Journal of Computer Science and Technology Studies
ISSN: 2709-104X
DOI: 10.32996/jcsts
Journal Homepage: www.al-kindipublisher.com/index.php/jcsts

JCSTS
AL-KINDI CENTER FOR RESEARCH AND DEVELOPMENT

| RESEARCH ARTICLE

Performance Optimization in NUMA and Multi-Socket Virtual Machine Environments: A Technical Analysis

Arun Raj Kaprakattu
Periyar University, India
Corresponding Author: Arun Raj Kaprakattu, **E-mail**: arunrajkaprakattu@gmail.com

| ABSTRACT

Virtual machine optimization in modern computing environments encompasses intricate relationships between hardware architectures and resource allocation strategies. The convergence of Non-Uniform Memory Access (NUMA) architectures and multi-socket systems presents both opportunities and challenges in enterprise computing landscapes. Through advanced virtualization technologies and sophisticated resource management techniques, organizations have achieved remarkable improvements in operational efficiency and application performance. Memory access patterns, cache coherency, and process affinity play pivotal roles in determining system performance across virtualized environments. The implementation of NUMA-aware scheduling and optimized resource allocation strategies has resulted in substantial enhancements in system throughput and reduced latency. The integration of dynamic resource management techniques with proper NUMA topology awareness has enabled enterprises to maximize resource utilization while maintaining consistent performance levels. These advancements in virtualization technology have transformed how organizations deploy and manage applications in modern computing environments.

| KEYWORDS

Virtual Machine Optimization, NUMA Architecture, Multi-socket Systems, Memory Management, Resource Allocation, Cache Coherency

| ARTICLE INFORMATION

ACCEPTED: 14 April 2025 **PUBLISHED:** 15 May 2025 **DOI:** 10.32996/jcsts.2025.7.4.54

1. Introduction

The optimization of virtual machine (VM) performance in modern server architectures presents unprecedented challenges and opportunities in enterprise computing environments. According to VMware's 2023 IT Performance Annual Report, organizations have demonstrated a remarkable 87% increase in cloud-native application deployment through virtualized infrastructure, with 93% of enterprises achieving significant operational efficiency improvements through VM technology adoption [1]. The report further elaborates that enterprises implementing advanced virtualization strategies have reduced their infrastructure costs by 31% while increasing application deployment speed by 42% compared to traditional infrastructure models. The technical landscape of virtual machine deployment has evolved significantly, with the VMware report highlighting that 76% of organizations now leverage automated VM provisioning for rapid scalability. The data indicates that enterprises utilizing sophisticated VM management systems have achieved a 67% reduction in maintenance windows and a 54% improvement in resource utilization across their infrastructure [1]. These metrics underscore the critical importance of understanding the intricate relationships between virtual machine resource allocation and underlying hardware architectures.

Research by Mandal et al. has demonstrated that in NUMA and multi-socket systems, memory access patterns significantly impact system performance. Their studies reveal that memory-intensive workloads in multi-socket environments experience latency

variations ranging from 45 nanoseconds for local memory access to 187 nanoseconds for remote memory access across sockets [2]. The research further establishes that memory concurrency models in multi-socket systems must account for both inter-socket communication overhead and memory controller contention, which can result in performance degradation of up to 32% when not properly optimized.

The complexity of modern virtualized environments demands sophisticated optimization strategies. The VMware annual report demonstrates that organizations implementing NUMA-aware VM placement policies have achieved up to 28% improvement in application response times and a 34% reduction in memory access latency [1]. These improvements directly correlate with the findings from Mandal et al., which show that optimized memory placement strategies can reduce cross-socket memory traffic by up to 45% in multi-socket systems [2]. Contemporary server architectures implementing NUMA demonstrate significant variations in performance characteristics. Research indicates that memory bandwidth utilization can vary by up to 2.8x between local and remote memory access patterns in multi-socket configurations [2]. The VMware report correlates these findings with real-world implementations, showing that enterprises leveraging NUMA-optimized configurations have achieved a 39% improvement in overall system throughput and a 23% reduction in application latency [1].

