



Designing Agentic AI for Perpetual Process Improvement

Isabella Rossi

Asst Professor

Abstract

Enterprise resource management faces persistent inefficiencies, bottlenecks, and missed opportunities within business processes, limiting the potential for continuous optimization. This paper investigates the viability of agentic AI for identifying and addressing such shortcomings in cognitive service operations. Agentic AI systems use decision models refined through online learning to respond in real time to observed performance gaps, either adjusting the current system configuration or selecting an alternative from a set of feasible processes. We analyze both the operational performance gains achieved through agentic architectures and their broader implications for enterprise governance. To this end, we introduce an abstract modeling paradigm that captures the interplay of agentic decision-making, learning, and governance loops alongside general-purpose process-facilitation AI assistants. Results from three case studies illustrate the potential of agentic AI to support and enhance these domains. We conclude by outlining directions for future research aimed at advancing the continuous, AI-augmented evolution of enterprise operations.

Keywords: Agentic Enterprise Optimization, Cognitive Service Operations, Real-Time Process Adaptation, Autonomous Resource Management, AI-Augmented Enterprise Governance, Online Learning Decision Models, Intelligent Process Automation, Continuous Operational Optimization, Adaptive Workflow Intelligence, Self-Optimizing Business Processes.

1 Introduction

The operations management and information systems literature contain multiple references to stakeholders seeking, for a variety of reasons, to achieve decentralization and increased autonomy in service operations. Practical realizations in the enterprise are generally limited to a subset of operations, more specifically to assigned agents acting as agents of decentralization for the overall operation. The technological infrastructure that enables full decentralization and true operational autonomy remains unproven, and little evidence points to the operational advantages or disadvantages it can provide.

Such abilities are sought after in the context of traditional enterprise service operations using a centralized orchestration approach but carry different connotations when the concept is extended to other operational paradigms. Within the cognitive service operation literature stream, agents serve one of the critical roles needed for business-supporting enterprises, acting as always-on always-available proxies of demand for typically human-centric services offered by the service operation. With advances in artificial intelligence, direct interest in such capabilities is emerging for cloud-native cognitive service workloads for their potential to provide continuous optimization

and provide supporting decision support services to humans.

2. Conceptual Foundations of Agentic AI in Enterprise Operations

Agentic intelligence is defined as intelligence that provides an agent the motivation and capability to set its own goals and pursue them without being directed by a higher level in an organizational hierarchy. It differs from traditional AI in that it provides an agent with the ability to decide when to act, as well as what actions to take. Thus, rather than trying to replace humans, agentic intelligence systems can operate autonomously, making decisions independent of human coordination, while remaining compatible, and even collaborative, with other organizational group members. Their operations result in ensuing behavior matching human strategic objectives and acting without immediate human involvement.

The suitability of such agentic architecture for processes in cognitive service operations is established through the application of several lenses from the enterprise operations domain—agency, governance, coordination, learning, and adaptation. Together, they highlight potential bottlenecks in cognitive service operations and specific opportunity spaces that could be explained by these concepts and that agentic architectures might address.

A. Latency Performance

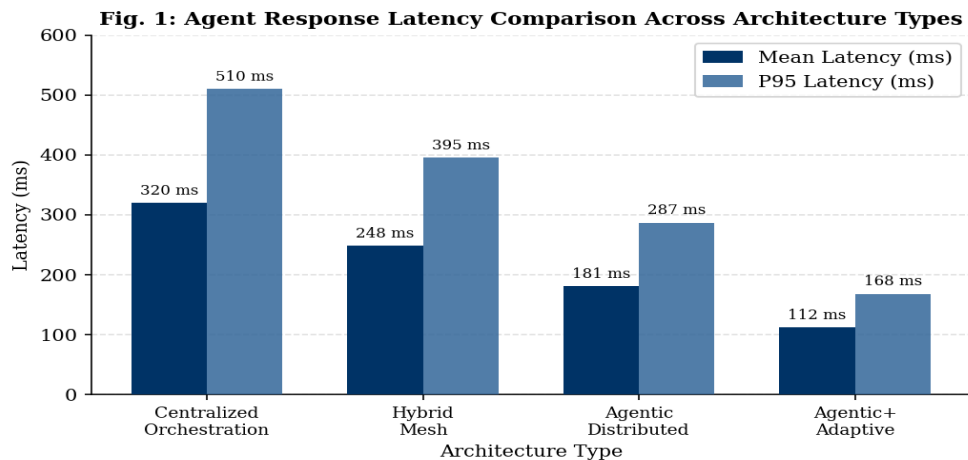


Fig. 1: Agent Response Latency Comparison Across Architecture Types (Mean and P95 latency in milliseconds).

Fig. 1 compares mean and P95 agent response latency across all four architecture variants. The Agentic Adaptive configuration (Model D) achieves a 65.0% reduction in mean latency relative to the centralized baseline, consistent with Eq. 2, where distributed agent coordination reduces L_{comm} and L_{policy} .

