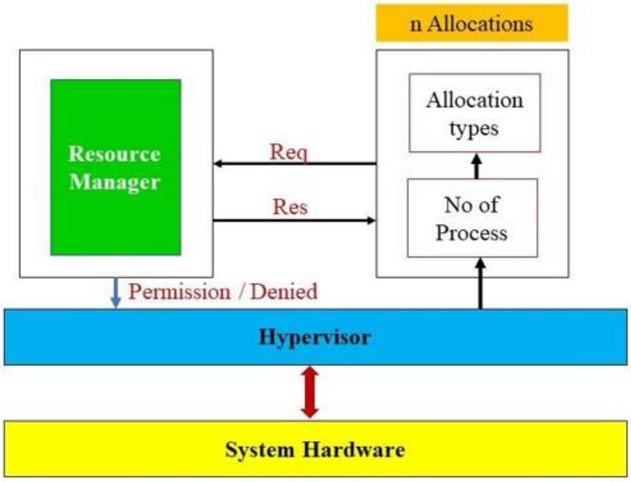

Fig.1.1ResourceAllocationinHypervisor

Hypervisor, some piece of software code to helpful in the cloud paradigm to enable the sharing multiple resources. The resources are divided into the cloud together more concurrent process on the virtual machines. Many of the operating systems were communicated through the hypervisor units. It acts like a layer between the original and the guest.

Whatever process has done through the cloud, hypervisor much helpful to fulfill the resource allocation parts. This is known as soft resource allocation in the virtual computing especially in the cloud and distributed paradigm [3]. Both the hypervisors much helpful to allocate the soft resources through the cloud. Security part of the hypervisor has very important role in the cloud and distributed paradigms. The more security has the more efficient part of the successful process completion. Many of the existing mechanism has enhanced various security and authorization aspects in the hypervisor unit. The resource allocation in the hypervisor unit helped to conduct the concurrency without any latency [3].

The services on the hypervisors satisfied by the underlying operating systems helped to maintain the concurrent process. The synchronization on the hypervisor is implemented on the cloud architecture which may implemented in the hots system or sometimes in the guest machines.

I. Backgroundandrelatedwork

A. Optimal Resource Allocation:

The optimal resource allocations are made by some algorithm patterns which helps to make the file systems. The algorithms are working in the background to make the efficient resource usage. The resource directed algorithms helped to map the function to get the proper resource management. The optimized analysis of the process to run the algorithms in the background process. The file allocation process addresses the algorithm function to help the concurrency issues in the systems. According to the algorithm units file allocations are made in the entire file system units [4]. The communication cost is the main consideration on the resource allocation units among the distributed paradigms. We have to maintain the concurrency logs among the resource allocation units.

B. Partitioned Scheduling:

Partitioned Scheduling in the distributed system helps to maintain the proper resource allocation among the processes. These make the critical physical systems to govern the concurrency in the proper way. The overload property among the distributed systems which address the mapping of the synchronizations. The critical systems allow to maintain the resource units according in the order of critical process units. The tasks of the machine units have arranged in the format of resource allocation orders [4].The existing algorithm helped to arrange the machine synchronization to maintain the concurrency units among the list of scheduled process. The contribution of the process concurrency is always maintained by the proper resource allocation algorithms.

C. **Critical Systems:**

Critical system in the existing models which implement the resource allocation by using the existing model algorithms. The tasks are arranged in the priority list which makes the urgent of the resource needs. The critically need of the time the algorithm determines the allocations. The tasks maybe completed in the concurrency point by based on the resource allocations. The real time systems may have variety of scheduling units for the task completion on the priority basis [5]. Such cost varies on the different distributed system by holding the proper resource allocation lead into the concurrency points. The resource allocations are maintained in the existing systems by their existing algorithms.

D. **Decentralized Units:**

Many of the decentralized algorithms helped to maintain the concurrency point of the units which helped to discuss the resource units on the system. The algorithm helps to allocate the proper resource units for all the tasks on the implemented priority lists[5]. The decentralized algorithms used to optimize the all units of the tasks in the priority lists. The priority lists may be decided by the units of the all the algorithm results in the decentralized systems. The amount of resource allocations in the decentralized part by the priority units in the system units [6]. The decentralized systems have been working in the concurrency by the proper resource usage.

Existingmethodology

A. **Priority Based Allocation:**

The nodes of the systems are interconnected by the network communications which we want to make the resource units for the concurrency. The concurrency has to be maintained for all the sectors of the resource units in the priority basis. The schedules on the various tasks on the scheduling part to allocate in the resource part. The maximum resource allocation in the nodes of the system in the communication is determined by the priority basis. The resource allocation is based on the common tasks for the process synchronization. The synchronization on the process units is delivered in the concurrency point of view which leads to success completion. The existing algorithm this model is based on the application point of view. The minimum values for the job pool have to be assigned. According to the job pools value the resource allocation is set to fix for the job completion. Once the job polis set then the resource has to be assigned for their tasks [5].

