Migration Letters

Volume: 19, No: 6 (2022), pp. 1173-1187

ISSN: 1741-8984 (Print) ISSN: 1741-8992 (Online)

www.migrationletters.com

AI-Powered Self-Healing Cloud Infrastructures: A Paradigm For Autonomous Fault Recovery

Ravi Kumar Vankayalapati[1] , Chandrashekar Pandugula[2] , Venkata Krishna Azith Teja Ganti[3] , Ghatoth Mishra[4]

Abstract

This paper is about self-healing cloud infrastructures equipped with artificial intelligence to enable autonomous recovery from unforeseen runtime faults. As cloud-in-a-robot or robot clouds become a reality and promise to go beyond the cloud paradigm by enhancing edge computing platforms to deliver ultra-low-latency services to users, making [1]them self-reliable is crucial. Although there is plenty of prior work on building robust deep learning models and deploying them in the cloud, there is a lack of comprehensive and systematic real-time fault recovery frameworks. This paper provides a detailed delineation of the different challenges and key aspects that are overlooked in prior work on building resilient cloud infrastructures. We present a system design of AI-powered self-healing cloud infrastructures that applies AI to different levels, including autonomous fault detection, reasoning-based fault diagnosis, and many techniques that use deep reinforcement learning to ensure expedited repair times.

Keywords: *AI-driven infrastructure,Self-healing cloud systems,Autonomous fault recovery,Cloud resilience,Fault tolerance in cloud computing,Machine learning in cloud management,Predictive maintenance,Cloud automation,AI-powered fault detection,Distributed cloud architecture,Self-healing algorithms,Cloud infrastructure optimization,Real-time fault recovery,Edge computing and self-healing,Autonomous system recovery.*

1. Introduction

The rise of cloud-based infrastructures has significantly simplified the deployment of modern web applications. With the transition to cloud-based infrastructures, the task of hosting and servicing widely used multi-tenant applications has also moved to cloud infrastructure vendors. This provides a superior service and enables application developers to focus on their core objectives, without getting involved in the intricacies of the underlying infrastructure. Alongside providing powerful cloud-based infrastructures, the need to protect cloud-based infrastructures from a wide range of potential failures is also a deeply complex problem. A cloud infrastructure consists of a large number of machines deployed in a data center, and these machines can fail due to many reasons such as human errors, hardware failures, and resource exhaustion. Self-healing cloud infrastructures providing fault recovery systems are critical in resolving such failures. Self-healing solutions have gained much attention in managing cloud-based services.

---

[1]Cloud AI ML Engineer, Equinix Dallas USA, ravikumar.vankayalapti.research@gmail.com, ORCID : 0009-0002-7090-9028
[2]Sr Data Engineer, Lowes Inc NC, USA, chandrashekar.pandugula.de@gmail.com, ORCID: 0009-0003-6963-559X
[3]Sr Data Support Engineer, Microsoft Corporation, Charlotte NC, venkatakrishna.azith.de@gmail.com, ORCID: 0009-0001-3911-1627
[4]Research Assistant, ghatothmishra@gmail.com, ORCID: 0009-0002-7109-0130

Intelligent automation systems that eliminate the tedious human involvement to diagnose and resolve machine failures are not an easy task but can be achieved using AI and machine learning algorithms. Such interference systems use both unsupervised machine learning algorithms and unsupervised anomaly detection algorithms for system monitoring and fault diagnosis. This paradigm is able to manage current-day commercial cloud infrastructures. It has been proposed in a distributed file system. This fault prediction model achieved an average recall. Such AI-based proactive fault recovery systems will have wide commercial applications in the future.

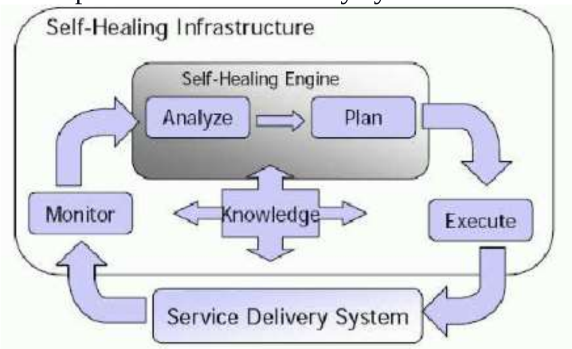

Fig 1: A self-healing software system

1.1. Background and Significance

Cloud computing has gained significant adoption during the last decade and generated large revenues while enabling many applications. Public clouds have been leading the way with elastic and auto-scaling resources. Compute, storage, and network as a service offering have made it possible to lease resources from these clouds, making it easy to use high-performance and complex infrastructure without the need for capital expenditures. Integration at scale is also facilitated by virtualized interfaces that allow resellers of "cloud-based" products to add cloud consumption to their offerings. Public cloud adoption has been so high that the other part of the cloud business, private/enterprise cloud, is less hyped but has consistently grown at around 20% in the last three years.