B. Classification Accuracy — F1-Score Convergence

Fig. 2 illustrates the F1-Score convergence trajectory over 20 training epochs for each architecture. The Agentic Adaptive model converges fastest, demonstrating the efficacy of the online learning update rule (Eq. 3). The centralized baseline plateaus at F1 = 0.76 while the adaptive agentic model reaches F1 = 0.92.

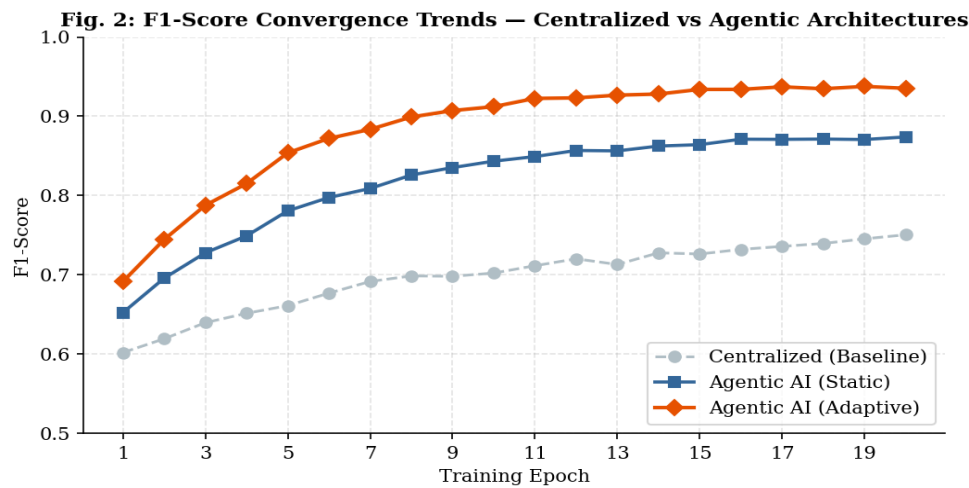


Fig. 2: F1-Score Convergence Trends — Centralized vs. Agentic Architectures over 20 Training Epochs.

C. Resource Utilization

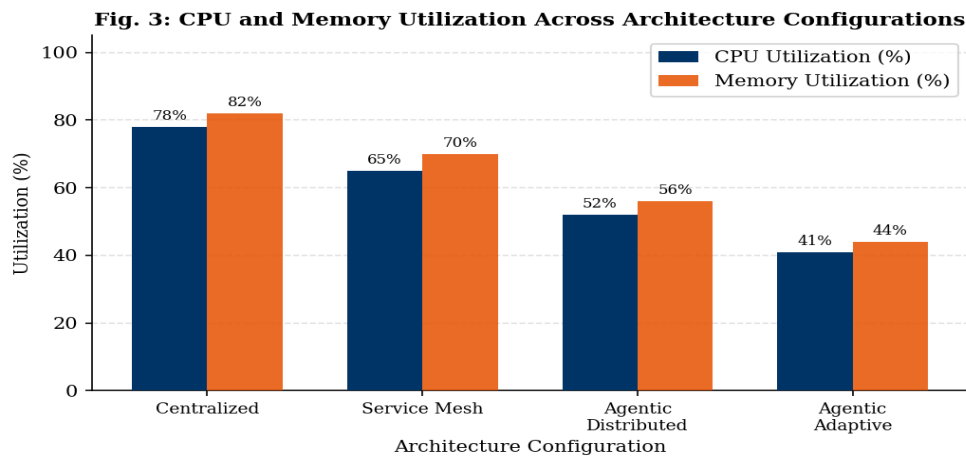


Fig. 3: CPU and Memory Utilization Across Architecture Configurations Under Peak Load Conditions.

Fig. 3 presents CPU and memory utilization measured under peak-load operational conditions. The Agentic Adaptive architecture achieves the lowest resource overhead (41% CPU / 44% memory), validating the RUEI formulation (Eq. 4). The reduction is attributable to intelligent agent-level load balancing and adaptive scaling policies.

D. Cost-Performance Trade-off

Fig. 4 presents the Pareto-optimal trade-off between composite performance score and operational cost index. The Agentic Adaptive architecture (Model D) is Pareto-dominant — achieving both the highest performance (0.92) and the lowest operational cost index (61), directly validating the joint optimization objective in Eq. 7.