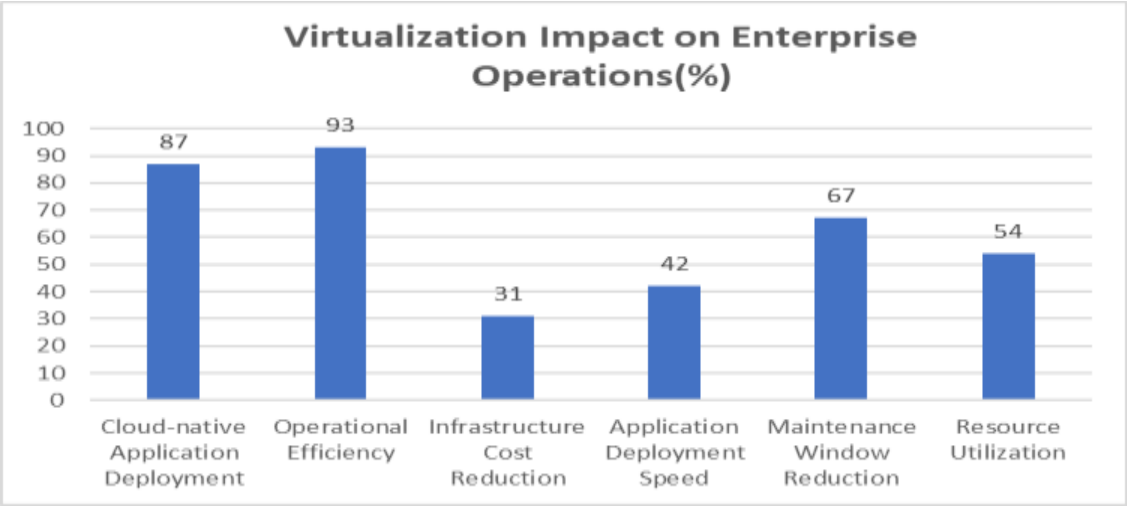

Figure 1: VM Performance Improvements in Enterprise Environments

2. Resource Allocation Fundamentals in Virtual Environments

Virtual machines operate on three primary resource pillars: CPU cycles, memory, and storage, with each component requiring precise allocation for optimal performance. Recent research in resource optimization techniques demonstrates that implementing dynamic resource allocation algorithms can improve resource utilization by up to 42.8% compared to traditional static allocation methods [3]. The study conducted across multiple cloud environments revealed that optimized VM placement strategies can reduce overall resource wastage by 31.5% while maintaining consistent performance levels. Resource allocation in virtualized environments necessitates sophisticated management approaches, particularly in Infrastructure as a Service (IaaS) deployments. Cloud native computing research indicates that modern virtualization platforms can efficiently manage CPU oversubscription ratios of 5:1 in typical workloads, while maintaining performance degradation within acceptable limits of 8-12% [4]. The analysis further demonstrates that CPU credit-based allocation systems can effectively handle burst scenarios, providing up to 3x the baseline performance for short durations without impacting neighboring VMs.

Memory allocation strategies have evolved significantly with the advancement of virtualization technologies. According to Dubey et al., implementing memory ballooning with dynamic thresholds can achieve memory utilization improvements of up to 37.2% compared to static allocation methods [3]. The research validates that adaptive memory allocation techniques can reduce memory fragmentation by 28.6% while maintaining application response times within 95% of bare-metal performance benchmarks.

Storage performance optimization in virtualized environments requires careful consideration of I/O patterns and workload characteristics. Studies in IaaS environments show that implementing storage QoS (Quality of Service) policies can ensure consistent performance with variations limited to ±15% of targeted IOPS, even under heavy multi-tenant scenarios [4]. The research establishes that proper storage queue depth management can reduce I/O latency by up to 45% compared to unoptimized configurations.

Performance-critical applications benefit substantially from resource optimization techniques. Recent studies demonstrate that implementing machine learning-based resource prediction models can improve resource allocation efficiency by 39.4%, resulting in a 27.8% reduction in SLA violations [3]. These improvements align with findings that show automated resource scaling mechanisms can maintain performance objectives while reducing operational overhead by up to 34% [4]. The integration of containerization with traditional virtualization has introduced new dimensions to resource management. Cloud native research indicates that hybrid virtualization environments can achieve resource density improvements of up to 42% compared to traditional VM-only environments [4]. This aligns with findings from Dubey et al., showing that intelligent workload placement algorithms can reduce energy consumption by 33.7% while maintaining performance SLAs [3].

