論文 / 著書情報
Article / Book Information

題目(和文)	オーバーサブスクライビングスケジューリング： HPCシステムにおける多様化するワークロードの効率性と応答性の両立
Title(English)	Oversubscribing Scheduling: Balancing Efficiency and Responsiveness in HPC Systems with Diverse Workloads
著者(和文)	南将平
Author(English)	Shohei Minami
出典(和文)	学位:博士(理学), 学位授与機関:東京科学大学, 報告番号:甲第233号, 授与年月日:2025年3月26日, 学位の種別:課程博士, 審査員:遠藤 敏夫,増原 英彦,坂本 龍一,安永 憲司,脇田 建
Citation(English)	Degree:Doctor (Science), Conferring organization: Institute of Science Tokyo, Report number:甲第233号, Conferred date:2025/3/26, Degree Type:Course doctor, Examiner:,,,,
学位種別(和文)	博士論文
Category(English)	Doctoral Thesis
種別(和文)	論文要旨
Type(English)	Summary

（博士課程）
Doctoral Program

論 文 要 旨

THESIS SUMMARY

系・コース： Department of, Graduate major in	数理・計算科学 系 数理・計算科学 コース	申請学位（専攻分野）： Academic Degree Requested	博士 Doctor of （ 理学 ）
学生氏名： Student's Name	南 将平	審査員主査： Chief Examiner	遠藤敏夫

要旨（英文 800 語程度）
Thesis Summary（approx.800 English Words ）

High-Performance Computing (HPC) systems face significant challenges in managing increasingly diverse workloads, particularly with the growing prevalence of interactive jobs alongside traditional batch-oriented scientific simulations. While these systems have historically excelled at handling batch workloads, they now struggle to efficiently support interactive computing needs, which are essential for AI/ML development among other use cases, while maintaining system efficiency. This dissertation evaluates Oversubscribing (OSub), a resource management approach that enables multiple jobs to share computational resources through time-division multiplexing.

Despite OSub's successful adoption in other computing domains such as operating systems and cloud computing, its implementation in HPC environments has been historically limited. This hesitation primarily stems from concerns about performance variability - HPC applications are often highly optimized for maximum performance, and any interference between jobs sharing resources could lead to unpredictable execution times and significant performance degradation. However, modern HPC workloads increasingly show intermittent resource usage patterns, particularly in interactive AI/ML development, making OSub potentially more viable. The core of our research is to provide systematic verification of OSub effectiveness in HPC environments, demonstrating that it can simultaneously provide users with immediate resource access and reduced waiting times, while maintaining acceptable system throughput and enabling simplified management for operators.

Our research contributions are structured around four major components. First, we conduct a comprehensive quantitative analysis of performance impact when multiple applications share computational resources under OSub scheduling. Using both parallel and sequential applications from the HPC benchmark suite, we investigate distinct aspects of performance degradation in these two categories. For parallel applications, we analyze synchronization overhead and communication patterns, demonstrating how partial and full oversubscription scenarios affect their performance. For sequential applications, we examine resource contention patterns and develop predictive models using hardware performance counters to characterize application compatibility under shared execution. Through this dual analysis approach, we provide crucial insights into the feasibility and limitations of resource sharing in HPC environments, offering a foundation for effective OSub implementation.

The second major contribution involves system-level evaluation using real workload traces from production supercomputers to assess the basic effectiveness of OSub scheduling. Through extensive simulation studies, we analyze system-wide metrics such as job waiting times and overall throughput to demonstrate the practical benefits and challenges of implementing OSub in real-world scenarios.

The third major contribution focuses on evaluating OSub's effectiveness for interactive workloads. Our innovative simulation framework incorporates detailed models of interactive job behavior derived from extensive analysis of production system logs. Our results show that OSub scheduling can provide immediate responsiveness for interactive jobs while maintaining acceptable performance levels for batch workloads, even in systems where interactive jobs constitute a significant portion of the workload.

The fourth contribution validates the effectiveness of OSub scheduling through comprehensive experiments in a physical computing environment. By developing and deploying a prototype system integrated with a production job management system, we demonstrate that the benefits of OSub observed

in simulations are achievable in real-world settings. Our experimental results confirm that OSub can effectively reduce waiting times while maintaining acceptable performance levels, even when accounting for real-world factors such as process synchronization overhead and system software interactions. This physical validation provides concrete evidence that OSub scheduling is a viable solution for production HPC environments, bridging the gap between theoretical analysis and practical application.

Key contributions of our research include three categories. Firstly, this research has demonstrated OSub's capability to simultaneously satisfy both user productivity and operator efficiency. More concretely, users benefit from eliminated waiting time with near-immediate job execution (within one second in most cases). Also, they can plan effectively with predictable and manageable performance trade-offs. In operators' side, they maintain system-wide performance impact within 2.7-5.2% even with aggressive oversubscribing. And they benefit from simplified management through single parameter (M) control, eliminating complex queue ratio (R) tuning that varies from 2-18% based on workload mix.

Secondly, this research has shown results of empirical assessment of OSub effects by harnessing workload data obtained from production environments. The assessment includes the first systematic analysis using real production workload traces and actual interactive job characteristics. The analysis of data of the interactive jobs shows that most busy periods last <30 seconds with around 10 repetitions. Next, the experiments of gang scheduling show it mitigates performance degradation of parallel applications around by 20%, while they can suffer more than 50% without gang. Experiments using physical system implementation validates findings via simulations.

Thirdly, this research has validated scheduling scenario where interactive jobs are dominant, which is expected to be more popular in future. Even with 76% interactive workloads, OSub scheduling maintains more stable performance with up to 4.8 times maximum slowdown. Also, OSub scheduling achieves robust performance to varying workload mixes without system parameter tuning.

This dissertation concludes by discussing the broader implications of our findings for future HPC system design and management. As computational workflows continue to diversify, approaches like OSub scheduling become increasingly relevant for maintaining efficient and responsive HPC environments. We identify several promising directions for future research: establishing performance-cost trade-off models that enable flexible resource allocation policies based on performance guarantees, developing comprehensive workflow optimization approaches that consider entire computational processes rather than individual jobs, and integrating power-aware computing strategies to address both environmental sustainability and operational costs in large-scale computing facilities.