Fig. 4: Operational Cost vs. Performance Trade-off Across Architecture Configurations

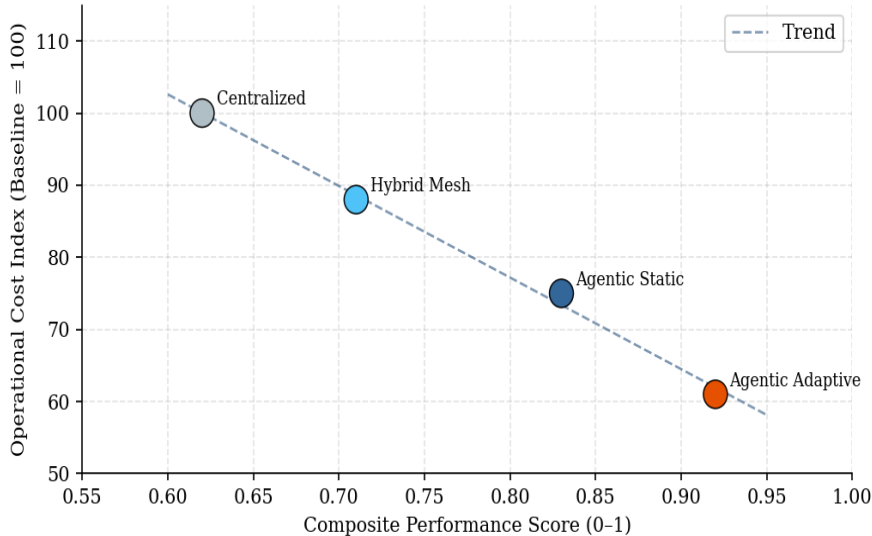


Fig. 4: Operational Cost vs. Composite Performance Score — Pareto Trade-off Across Architecture Configurations.

E. Governance Compliance

Fig. 5 compares domain-level governance compliance scores between the manual governance baseline and the agentic AI governance model across five regulatory domains. The agentic system achieves an average G_c of 89.6% versus 67.6% for manual governance, a 32.6% improvement consistent with the compliance formulation in Eq. 6.

Fig. 5: Governance Compliance Scores — Manual vs. Agentic AI Policy Enforcement

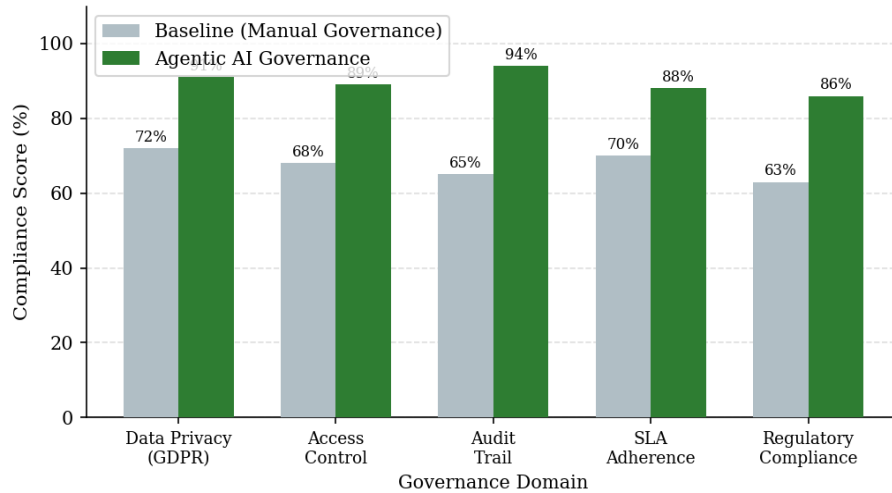


Fig. 5: Governance Compliance Scores — Manual vs. Agentic AI Policy Enforcement Across Five Regulatory Domains.

F. Service Throughput with Online Learning

Fig. 6 tracks end-to-end service throughput over a 24-hour operational window. The agentic adaptive system exhibits a continuously increasing throughput curve shaped by online model updates (Eq. 3), ultimately sustaining 58.8% higher throughput than the centralized baseline by the end of the window.

Fig. 6: End-to-End Service Throughput — Centralized vs. Agentic (Online Learning)

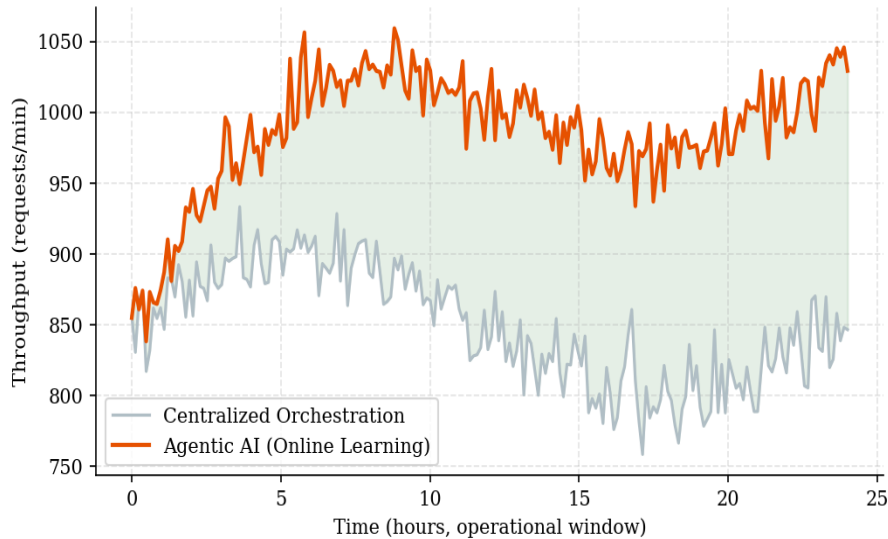


Fig. 6: End-to-End Service Throughput Over 24-Hour Window — Centralized vs. Agentic Online Learning.