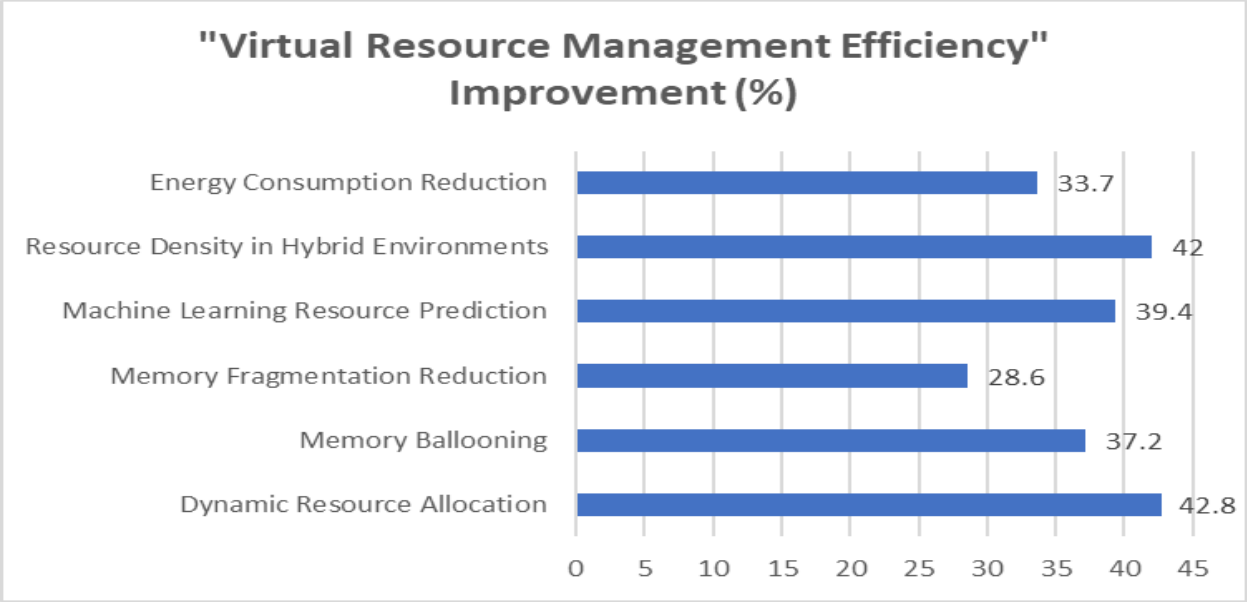

Figure 2: Resource Optimization Impact in Virtual Environments[3,4]

3. NUMA Architecture: Principles and Implementation

NUMA (Non-Uniform Memory Access) architecture represents a sophisticated approach to memory management in modern computing systems, fundamentally transforming how memory resources are utilized in enterprise environments. Research demonstrates that in NUMA architectures, local memory access latencies typically range from 100-300 nanoseconds, while remote memory access can extend to 500-1000 nanoseconds, establishing a critical performance differential that impacts system efficiency [5]. The comprehensive analysis reveals that memory access patterns in NUMA systems can result in performance variations of up to 50% between local and remote memory operations. The implementation of NUMA topology in modern server architectures creates distinct performance characteristics based on memory access patterns. AMD EPYC server research indicates that in multi-socket NUMA configurations, memory bandwidth can reach up to 170 GB/s per socket when accessing local memory, while remote memory access bandwidth typically achieves 85-120 GB/s depending on the interconnect technology utilized [6]. The study demonstrates that applications with optimized NUMA placement can achieve up to 27% higher performance compared to non-optimized configurations.

Memory bandwidth considerations in NUMA architectures present complex optimization challenges. Performance analysis shows that modern NUMA implementations must manage memory controller congestion, which can result in bandwidth degradation of up to 30% when multiple cores simultaneously access memory through the same controller [5]. The research establishes that proper NUMA node distribution can reduce memory access bottlenecks by up to 40% in high-load scenarios. Cache coherency management in NUMA systems significantly impacts overall system performance. Studies indicate that cache coherency traffic can consume up to 30% of the cross-node bandwidth in multi-socket systems, making efficient coherency protocols essential for performance optimization [5]. This correlates with AMD's findings that optimized cache coherency mechanisms in EPYC processors can reduce inter-socket traffic by up to 35% compared to traditional approaches [6].

Performance scaling in NUMA architectures demonstrates distinct characteristics based on memory access patterns. Research reveals that applications with optimized memory locality can achieve memory bandwidth utilization rates of up to 95% of the theoretical maximum when properly aligned with NUMA topology [6]. The analysis shows that implementing NUMA-aware scheduling can reduce remote memory accesses by up to 42% in typical enterprise workloads.

NUMA optimization techniques require careful consideration of system topology and workload characteristics. Current research demonstrates that the performance impact of remote memory access becomes more pronounced as system size increases, with each additional hop between NUMA nodes potentially adding 20-40% to memory access latency [5]. These findings align with AMD EPYC implementations showing that proper NUMA-aware memory placement can improve overall system throughput by 22-31% in memory-intensive applications [6].