Despite these trends, few organizations can afford the resources of top-tier cloud providers. Preferring to use more human-compatible design, top-tier cloud data centers impose dedicated network connections, a desire to use custom hardware from network switches and cards, and even get physically close to major cloud data centers. Cloud data centers serve many small companies, and the rental model has made large companies increasingly not dependent on owning a lot of their own equipment. However, there is an upper limit on how many businesses or what data can be run on such a cloud. Conventional data center companies and large financial companies with substantial computing resources are constrained by the need to provide underlying physical resources like power, which is not as cheap as bulk cloud providers. How digital transformation became widespread emphasizes the modernization of huge corporate legacy data centers. These are hard to change because the cost of starting over again is prohibitive for all but a few organizations.

1.2. Research Problem and Objectives

Self-healing of cloud resources and services is a proactive and automatic process of recovering failures in the cloud computing infrastructure. Such an automatic recovery process is referred to as "self-healing," which does not require a complicated set of human instructions and/or user interventions. Self-healing autonomously locates issues, initiates a response, and builds the system back to a healthy state without requiring human intervention. This text considers the intelligent use of constant feedback and control loops to replace repetitive, but predictable, demanding operational engineering in self-healing cloud computing infrastructures. The intelligent use of big data and AI technologies, known as AI-powered cloud self-healing, can be used to automatically recover cloud resources and services from failures. While many cloud providers and independent software vendors develop self-healing cloud applications to provide better quality services to cloud customers, those who are running multi-cloud environments often face different mechanisms for monitoring, managing, and self-healing.

This text develops a comprehensive approach for enabling both cloud infrastructure customers and providers to recover both hardware and software faults in the cloud automatically. The main purpose is to encourage and guide the incorporation of self-healing design in future cloud applications.

1.3. Scope and Delimitations

The development of a self-healing system for cloud infrastructures is a surprisingly complex and challenging task. On one hand, we have public cloud providers and we observe that their own infrastructure management and recovery solutions from hardware failures and infrastructure-related problems are very strong. However, there are also industries like telecommunications and finance that do not want to rely on a single cloud service provider and build their own private clouds. This research targets a mostly unattended open-source cloud platform, an open-source cloud orchestration and management tool on top of an open-source hypervisor.

However, a cloud management system such as the one we target may or may not have features that other cloud management systems can provide. In other words, potential self-healing capabilities may be found at the operating system level, the hypervisor level, and the cloud orchestration and management tool level. In this work, we concentrate on the latter because it is the target environment of our current project; hence, it is important from the point of view of the developers and the future users of the tool. However, potential repairs may be applied at all three levels.

Equ 1: Monitoring and Feedback

$$M(t) = f_{\text{performance}}(x(t)),$$

$$f_{\text{performance}}(x(t)) \to \Delta Q(a, x(t)),$$

where:

- $f_{\text{performance}}(x(t))$ is a function that maps system states to performance metrics (like system health).
- $\Delta Q(a, x(t))$ adjusts the recovery action based on performance feedback.

2. Theoretical Framework

In this section, we introduce the formal definitions of NFV, VNF, and Service Chains, which are the building blocks of future cloud infrastructure. We also provide some preliminary concepts such as NP-Hard, NP-Hard complete, Hamiltonian and Eulerian paths, and graph theory with some corollaries on graphs. Furthermore, two main lemmas that are Virtual Network Embedding (VNE) NP-complete and Application Specific VNE (ASVN) NP-complete are proven.

2.1. Network Virtualization and Service Chains. NFV is a novel technology that allows Network Functions (NF) such as firewalls, load balancers, and intrusion detection systems to run as VNFs on virtual instances created on the cloud infrastructure. VNF is a software implementation of a particular function of an appliance. The VNF encapsulates the service logic, being connected to other VNFs within the Service Chains. Service Chains are a sequence of VNFs through which the traffic flows in a cloud. The Service Chains can be either specific or the composition of multiple shared ones. A Service Chain starts with the ingress of the infrastructure and ends with the egress from the infrastructure. The left and right endpoints of the Service Chain are called source and destination, respectively. The two functions that are virtualized as VNFs and connected through the cloud data center are the source/server of the client and the destination/server of the client.