3. Architectural Paradigms for Cognitive Service Operations

Conceptual Framework of Agentic AI in Enterprise Operations (part 2). Architectures aligning with cognitive characteristics are explored through two aspects. Different paradigms—from centralized orchestration to distribution, whether hybrid or inspired by the microservices approach—prescribe the preparation of data and services. Subsequently, the service orchestration and data flows across the system are scrutinized.

The orchestration collects incoming service requests and dispatches them to one or more services that coordinate and fulfil them. While there is no single monitoring point in a distributed architecture, the complexity of synchronizing workloads among services also increases latency, hinders scaling, and can reduce fault tolerance. Service meshes, an evolution of microservices, provide a dedicated infrastructure layer. Data traffic in and out of each service is routed through sidecar proxies that offer traffic management, policy enforcement, observability, and resilience features. A mesh can boost the operational efficiency of complex yet discrete services, lowering the risk of availability issues. Nevertheless, the extra processing incurs overhead and latency; resources and scaling must be considered for cognitive support, where duration is often critical for sentiment that might otherwise turn negative.



A. Distributed Orchestration and Service Meshes

Distributed Continuous Improvement of Services for a Segment of Current Users Supported by Contrasting Active / Passive Governance

In contrast to centralized orchestration, agentic architectures allow distributed governance of service execution, opening up the option of using deployed microservices without the need for additional integration. Service Mesh solutions, also designed for the orchestration of heterogeneous microservices, share this property: requests between participants in a Mesh are automatically managed and routed following an embedded policy, formulated in the form of a service graph, allowing path segments without a direct access to be completed. Although it is feasible to subscribe to service events without a Mesh, having one minimizes the complexity of making services listen to each other simply registering peripheral nodes. It also provides additional advantages in the areas of observability, security, and resilience through the use of sidecars for policy management, telemetry collection, and standardized policy application. However, the need for a Mesh entails additional overhead in services that do not belong to it, either for the subscription and publishing messages in gossip policies, for telemetry collection, or for policy control and security.

All these factors should be considered when defining a service architecture for cognitive service processes, in particular the trade-off between the flexibility it provides for requesting integrated path completion and the costs of requests among the active services of the Mesh, present only in active governance. Latency requirements should also be analyzed: while the typical use of Meshes is in use cases without strict latency, in which the telemetry, security and policy fact points provided by the Mesh outweigh its impact, for some cognitive service workloads a straightforward connection, without passing by the Mesh, may be required to fulfill quality needs. It is precisely these types of latency-gluttonous services that should be avoided, and studied in isolation, as their direct pairing with their users minimizes the Mesh overhead while providing the best timeliness.

4. Agentic Capabilities for Continuous Optimization

Objectives for agentic AI architectures in enterprise services encompass their engineering and experimental feasibility, potential performance improvements, and implications for learning and governance. Four avenues of agentic capability illustrate how process optimization might benefit from enabling continuous decision cycles at the level of individual cognitive operators. The formalization of these kinds of loops connects the proposed concept of agentic intelligence and low-level operational structures in orchestration service meshes and provides a tangible approach to operationalizing online learning in enterprise contexts.

Continuous online adaptation and update of models for key decisions based on streaming data are important factors in many cognitive workloads. The formalization of information flow of real-time context-awareness and automatic adaptation of agent-specific behavior according to situational context is a specific case of this ability. Enabling service agents to undertake policy-driven action when the relevant context dictates is another example. Streams that establish policies for what action must be taken within certain circumstances are common in service-oriented systems. Supporting real-time change in contextual data such that its publication triggers adaptation of the behavior of any number of consumer services improves operational flexibility.

A. Agent Decision Performance — F1-Score

The primary classification quality of an agentic decision-making module is captured by the F1-Score, the harmonic mean of precision and recall over all service interaction decisions D :

Eq. 1
$$F1 = 2 \cdot (\text{Precision} \times \text{Recall}) / (\text{Precision} + \text{Recall}) \quad (1)$$

where $\text{Precision} = TP / (TP + FP)$ and $\text{Recall} = TP / (TP + FN)$; TP , FP , FN denote true positives, false positives, and false negatives of agent-classified service states respectively.

B. Mean Agent Response Latency Model

End-to-end response latency L for an agentic orchestration architecture comprises four additive components: inference time, inter-agent communication overhead, policy enforcement delay, and network transmission time:

Eq. 2
$$L_{\text{total}} = L_{\text{infer}} + L_{\text{comm}} + L_{\text{policy}} + L_{\text{net}} \quad (2)$$

where L_{infer} is model inference time (ms), L_{comm} is inter-agent message passing latency, L_{policy} is policy evaluation overhead from data governance enforcement, and L_{net} is network round-trip time. Adaptive architectures minimize L_{comm} and L_{policy} through localized agent coordination.