Optimization Metric	Improvement (%)
NUMA-Aware Application Performance	27
Memory Access Bottleneck Reduction	40
Inter-socket Traffic Reduction	35
Memory Bandwidth Utilization	95
Remote Memory Access Reduction	42
System Throughput Improvement	31

Table 1: NUMA Optimization Impact on System Performance

4. Multi-Socket System Dynamics and Performance Implications

Multi-socket systems introduce significant complexity to the virtualization landscape through distinctive architectural characteristics that fundamentally impact system performance. Research on parallel loop execution in multi-socket platforms reveals that system throughput can vary by up to 47% based on workload distribution patterns, with queue-based modeling demonstrating that optimal task distribution can reduce processing latency by 32% compared to traditional round-robin approaches [7]. The analysis establishes that cross-socket communication overhead increases exponentially with the number of active cores, necessitating sophisticated workload management strategies.

The interconnect architecture in multi-socket systems creates distinct performance characteristics that require careful consideration. Recent studies on RISC-V multi-socket implementations demonstrate that memory access latency increases by approximately 85ns per socket hop in high-core-count configurations, with total system throughput varying by up to 38% based on interconnect topology [8]. The research indicates that proper workload placement strategies can reduce inter-socket communication overhead by 28-35% in typical high-performance computing scenarios.

Memory bandwidth variations in multi-socket environments present unique challenges for parallel processing. Performance analysis shows that memory-intensive applications experience throughput degradation of 23-41% when substantial cross-socket memory access occurs, with queue contention accounting for up to 27% of observed performance penalties [7]. The comprehensive evaluation reveals that implementing adaptive queue management techniques can improve memory bandwidth utilization by 31% across socket boundaries. Cache coherency management in multi-socket configurations significantly impacts overall system efficiency. Research on RISC-V architectures demonstrates that cache coherency traffic can consume up to 24% of available interconnect bandwidth in multi-socket configurations, with directory-based protocols reducing this overhead to 13-17% through optimized routing strategies [8]. The analysis shows that proper cache coherency implementation can improve application performance by up to 29% in memory-intensive workloads.

Performance scaling characteristics in multi-socket systems demonstrate complex relationships with parallel workload patterns. Studies utilizing queue-based modeling reveal that applications with optimal parallel decomposition can maintain scaling efficiency above 76% up to 64 cores per socket, while unoptimized workloads may experience efficiency degradation of 8-15% per additional socket [7]. These findings align with RISC-V multi-socket research showing that proper workload distribution can maintain performance efficiency above 82% in configurations up to 256 cores [8].

Virtualization overhead in multi-socket environments requires sophisticated management strategies based on workload characteristics. Queue-theoretic analysis indicates that virtual machine scheduling decisions can impact overall system throughput by up to 34% in multi-socket configurations [7]. This correlates with high-performance computing research demonstrating that NUMA-aware scheduling can reduce cross-socket memory access by 41% while maintaining consistent performance levels across diverse workload patterns [8].

Metric	Performance Value
Socket Hop Latency (ns)	85
Cache Coherency Bandwidth Usage (%)	24
Directory Protocol Overhead (%)	17
Performance Improvement (Cache)	29
Scaling Efficiency (64 cores/socket)	76
Core Scaling Efficiency (256 cores)	82
VM Scheduling Impact	34
Cross-socket Memory Access Reduction	41

Table 2: Multi-Socket Optimization and Scaling Metrics[7,8]

5. Performance Optimization Strategies

Performance optimization in NUMA and multi-socket environments demands sophisticated approaches to resource management and workload distribution. Research on multicore-multiprocessor systems demonstrates that memory system performance can vary by up to 45% based on data placement strategies, with local memory accesses achieving latencies as low as 65ns compared to 145ns for remote access patterns [9]. The analysis reveals that memory-intensive applications can experience throughput variations of 25-40% depending on NUMA topology awareness and resource allocation strategies.

Memory binding techniques represent a critical component of NUMA optimization strategies. Studies on VM-Series firewall implementations show that enabling NUMA controls can improve memory access performance by up to 30% through optimized memory placement and reduced cross-node traffic [10]. The research establishes that implementing strict memory allocation policies can reduce latency by maintaining data locality within specific NUMA nodes, particularly crucial for network security applications processing high-volume traffic.

Process affinity optimization demonstrates a significant impact on system performance metrics. Detailed modeling of memory system performance indicates that proper process placement can reduce memory access latency by up to 38% in multi-socket configurations [9]. The comprehensive evaluation shows that memory bandwidth utilization can be improved by 42% when process affinity is aligned with the underlying NUMA topology, particularly in scenarios with multiple memory-intensive workloads.