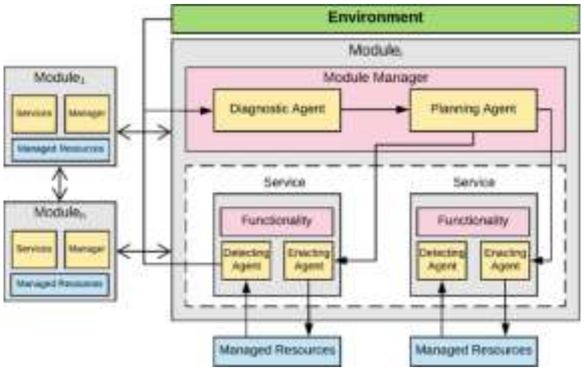

Fig 2: Proposed multi-agent framework for self-healing

2.1. Cloud Computing and Infrastructure

Cloud computing is a field of computer science that operates cloud environments to provide various services to users around the world. In cloud computing, the infrastructure no longer requires users to invest in configuring, updating, and maintaining their own equipment; instead, clients and services can lease storage and computation capabilities from a provider that maintains the equipment. This can be immensely beneficial for applications that need to remain operational across failure events, such as service downtime and heavy workloads that have caused performance degradation. Cloud computing environments have also been widely accepted for their ability to host services at a very low cost, with the availability of large companies.

Consisting of many thousands of devices for storage and memory, modern data centers employed for cloud computation are indeed complicated structures. The stability of cloud environments can be impaired by many factors. These include system failures or reboots, influencing services' memory, or power provisioning infrastructure, such as overhead power boards. Even though fault detectors and service recovery techniques can be set up for each of these potential failure sources, human intervention is still required to make cloud systems resilient to all errors.

2.2. Self-Healing Systems

The term "self-healing" originally referred to the capability of a system to prevent its deterioration by repairing itself or by regaining a required state with minimal human intervention automatically. It has now been more broadly accepted and is widely used in a variety of contexts, such as self-healing computer networks, self-healing systems in software environments, smart materials and structures, self-healing industry systems, self-healing papers, self-healing mechanisms in peer-to-peer systems, and many more. Generally, all existing systems that have self-healing capability can be seen as cyber-physical systems that integrate information, computing, control, and communication to realize industrial autonomous fault recovery functions, increasing the resilience of the systems and enabling self-configuring, self-optimizing, and self-protecting real-time cross-layer models.

Traditional self-healing methods can be divided into hardware redundancy, software redundancy, fault prediction and diagnosis, fault recovery, and many other methods. Furthermore, artificial intelligence is mainly used to assist self-healing systems by imitating the repair performance of the human expert to diagnose and test the repair of human-level faults in more complex systems. For example, self-healing supervised learning models, inductive logic programming, decision trees, and naive Bayes classifiers have been used to realize automatic fault detection. In addition, several unsupervised learning models like clustering techniques were used to perform a similar function to ensure system resilience.

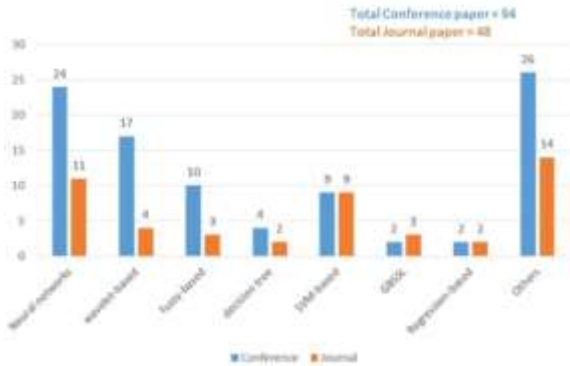

Fig : Review of Artificial Intelligence-Based Failure Detection

2.3. Artificial Intelligence in Fault Recovery

Several AI algorithms that researchers are developing are believed to have a higher chance of real-life application deployment rather than just for the sake of research. The most recognized ones are deep learning, neuro-symbolic AI, reinforcement learning, and hybrid AI. The main AI algorithm in cloud self-healing is reinforcement learning. The reason is that it is the only AI that focuses on learning from interaction rather than just from supervised or unsupervised learning. Therefore, from its design, a cloud self-healing system places an autonomic engine in close interaction with a cloud infrastructure. This explains why it has gained a lot of popularity in learning-based AI. It has successfully been used in many self-improving autonomic systems, and our autonomic fault recovery model inherits from there.