B. **Resource Sharing Architecture:**

The new architecture has proposed to maintain the resource allocation for the distributed systems. The economic based architecture is maintained for the proper resource units for the process units. The virtual currency for the all the process based on the processs contribution. The contribution on the process is explained for the concurrency points. In every point of the view concurrency has to maintained in the resources sharing. The resource sharing on the job pool for the process completion in the existing architecture [6]. The architecture on this model is explained for the producer and the consumer units. The goal of the architecture is designed to allocate based on the minimal point of the usage. The demand is always carried out to follow the allocations. The resource market is proposed to maintain the architecture design. This architecture model is used to explain that how to use the minimal usage according to the economical point of view. The concurrency is saved on the resource allocation process on the job pool.

C. **Strategies of Resource Sharing:**

Handling resources for the task list on the existing machines which based on some RSA implementation. The difficult of resource sharing for the generic solutions for the distributed cloud platforms. There was a risk of sometime to achieve all the concurrency units for the task lists. The optimization of cloud resource on the existing models which based on some algorithm view[7]. The virtual resource on the various task list forth existing model to maintain the concurrency for the task completions. The generic solution for the distributed system based on the existing algorithm solutions to produce the task list in priority based. The interaction between the various process on the distributed system must be maintained for the resource allocation plan. The planning of the virtual resource on the cloud units should be traced on their existing model which concerned in the way of telling concurrency. The concurrency of the model based on full and full satisfaction of the task priority for their process completions. The selection strategies for the resource unit sis judged here by the RSA units of the existing methods [7].

D. Resource Utilization on job pool

Resource utilization among the task list on the various existing model is to be little hard to achieve the concurrency. The model has some methodologies for the task list for the completion. The tasks are arranged according to the priority list. Many strategies of optimization have been followed for the resource allocation for the concurrency maintaining in the job pool. The job pool has many priority lists for job completion to complete the task. The algorithm model has been followed to map the virtual resources to allocate for the participating nodes. Some complex methodologies for the participating nodes expect some security to allocate the virtual resource to complete the task. The infrastructure for the resource units allocating on the cloud units have to be performed in the concurrency view. The concurrency points on every complex system are based on the resource sharing only [8]. Some of the data centre need extra resources units to done the process completion n for the job pools.

The existing algorithms have done the in process on the job pool according to their process priorities. The locations and the priorities of the job pool is taken care by the existing model by their own methodologies.

E. Independent Critical Units

There is a problem of critical system on the allocating independent process for the completion. The synchronization of resource requirements for the tasks is very hard to allocate. There is a buffer method to eliminate inter task dependencies to avoid resource collisions. The building blocks of the model is analysed to allocate the resource for the job completion. The scheduling on the partial side for the critical system is to achieve the synchronization by their job pool. The job metrics is measured by the resource unit on the job priorities. The migration points, rescheduling units on the models have been carried out by the distributed existing algorithms. Critical process and their process metrics are calculated for allocating proper resource units [8]. The inversion problem is found on the unallocated resource for the process completion. The critical mode on the model is analysed to measure the synchronization. The synchronization methods for concurrency are less complicate on the critical system. Some of the algorithms have helped to furnish the concurrency units. For this critical distributed system, some of resource architecture has been built to complete.

II. Importance Of Resource Allocation In Distributed And Cloud

There source allocation in distributed paradigm helps to govern the consistency at the process completion. There is multiple resource are need to allocate for the various process task lists. The importance for the distributed and cloud platforms is concurrency in the process scheduling and completion. The allocating workload is determined by the various characteristics of the resource management. The resource allocation has been made by the units in the all system according to their priority lists. If the priority is given in the critical system which is denoted by the existing machine algorithms. There are various utilities judged the resource management for the priority lists. However, there is some certain resource allocation characteristics helped to manage the resource allocating mechanism. They are,

- Resource discovery
- Resource price & Accounting
- Resource prediction
- Resource Scheduling

A. Efficient Data Share:

When the resources are shared in the distributed platforms, the data also been shared through the various points. The distributed resource allocation is done by the existing distribute algorithms in the networking nods. The data sharing is very important on the any system model to avail and free the resources on the time. The data sharing in the unit of distributed computing allows the concurrency of the entire job completion. The data sharing consists of the various aspects to satisfy the process in the resource making. The resources of each and every process of job pool in the task list is being arranged for the efficient resource allocation [9]. These are all done by the existing algorithms of the system model. Every system model has their own algorithms to determine the resource sharing among the process. But the efficient is meant by the only the observation of entire process's concurrency things.