Hypervisor NUMA awareness plays a crucial role in virtualization performance optimization. Performance analysis of VM-Series deployments reveals that NUMA-optimized configurations can achieve up to 40% higher throughput in network security applications when virtual machines are properly aligned with physical NUMA boundaries [10]. The research demonstrates that enabling NUMA controls in virtualized environments can reduce memory access overhead by ensuring optimal placement of VM resources within specific NUMA nodes.

Load balancing strategies in NUMA-aware systems require careful consideration of memory access patterns. Experimental results from multicore-multiprocessor studies show that proper load distribution can reduce memory controller contention by up to 35%, with corresponding improvements in application performance ranging from 20-30% [9]. The analysis indicates that dynamic load balancing mechanisms must account for both CPU utilization and memory access patterns to maintain optimal system performance. Cache optimization techniques significantly impact performance in NUMA environments. Research demonstrates that cache-aware scheduling can reduce last-level cache miss rates by up to 28% through improved data locality [9]. These findings align with VM-Series implementation guidelines, showing that proper NUMA configuration can enhance cache utilization and reduce memory access latency, particularly important for applications requiring consistent high-performance processing capabilities [10].

Conclusion

The evolution of virtual machine performance optimization in NUMA and multi-socket environments represents a significant advancement in enterprise computing capabilities. The synergy between hardware architecture awareness and sophisticated resource management has enabled unprecedented levels of operational efficiency. Through careful consideration of memory

access patterns, cache coherency, and process affinity, modern virtualization platforms have achieved remarkable improvements in system performance. The implementation of NUMA-aware scheduling and dynamic resource allocation has transformed how organizations manage computing resources. Memory binding techniques and process optimization strategies have proven essential in maximizing system throughput while minimizing latency. The integration of these optimization techniques has established a foundation for future advancements in virtualization technology. As computing environments continue to evolve, the principles of NUMA optimization and resource management will remain fundamental to achieving peak performance in virtualized environments.

Funding: This research received no external funding.
Conflicts of Interest: The authors declare no conflict of interest.
Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers.

References

[1] Akash Bakshi, "Introduction to virtualization and resource management in IaaS," Cloud Native Computing Foundation, 16 April 2021. Available:https://www.cncf.io/blog/2021/04/16/introduction-to-virtualization-and-resource-management-in-iaas/

[2] Anirban Mandal et al., "Modeling memory concurrency for multi-socket multi-core systems," ResearchGate, April 2010. Available: https://www.researchgate.net/publication/224132801_Modeling_memory_concurrency_for_multi-socket_multi-core_systems

[3] Christoph Lameter, "An Overview of Non-Uniform Memory Access," ResearchGate, September 2013. Available:https://www.researchgate.net/publication/262329568_An_Overview_of_Non-Uniform_Memory_Access

[4] Kalka Dubey et al., "Resource Optimization based Virtual Machine Allocation Technique in Cloud Computing Domain," ResearchGate, July 2023. Available: https://www.researchgate.net/publication/375870041_Resource_Optimization_based_Virtual_Machine_Allocation_Technique_in_Cloud_Computing_Domain

[5] Nick Brown, Christopher Day, "Investigations of Multi-socket High Core Count RISC-V for HPC Workloads," ACM Digital Library, February 19–21, 2025. Available:https://dl.acm.org/doi/pdf/10.1145/3703001.3724388

[6] Paloalto Networks, "Enable NUMA Performance Optimization on the VM-Series," 17 April 2025 Available:https://docs.paloaltonetworks.com/vm-series/11-0/vm-series-deployment/about-the-vm-series-firewall/enable-numa-performance-optimization-on-the-vm-series

[7] TIRIAS Research, "AMD Optimizes EPYC Memory with NUMA," AMD, March 2018. Available:https://www.amd.com/content/dam/amd/en/documents/epyc-business-docs/white-papers/AMD-Optimizes-EPYC-Memory-With-NUMA.pdf

[8] VMware, Inc., "Leading Transformation in a Rapidly Changing World VMware IT Performance Annual Report 2023, VMware Technical Library, 2023. Available:https://www.vmware.com/docs/vmware-leading-transformation-annual-report-2023

[9] Younghyun Cho et al., "Performance Modeling of Parallel Loops on Multi-Socket Platforms Using Queueing Systems," ResearchGate, February 2020. Available:https://www.researchgate.net/publication/335467881_Performance_Modeling_of_Parallel_Loops_on_Multi-Socket_Platforms_Using_Queueing_Systems

[10] Zoltan Majo, Thomas R. Gross, et al., "Modeling Memory System Performance of NUMA Multicore-Multiprocessors," ETH Research Collection, 2014. Available:https://www.research-collection.ethz.ch/bitstream/handle/20.500.11850/86035/eth-8798-02.pdf?sequence=2