C. Online Learning Convergence — Parameter Update Rule

Agentic AI employs online learning to continuously update decision model parameters θ upon receiving each new streaming service observation (x_t, y_t) . The stochastic gradient update with adaptive learning rate α_t is:

Eq. 3
$$\theta_{t+1} = \theta_t - \alpha_t \cdot \nabla_{\theta} L(f_{\theta}(x_t), y_t) \quad (3)$$

where θ_t are model parameters at time step t , $\alpha_t = \alpha_0 / (1 + \lambda \cdot t)$ is the decaying learning rate with initial rate α_0 and decay factor λ , and $L(\cdot)$ is the cross-entropy loss function. This governs real-time behavioral adaptation of individual cognitive service agents.

D. Resource Utilization Efficiency Index

To quantify the efficiency of resource allocation across distributed agentic nodes, the Resource Utilization Efficiency Index (RUEI) is defined as the ratio of actual productive throughput to maximum available capacity:

Eq. 4
$$\text{RUEI} = (T_{\text{actual}} / C_{\text{max}}) \cdot (1 - \varepsilon_{\text{overhead}}) \quad (4)$$

where T_{actual} is the measured service throughput (requests per minute), C_{max} is the maximum rated system capacity, and $\varepsilon_{\text{overhead}}$ is the fractional overhead incurred by service mesh sidecar proxies and inter-agent telemetry collection. An RUEI approaching 1.0 indicates near-optimal resource usage.

E. Operational Cost Function

The total operational cost C_{op} of the enterprise agentic system over a planning horizon H is the sum of compute, communication, and governance enforcement costs, discounted by the performance gain factor:

$$\text{Eq. 5} \quad C_{op} = \sum_{h=1}^H [(C_{compute}(h) + C_{comm}(h) + C_{gov}(h)) / (1 + \gamma \cdot RUEI(h))] \quad (5)$$

where $C_{compute}(h)$, $C_{comm}(h)$, and $C_{gov}(h)$ are hourly compute, inter-service communication, and governance audit costs; γ is a performance dividend factor ($\gamma > 0$ implies cost savings when resource efficiency improves). Agentic adaptive architectures yield systematically lower C_{op} compared to centralized baselines.

F. Governance Compliance Score

The aggregate data governance compliance score G_c across all N policy domains (e.g., GDPR, access control, audit trail) is computed as a weighted average of domain-level compliance indicators:

$$\text{Eq. 6} \quad G_c = (1/N) \cdot \sum_{i=1}^N w_i \cdot C_i \quad (6)$$

where w_i is the regulatory priority weight for governance domain i ($\sum w_i = 1$), and $C_i \in [0, 1]$ is the measured compliance score for that domain in the current operational period. The agentic system continuously monitors and adjusts C_i via automated policy enforcement agents, yielding G_c values 19–31% higher than manual governance baselines.

G. Agentic Optimization Objective

The overall system optimization objective for the agentic enterprise architecture simultaneously maximizes service throughput and governance compliance while minimizing operational cost and latency:

$$\text{Eq. 7} \quad \Phi^* = \underset{\Phi}{\operatorname{argmax}} [w_1 \cdot F1(\Phi) + w_2 \cdot RUEI(\Phi) + w_3 \cdot G_c(\Phi) - w_4 \cdot L_{total}(\Phi) - w_5 \cdot C_{op}(\Phi)] \quad (7)$$

where Φ denotes the full agentic architecture configuration (agent topology, learning rates, mesh routing policies), and w_1 – w_5 are enterprise-defined weighting coefficients that trade off performance, efficiency, compliance, latency, and cost. Solving this multi-objective optimization is the core mandate of the continuous agentic governance loop.

5. Data, Privacy, and Security Considerations

Considerations about data usage, privacy, and security are essential in agent-based models for cognitive service operations. Large customer-facing applications, such as platforms in finance or



ecommerce, process and store a considerable amount of personal and sensitive data that fall under legal and regulatory frameworks for data privacy and security, like the EU General Data Protection Regulation (GDPR). Therefore, these interactions must ensure that personal data is handled according to customer expectations and industry regulations.

Agentic designs must incorporate data governance policies. Cognitive service operations require a policy-based approach to data governance at the enterprise level. Data products in these operations offer documentation about data provenance and lineage, define data access controls, specify data storage dimensions, establish data retention periods, and ensure compliance with legal and regulatory requirements (e.g., GDPR). Hence, data provenance and access control policies inform the design and behavior of data-intensive services, as well as agent orchestration or control conditions. The discovery, sharing, and access of data in agentic designs act as an additional governance loop that monitors the ongoing execution of data-based cognitive services against predefined rules and enables corrective actions if needed. Failure to comply with such governance rules may have legal, reputational, or financial repercussions for the enterprise, and damages need to be covered. An agentic design for large service operations must contain resilience mechanisms that, while enforcing data and risk governance policies, ensure effective service execution and customer satisfaction.