B. Satisfied Scalability:

The scalability of the system is maintained by the reliability of the system model. The system model can have the ability to produce the concurrency of the job lists. The satisfied scalability is meant by the facing the problem of system's ability. The inability of the system is made by the IO device errors, network issues and node crashes [8]. The replication units and the optimization of the system model is determined by the own existing algorithm in the distributed and the cloud units. The resource allocation units are helped to measure the scalability of entire job pool. Every time the scalability of the units is getting differ with various occasions on the time slices. The response changes in the performances of the system have monitored by the various resources. These resources are allocated in the fixed time interval without leading into some starvation issues [9]. The growth of the resource allocation is measured in differ by the changes in increasing and decreasing performances.

C. Long Lasting Reliability:

Reliability of every system is based on the failures of the software or hardware on the process transactions. The transaction of every job pool has some reliability of the failures of the execution which affected by the improper resource allocation. Sometime the availability of the existing system is performed by the required time period in the specific period. The portability of model is based on the reliability which is meant that no repair required on the model. The performance specification on the reliability should be reliable on longtime which achieved by the resource allocation units. The resource allocation has established in the algorithm parts which lead into the long-time durability. The concurrency on the existing system model is succeeded by the resource units on the specific time unit. The long lasting reliable is based on the cloud resources which locates the specific time units. The specific duration for the resource units on the cloud part is mainly focused on the business specifications.

D. Reducing Latency Issues:

The resource allocation in the distributed and cloud units which lead into the reducing the time delays. The delay part is very important in the execution speed. The main advantage of the reducing latency issue is focused to increase the process completion speed. Basically, the speed of the job pool completion is getting by the hardware resources on the right time. The latency issues are reduced in the cloud and efficient distributed system by the proper resource management plans. If there sources are managed in the right-way then the process of all transaction lead in to the good concurrency point. So, the latency issues are reduced by the way of getting the good resource allocation units. The availability and the maintainability is also measured in the better range while the allocation on the proper way. This is some kind of motivation to propose various resource architecture to fulfill the resource unit part.

III. Comparison & Results

The comparison for the various characteristics to determine the resource allocation in the hypervisor models is recorded in the table 5.1. The table has showed with five different characteristics which are important to recognise the efficient resourceallocationmodels.Thetable5.1hassplitintothetwo parts as Type–I and Type–II hypervisor. For this experiment we have taken the type I hypervisors are KVM, RHEV, XEN and like as same type -II hypervisors are MSVir, OracleVM and VMFusion. So, the characteristics are Data sharing, Scalability, Reliability and latency issues handling to check the performances. Initially we have listed 1000 transactions and those transactions are segregated into successful and unsuccessful parts. The type-I hypervisors are given much better performances in the table 5.1. In the data sharing characteristics 628 transactions are successfully done by the host and the remaining transactions of not done or partially failed due to some improper resource allocations. In the scalability wise little less than 50% percentage of transactions satisfies the better scalability. But more than 50% percentage of transactions are unsuccessful in the type I hypervisors. Latency issues handling in the type I hypervisors has given far better results in the range of 80% less than 20% transactions are getting impact of affecting by the resource allocation. Out of 1000 transactions in the type 1 hypervisors with related to the reliability property 752 transactions are done as successfully whereas only 24% of transactions are getting affected. These are the statistics of only type I hypervisors. They are only gives much far performances with compared to the type II hypervisors.

Table5.1Comparison of Type1 and Type2 Hypervisors

CHARACTERISTICS	TYPE–I (Outof1000 transactions)		TYPE–II(Outof1000 transactions)	
	Successful	success ful	Successful	success ful
Data Sharing	628	372	458	542
Scalability	489	511	251	749
Reliability	752	248	626	374
Latency issues handling	815	185	648	352

In the type II hypervisors has given same performances like the type I but in the performances wise type I is listed as much better with compared to type II. First in the data sharing property, 458 transactions out of 1000 transactions in the successful stage where as remaining transactions are listed in the unsuccessful part in the table 5.1. In the scalability issuesinthetypeIIhypervisorsonly25%transactionsaredoneinthe type II as successful whereas remaining all the transactions of 75% as unsuccessful due to some network changes like not supporting hardware structures. But in the reliability wise type IIhypervisorgivesbetterresultsof60%inthesuccessfulrange. The type II hypervisors have less cost with compared to the type I hypervisor. This economical savings gives the results in the performances. Handling of latency issues in the type II hypervisors has given the 64% over the successful rate where only 36% are recorded in the unsuccessful range in the table 5.1.