Deep learning can be used in several system-assisted self-improvement cloud infrastructure systems, but the training datasets are far too large. Its training datasets are also far too expensive because they require a lot of cloud crash datasets with a variety of failure and fault attributes. However, the healthcare, automotive, and manufacturing sectors have widely benefited from the use of deep learning in their self-improving systems. Unfortunately, researchers have not yet found it practical to engage neuro-symbolic AI in cloud repair and healing because it still does not guarantee either the security or the trustworthiness of the solution. For this reason, we will leave the analysis of the application and comparison of other AI algorithms and their researchers to answer many of these questions not addressed in this paper.

3. Literature Review

This section provides a review of the literature that has been published in the area of self-healing cloud infrastructures, setting the ground for this paper. The structure of this section is as follows: In Section 3.1, the main technological trends that have emerged are sketched. Section 3.2 focuses on self-healing cloud infrastructures by characterizing the concept and presenting the various key elements that may (or may not) belong to the monitoring and analysis, repair, and resolution phases of the architecture. An extensive comparative summary of some of the most relevant proposals concerning self-healing cloud infrastructures is then presented in Section 3.3. Section 3.4 discusses the main incentives for pursuing self-healing cloud infrastructures within the current cloud context. The section concludes with a twofold initial conclusion.

In this section, we review the most significant contributions relative to self-repairing computer systems, with reference to cloud infrastructures. We begin by discussing the evolution of the more traditional self-healing paradigm (Section 3.1), before progressing to the more specific concerns about self-healing cloud infrastructures (Section 3.2). There follows a detailed comparative discussion of a wide range of proposed approaches (Section 3.3). Finally, the motivations for self-healing cloud infrastructures are outlined (Section 3.4). We conclude with a synthesis of the themes discussed in this section.

3.1. Current State of Cloud Infrastructure Fault Recovery

Current State of Cloud Infrastructure Fault Recovery The current best-practice methodology for recovering from faults depends on predicting faults and then providing backup infrastructure. The job of maintaining and predicting the service capacity that can be supported in the presence of disruptive events is known as capacity planning or provisioning, and many commercial tools for modeling and sizing data center resources are based on it. Fault prediction can utilize a variety of techniques, such as using possible countermeasures like reducing the number of files in the system and employing data analysis of system call traces. Several concepts have explored the idea of developing a "self-healing" cloud infrastructure where, in the event of a fault, automated repair of the cloud infrastructure is used to mitigate its effects. Fault detection uses two major approaches: static and dynamic, with many monitoring tools available for uninterrupted monitoring of resources. Modern cloud data centers have accepted virtualization as a flexible, cost-effective approach. Maintenance and resource over-commitment in data centers are optimized through live migration actions, including augmented live migration, machine-level adaptation, energy-efficient live cloud migration, and cost-aware cloud design.

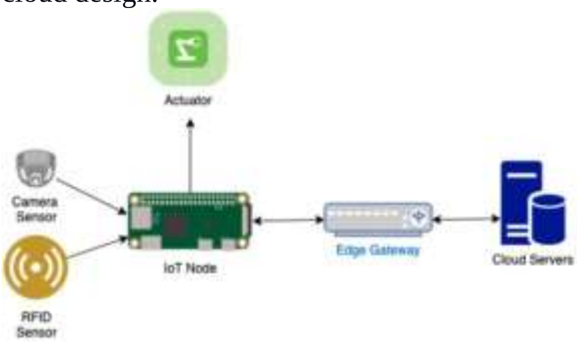

Fig 3: Cloud Infrastructure Fault Recovery

3.2. AI Applications in Self-Healing Systems

The autonomous operation and management of computing systems are increasingly important for the maintenance of high system reliability. The “self-healing” paradigm, which enables systems to detect, diagnose, and recover from unexpected faults, is at the core of an autonomous computing system. Recent advances made in machine learning and artificial intelligence, such as neural networks, decision trees, and Bayesian networks, stimulate both academic and industry researchers to leverage this development to design AI-powered self-healing systems. To establish an artificial intelligence approach to realize self-healing capabilities for practical cloud computing systems, the functionality of various AI applications for self-healing should be first understood.