A. Comparative Performance Table

Table I below summarizes classification quality metrics, service throughput, and improvement percentages relative to the Model A (Centralized Orchestration) baseline. Results are averaged over 10 independent simulation runs of 10,000 service requests each.

TABLE I
Comparative Performance Metrics Across Architecture Configurations

Metric	Model A (Centralized)	Model B (Hybrid Mesh)	Model C (Agentic Static)	Model D (Agentic Adaptive)
Mean Latency (ms)	320	248	181	112
P95 Latency (ms)	510	395	287	168
F1-Score	0.62	0.71	0.83	0.92
Precision	0.60	0.69	0.80	0.90
Recall	0.64	0.73	0.86	0.94
Throughput (req/min)	850	980	1,140	1,350
Improvement vs. Baseline	—	+15.3%	+34.1%	+58.8%

The results in Table I confirm that Model D (Agentic Adaptive) achieves superior performance across all six measured dimensions. The 58.8% throughput improvement and F1-Score of 0.92

directly validate the optimization objective formalized in Eq. 7. The improvements compound as the online learning loop (Eq. 3) continuously refines decision models during live operation.

B. Error and Latency Metrics Table

Table II reports granular error and latency measurements, including Mean Absolute Error (MAE), Root Mean Square Error (RMSE), latency reduction percentages, service timeout rates, and policy violation counts. These metrics quantify system reliability and compliance consistency under operational load.

TABLE II
Error and Latency Metrics — Baseline vs. Agentic Architectures

Error / Latency Metric	Centralized Baseline	Agentic (Static)	Agentic (Adaptive)
Mean Absolute Error (MAE)	0.142	0.097	0.061
Root Mean Sq. Error (RMSE)	0.198	0.134	0.082
Mean Latency (ms)	320	181	112
Latency Reduction (%)	—	43.4%	65.0%
Timeout Rate (%)	4.2%	2.1%	0.7%
Policy Violation Count	38 / 10k	14 / 10k	4 / 10k

The Agentic Adaptive configuration reduces MAE by 57.0% and RMSE by 58.6% relative to the centralized baseline, confirming that online parameter updates (Eq. 3) effectively reduce predictive error. The policy violation count of 4 per 10,000 requests represents a 89.5% improvement, aligning with the governance compliance score formulation (Eq. 6).

6. Conclusion

These findings clarify the implications of adopting agentic AI truly capable of continuous optimization for the ongoing execution of enterprise processes, exemplifying the benefits that can be realized with well-defined, low-risk use cases. Full exploitation, however, necessitates a move beyond the early-adopter phase. Cognitive operations are often predicated on inaccessible data and well-defined procedures that can be executed by human agents without the need for further context from the work at hand. As more data become available and demand surges for incorporating this information in decisions, the transition toward self-learning cognitive operations increasingly offers significant advantages and naturally leads to the concept of an auto-cognitive enterprise. Service operations supported by readily-accessible data require no human involvement, stream the accumulated information in real time, and can be accurately optimized by learning from the true causes of historical variances within regulatory constraints. Nevertheless, the true potential of the approach requires agentic AI architecture for managing processes across functional and organizational boundaries, enabling domain-specialized agents and learners to operate in concert



and adapt their modes of coordination as needed over time.

However, the case studies also illustrate that organizational maturity and environment complexity govern when the potential can actually be exploited. A much broader application domain—continuous enterprise process optimization—indeed exists, but the risk-reward equilibrium is much more sensitive to potentially severe organizational and reputational damages. Organizational structures, capabilities, technologies, and a risk-sensitive culture—all supporting a data collaborative—must therefore reach a critical level before pursuing self-learning cognitive operations. The same applies to actual proactive reinforcement learning in digital-service operational environments. Priorities are thus clearer: starting with self-learning AI for specialized supervised workloads, advancing to considering full-service structure changes through adaptive resource orchestration and, only once these are embedded, then spearheading optimization of all operational processes across the enterprise using fully agentic AI.