The figure 5.1 is the graph variations of the table 5.1 which clearly said about the type II hypervisors give less performances than the type I hypervisors. However, the type I hypervisor sare little cos in the market but for the well satisfies business people prefers the typeI hypervisor. The graph(figure 5.1) has given the high peak for all the successful transaction. But we have to take the comparison with the type I and type II hypervisors not for the successful and the unsuccessful. In the type II part successful ranges are getting high peak in the figure

5.1 but with compared to the type I, they are little less peak. These are all recorded in the figure 5.1. In the last peak of unsuccessful in the type II, the scalability is getting more impact of negative range. The seare like happened due to some of the improper resource allocation and not well structured for the resource maintaining.

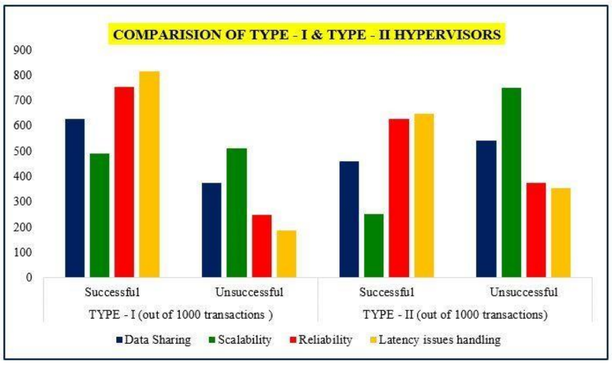

Figure6.1 Comparison of Type 1 and Type 2 Hypervisors

Thetype2 hypervisors in the graph shows the less peak in the successful range in the defined characteristics. The proper structures for the resource allocation makes the hypervisor performances in the various ranges in high successful peaks and at the same time they will get high paid economical hypervisors.

IV. Conclusions

The resource allocation for all the system is very important to gain better performances. So, the proper resource allocation is maintained by the proper structure of the system. The proper resource allocation is happened by the various characteristics like data sharing, scalability, reliability and the latency issue clearance. The type I hypervisor have better performances among the type II hypervisor transactions. This shows that the various experiments have been take between those hypervisors. The proper allocations have been maintained by the mentioned characteristics. The proposed research methodologies maybe implemented in the various host and network models the efficient usage will get more enhancement. New methodologies have to be proposed to get good resource planning among the various cloud and distributed sites to motivate the economical and business side gains. The proper resource planning should give the more part to enhance the concurrency things in the hypervisor models. However, the type I hypervisor gets impact of better performances but the concurrency things should evolve enhancing the performances in the hypervisors. The resource allocation is getting more enhancing by increasing the concurrency in the all types of participated system models.

References:

[1] Li, Shih-Wei, et al. "A secure and formally verified Linux KVMhypervisor."2021 IEEE Symposium on Security and Privacy (SP).IEEE, 2021.

[2] Kurose, James F., and Rahul Simha. "A microeconomic approach tooptimal resource allocation in distributed computer systems." IEEETransactions on computers 38.5 (1989): 705-717.

[3] Lakshmanan, Karthik, et al. "Resource allocation in distributed mixed-criticality cyber-physical systems." 2010 IEEE 30th InternationalConference on Distributed Computing Systems. IEEE, 2010.

[4] Huang, Yin-Fu, and Bo-Wei Chao. "A priority-based resourceallocation strategy in distributed computing networks." Journal ofSystems and Software 58.3 (2001): 221-233.

[5] Singh, Upendra, et al. "Scalable priority-based resource Allocation scheme for M2M communicationin LTE/LTE-A network."Computersand Electrical Engineering 103 (2022): 108321.

[6] Anuradha, V. P., and D. Sumathi. "A survey on resource allocation strategies incloud

computing."International Conference on Information Communication and Embedded Systems (ICICES2014). IEEE, 2014.

[7] Sun, Lu, Liangtian Wan, and Xianpeng Wang. "Learning-basedresource allocation strategy for industrial IoT in UAV-enabled MECsystems." IEEE Transactions on Industrial Informatics 17.7 (2020):5031-5040.

[8] Selvaprabhu, Poongundran, et al. "Priority-based resource allocationand energy harvesting for WBAN smart health." WirelessCommunications and Mobile Computing 2022 (2022).

[9] Pesonen, Mauno, et al. "Assessing the priorities using A'WOT among resource management strategies at the Finnish Forest and Park Service."Forest science 47.4 (2001): 534-541.

[10] Chao, Hung-Lin, and Pao-Ann Hsiung. "A fair energy resource allocation strategy formic rogrid."Microprocessors and Microsystems 42 (2016): 235-244.