



A Spectral Clustering-Based Decentralized Self-Adaptation Framework for Cloud-Native Service-Oriented Applications

REENA SHARMA

Department of Computer Science Engineering, Lovely Professional University, Punjab-144 402, India.

Abstract

As cloud-native architectures become more prevalent, service-based applications demand robust, scalable, and autonomous management strategies. This paper presents a decentralized self-adaptation mechanism using spectral clustering to dynamically manage services in a cloud environment. The proposed framework enables services to adapt to changing workloads and environmental conditions without centralized control, promoting resilience, scalability, and efficiency.



Article History

Received: 10 May 2025
Accepted: 27 July 2025

Keywords

Self-Adaptation, Robust, Scalable, Autonomous Management Strategies

Introduction

Service-based applications in the cloud are often composed of numerous loosely coupled micro-services. These systems require mechanisms to handle failures, manage resources efficiently, and adapt to varying workloads. Traditional centralized adaptation approaches suffer from bottlenecks and single points of failure. In this study, we propose a decentralized approach utilizing spectral clustering for service grouping and adaptive decision-making.

Related Work

Most adaptation frameworks rely on centralized monitoring and control, which introduces latency and hinders scalability. Decentralized approaches have gained traction, especially with the rise of edge computing. Clustering techniques like k-means have been explored, but spectral clustering provides superior performance in identifying non-linear relationships among services.

Proposed Framework

Architecture Overview

The proposed framework includes the following key components:

- **Monitoring Agents:** Collect metrics (latency, throughput, resource usage).
- **Spectral Clustering Engine:** Groups services based on similarity in behavior and performance.
- **Adaptation Manager:** Each cluster autonomously adjusts configurations using local policies.

Spectral Clustering

Spectral clustering uses the eigenvalues of a similarity matrix to reduce dimensions before clustering, making it ideal for detecting complex patterns in high-dimensional cloud metrics.

CONTACT Reena Sharma ✉ r47607106@gmail.com 📍 Department of Computer Science Engineering, Lovely Professional University, Punjab-144 402, India.



Table 1: Comparison of Clustering Techniques

Technique	Complexity	Handles Non-Linearity	Suitability for Cloud
K-Means	Low	No	Moderate
Hierarchical	High	Partial	Moderate
Spectral	Medium	Yes	High

Implementation and Results

We simulated a cloud-native application comprising 10 services with fluctuating workloads. The system was evaluated on its ability to reduce response time and improve fault recovery.

Response Time Improvement

The graph below illustrates the improvement in response time after deploying the adaptation mechanism.

Resource Utilization

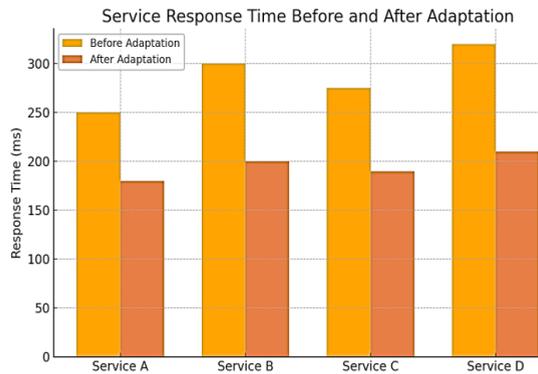


Table 2: CPU and Memory Utilization Before and After Adaptation

Service	CPU Before(%)	CPU After(%)	Memory Before (MB)	Memory After (MB)
Service A	80	55	450	300
Service B	75	50	400	280
Service C	85	60	470	320
Service D	90	65	500	350

Discussion

The results show that spectral clustering enables more intelligent grouping of services based on workload patterns, leading to faster local decisions and more efficient adaptation. Decentralization ensures resilience in case of node or cluster failure, as each cluster operates independently.

Conclusion

This paper introduces a novel decentralized self-adaptation framework leveraging spectral clustering for cloud-native, service-oriented applications. The results demonstrate significant improvements in response time and resource utilization, validating

the effectiveness of the proposed approach.

Future Work

Further work includes integrating reinforcement learning for proactive adaptation and testing the framework under multi-cloud or hybrid cloud scenarios.

Acknowledgment

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

Conflict of interest

The author declare that we have no conflict of interest.

REFERENCES

- Xu, J., Zhao, M., Fortes, J., Carpenter, R., & Yousif, M., Autonomic resource management in virtualized data centers using fuzzy logic-based approaches., *Cluster Computing*, 11(3), 213–227, 2007.
- Dean, J., & Ghemawat, S. Map Reduce: simplified data processing on large clusters., *Communications of the ACM*, 51(1), 107–113, 2008.
- Ng, A. Y., Jordan, M. I., & Weiss, Y. On

- spectral clustering: Analysis and an algorithm. *Advances in Neural Information Processing Systems*, 14, 2002.
4. Chen, X., Liu, H., Zhang, J., & Lyu, M. R. A model-based self-adaptive approach to achieving performance SLAs in cloud environments. *SEAMS*, 85–94, 2014.
 5. Calinescu, R., Dynamic QoS management and optimization in service-based systems., *IEEE Transactions on Software Engineering.*, 37(3), 387–409, 2011.
 6. Salehie, M., & Tahvildari, L. *Self-adaptive software: Landscape and research challenges.* ACM TAAS., 4(2), 1–42, 2009.
 7. Mao, M., & Humphrey, M. A performance study on the VM startup time in the cloud., *IEEE CLOUD*, 423–430, 2011.
 8. Dustdar, S., & Schreiner, W. A survey on web services composition., *International Journal of Web and Grid Services*, 1(1), 1–30, 2005.
 9. Alshahrani, A., & Harous, S., Self-adaptive microservices architecture for managing scalability issues., *Journal of Cloud Computing.*, 8(1), 1–11, 2019.
 10. Li, X., & Venugopal, S. Adaptive resource provisioning for preemptable jobs in cloud systems., *CloudCom.*, 105–112, 2011.
 11. Kephart, J. O., & Chess, D. M. The vision of autonomic computing., *Computer.*, 36(1), 41–50, 2003.
 12. Liu, H., & Shen, H., Fine-grained resource sharing and scaling in cloud computing., *IEEE TPDS.*, 24(6), 1144–1153, 2011.
 13. Zhang, Q., Cheng, L., & Boutaba, R., Cloud computing: State-of-the-art and research challenges., *Journal of Internet Services and Applications.*, 1(1), 7–18, 2010.
 14. Urgaonkar, B., Agile dynamic provisioning of multi-tier internet applications., *ACM TAAS.*, 3(1), 1–39, 2008.
 15. Fernandez, H., Pierre, G., & Kielmann, T. Autoscaling web applications in heterogeneous cloud infrastructures., *IEEE IC2E*, 195–204, 2014.