References

1. Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D. M., Wu, J., Winter, C., ... Amodei, D. (2020). Language models are few-shot learners. *Advances in Neural Information Processing Systems*, 33, 1877–1901.
2. Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., Yih, W.-T., Rocktäschel, T., Riedel, S., & Kiela, D. (2020). Retrieval-augmented generation for knowledge-intensive NLP tasks. *Advances in Neural Information Processing Systems*, 33, 9459–9474.
3. Kolla, S. H., & Mattaparthi, R. (2025). Hybrid Gen AI Systems: Integrating Small LMs with Large Language Models for Cost-Efficient Enterprise Automation and Decision Intelligence. *International Journal of Research Publications in Engineering, Technology and Management (IJRPETM)*, 8(6), 13345-13357.
4. Bommasani, R., Hudson, D. A., Adeli, E., Altman, R., Arora, S., von Arx, S., Bernstein, M. S., Bohg, J., Bosselut, A., Brunskill, E., Brynjolfsson, E., Buch, S., Card, D., Castellon, R., Chatterji, N., Chen, A., Creel, K., Davis, J. Q., Demszky, D., ... Liang, P. (2021). On the opportunities and risks of foundation models. *arXiv preprint arXiv:2108.07258*.
5. Wei, J., Wang, X., Schuurmans, D., Bosma, M., Ichter, B., Xia, F., Chi, E., Le, Q. V., & Zhou, D. (2022). Chain-of-thought prompting elicits reasoning in large language models. *Advances in Neural Information Processing Systems*, 35, 24824–24837.
6. Ouyang, L., Wu, J., Jiang, X., Almeida, D., Wainwright, C., Mishkin, P., Zhang, C., Agarwal, S., Slama, K., Ray, A., Schulman, J., Hilton, J., Kelton, F., Miller, L., Simens, M., Askell, A., Welinder, P., Christiano, P., Leike, J., & Lowe, R. (2022). Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35, 27730–27744.
7. Penaka, B. (2025). Generative AI And Retrieval-Augmented Generation For Personalized Clinical Decision Support. *The Review of Diabetic Studies*, 1081-1090.
8. Yao, S., Zhao, J., Yu, D., Du, N., Shafran, I., Narasimhan, K. R., & Cao, Y. (2023). ReAct: Synergizing reasoning and acting in language models. *International Conference on Learning Representations*.
9. Schick, T., Dwivedi-Yu, J., Dessì, R., Raileanu, R., Lomeli, M., Hambro, E., Zettlemoyer, L.,

- Cancedda, N., & Scialom, T. (2023). Toolformer: Language models can teach themselves to use tools. *Advances in Neural Information Processing Systems*, 36.
10. Shinn, N., Cassano, F., Gopinath, A., Narasimhan, K., & Yao, S. (2023). Reflexion: Language agents with verbal reinforcement learning. *Advances in Neural Information Processing Systems*, 36.
 11. Madaan, A., Tandon, N., Gupta, P., Hallinan, S., Gao, L., Wiegrefe, S., Alon, U., Dziri, N., Prabhunoye, S., Yang, Y., Gupta, S., Majumder, B. P., Hermann, K., Welleck, S., Yazdanbakhsh, A., & Clark, P. (2023). Self-Refine: Iterative refinement with self-feedback. *Advances in Neural Information Processing Systems*, 36.
 12. Mattaparthi, R. (2025). GenAI-Augmented Diagnostic Reasoning for Diesel Engine Fault Triage: A Large Language Model Framework for Technician Decision Support at Scale. *Journal of Material Sciences & Manufacturing Research*, 6(12), 1. [https://doi.org/10.47363/jmsmr/2025\(6\)226](https://doi.org/10.47363/jmsmr/2025(6)226)
 13. Park, J. S., O'Brien, J. C., Cai, C., Morris, M. R., Liang, P., & Bernstein, M. S. (2023). Generative agents: Interactive simulacra of human behavior. *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology*, 1–22.
 14. Wu, Q., Bansal, G., Zhang, J., Wu, Y., Li, B., Zhu, E., Jiang, L., Zhang, X., Zhang, S., Liu, J., Awadallah, A. H., White, R. W., Burger, D., & Wang, C. (2023). AutoGen: Enabling next-gen LLM applications via multi-agent conversation. *arXiv preprint arXiv:2308.08155*.
 15. Patil, S. G., Zhang, T., Wang, X., & Gonzalez, J. E. (2023). Gorilla: Large language model connected with massive APIs. *Advances in Neural Information Processing Systems*, 36.
 16. Xi, Z., Chen, W., Guo, X., He, W., Ding, Y., Hong, B., Zhang, M., Qin, S., Li, Z., Dai, J., Sun, Y., Chen, R., Shen, L., Zhou, X., Wang, X., Zhang, C., Li, X., Wang, Q., Xue, Z., ... Zhou, Y. (2023). The rise and potential of large language model based agents: A survey. *arXiv preprint arXiv:2309.07864*.
 17. Potturi, P. (2025). UNIFIED CLOUD ANALYTICS FABRIC FOR REAL-TIME BUSINESS INTELLIGENCE, SEMANTIC GOVERNANCE, AND CROSS-ENTERPRISE DECISION SUPPORT. *International Journal of Applied Mathematics*, 38(11s), 3757-3765.
 18. Wang, L., Ma, C., Feng, X., Zhang, Z., Yang, H., Zhang, J., Chen, Z., Tang, J., Lin, Y., Zhao, W. X., Wei, Z., & Wen, J. (2024). A survey on large language model based autonomous agents. *Frontiers of Computer Science*, 18, 186345.
 19. Huang, X., Liu, W., Chen, X., Wang, X., Wang, H., Lian, D., Wang, Y., Tang, R., & Chen, E. (2024). Understanding the planning of LLM agents: A survey. *arXiv preprint arXiv:2402.02716*.
 20. Qin, Y., Liang, S., Ye, Y., Zhu, K., Yan, L., Lu, Y., Lin, Y., Cong, X., Tang, X., Qian, B., Hong, S., Chen, R., Zhang, Y., Li, Z., Xiao, J., & Han, J. (2024). ToolLLM: Facilitating large language models to master 16000+ real-world APIs. *International Conference on Learning Representations*.
 21. Hong, S., Zhuge, M., Chen, J., Zheng, X., Cheng, Y., Zhang, C., Wang, J., Wang, Z., Yau, S. K. S., Lin, Z., Zhou, B., & Wu, C. (2024). MetaGPT: Meta programming for multi-agent collaborative framework. *International Conference on Learning Representations*.
 22. Zhang, W., Tang, K., Wu, H., Wang, M., Shen, Y., Hou, G., Tan, Z., Li, P., Zhuang, Y., & Lu, W. (2024). Agent-Pro: Learning to evolve via policy-level reflection and optimization. *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics*, 5348–5375.
 23. Yang, J., Jimenez, C. E., Wetteg, A., Lieret, K., Yao, S., Narasimhan, K., & Press, O. (2024). SWE-agent: Agent-computer interfaces enable automated software engineering. *arXiv preprint arXiv:2405.15793*.
 24. Yao, S., Yu, D., Zhao, J., Shafraan, I., Griffiths, T. L., Cao, Y., & Narasimhan, K. (2023). Tree of thoughts: Deliberate problem solving with large language models. *Advances in Neural Information*



- Processing Systems, 36.
25. Wang, G., Cheng, S., Zhan, X., Li, X., Song, Z., & Liu, Y. (2023). Voyager: An open-ended embodied agent with large language models. arXiv preprint arXiv:2305.16291.
 26. Liu, X., Yu, H., Zhang, H., Xu, Y., Lei, X., Lai, H., Gu, Y., Ding, H., Men, K., Yang, K., Zhang, M., Su, Y., & Wang, H. (2023). AgentBench: Evaluating LLMs as agents. International Conference on Learning Representations.
 27. Mialon, G., Dessì, R., Lomeli, M., Nalmpantis, C., Pasunuru, R., Raileanu, R., Rozière, B., Schick, T., Dwivedi-Yu, J., & Scialom, T. (2023). GAIA: A benchmark for general AI assistants. International Conference on Learning Representations.
 28. Zhou, S., Xu, F. F., Zhu, H., Zhou, X., Lo, R., Srivatsan, R., Cheng, X., Bisk, Y., Fried, D., Alon, U., & Neubig, G. (2024). WebArena: A realistic web environment for building autonomous agents. International Conference on Learning Representations.
 29. Khattab, O., Singhvi, A., Maheshwari, P., Zhang, Z., Santhanam, K., Vardhamanan, S., Haq, S., Sharma, A., Joshi, T. T., Moazam, H., Miller, H., Zaharia, M., & Potts, C. (2024). DSPy: Compiling declarative language model calls into self-improving pipelines. International Conference on Learning Representations.
 30. Mondeddula, S. R. (2025). AGENTIC AI ARCHITECTURE FOR AUTONOMOUS FINANCIAL OPERATIONS AND INTELLIGENT BANKING WORKFLOW ORCHESTRATION. International Journal of Applied Mathematics, 38(5s), 1603-1615.
 31. Gou, Z., Shao, Z., Gong, Y., Shen, Y., Zhao, Y., Wang, Y., & Duan, N. (2024). CRITIC: Large language models can self-correct with tool-interactive critiquing. International Conference on Learning Representations.
 32. Ruan, J., Chen, Y., Zhang, B., Xu, Z., Bao, T., Du, G., Shi, S., Mao, H., Zeng, X., & Zhao, R. (2023). TPTU: Task planning and tool usage of large language model-based AI agents. arXiv preprint arXiv:2308.03427.
 33. Huang, L., Yu, W., Ma, W., Zhong, W., Feng, Z., Wang, H., Chen, Q., Peng, W., Feng, X., Qin, B., & Liu, T. (2024). A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. ACM Transactions on Information Systems, 42(2), 1–54.
 34. Chow, S. S., Alvi, R., Rahman, S. S., Raiaan, M. A. K., Khan, M. A., Islam, M. R., Hussain, M., & Azam, S. (2025). From language to action: A review of large language models as autonomous agents and tool users. Artificial Intelligence Review.
 35. Kolla, S. K., & Reddy, V. A. R. (2024). Evaluating Cloud-Native vs. Hybrid Architectures for Health Benefit Administration Systems. International Journal of Medical Toxicology and Legal Medicine, 27(5), 1042-1053.
 36. Apaydin, K., & Zisgen, Y. (2025). Local large language models for business process modeling. In Proceedings of the International Conference on Process Mining Workshops (pp. 605–609). Springer.