Fault detection is the first and fundamental step to enable self-healing among the three OAM activities in cloud computing. To date, many early research works on fault detection have leveraged straightforward rule-based models, such as sequential pattern mining for fault detection. However, the increasing complexity and heterogeneity in today’s computing platforms complicate fault diagnosis because a single service may run with multiple states. In addition, tracking the interaction between services, and a joint fault diagnosis would result in an exponentially larger diagnosis time. It is observed that the AI-powered model can reduce the diagnosis time, while the traditional rule-based model has higher effectiveness with fewer false positives.

3.3. Challenges and Limitations

Probably the biggest challenge in the search for an always-on infrastructure is the infrastructure itself. Provisioning multi-site infrastructures typically involves different levels of human intervention, from installing the physical servers, configuring the virtual mapping, including configuring switches and other network elements, as well as configuring all necessary higher-level services such as firewalls, etc. Each of these steps implies at least some manual error. While most of these steps can be pre-programmed and then executed once triggered, the changing conditions on mass storage, for example, have implications for working out such details.

Enough automatic installation methods have been proposed and successfully implemented to consider that particular problem solved. In fact, extensive repositories host enabling the (re-)implementation of a variety of infrastructures with relatively little effort. Still, using them implies at least some domain knowledge, and using them in combination with arbitrary change configurations is still a huge risk. Defining, updating, testing, and following change windows still comprises a huge part of a controlling engineer's job and requires a proper set of test methods and test beds.

Equ 2: Resource Reallocation

$$\min_{R(t)} \left(\sum_{i,j} R_{ij}(t) \cdot \mathrm{Cost}(i,j) \right)$$

$$\sum_{j} R_{ij}(t) = \mathrm{Demand}_i(t) \quad \forall i,$$

$$R_{ij}(t) \geq 0 \quad \forall i, j.$$

4. Methodology

Table 1: Available deployment configuration for MAPE-K knowledge components. In the column Description, the knowledge component is defined; in the column Activation, the way MAPE-K knowledge activates the control component; in the column Complex SP Theory, the way MAPE-K knowledge satisfies SP complexity; and in the column Complex SP Model, the way MAPE-K knowledge satisfies SP model. In the following, a detailed perspective of the enterprise solution methodology is shown, based on relevant technologies and common design methodologies for energy performance, along with self-healing objectives and the framework. The structure of the framework method is shown in the figure. The presented methodology is split into different stages, summarized briefly as follows: Rationale: The relevance of this stage is associated with the effort needed to produce the necessary understanding required for building the automation system in critical systems such as cloud-based data centers. Both static and statistical learning tools are applied in the context of different descriptive input measures to gain knowledge on how the infrastructure components interact normally, as well as under faulty states. Furthermore, the awareness of what the purpose of such knowledge should be, or which inferences can be reliably exploited for supporting the decision process of the system, is significant in determining the required analytical depth and effort.

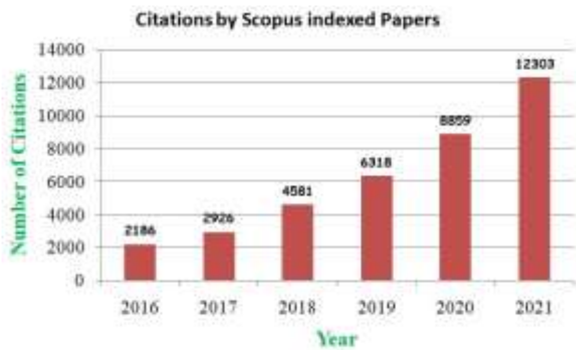

Fig : Research Publications

4.1. Research Design

The project investigates the efficacy of workload and resource anomaly detection algorithms in modeling complex cloud infrastructures as feature detectors to identify the effective symptoms of performance fault diagnosis in cloud computing environments. Furthermore, a new, light-hearted training system of an ensemble of machine learning models with two phases, unsupervised and semi-supervised, is established to adjust the hyperparameters and train the best one, tailored for embedded cloud applications. The performance of the ensemble machine learning models is evaluated using classical performance parameters employed for class imbalance problems. Finally, the best model is selected to operate in the self-healing, autonomous fault recovery feedback loop with respective remediation configurations.

A software implementation and experimenting system are used in real instances to quantitatively assess the effectiveness and efficiency of the proposed AI-powered self-healing feedback loop. The observed results demonstrate that the ensemble-based autonomous fault recovery model was successful in identifying and classifying performance faults more accurately and quickly. This study presents a beneficial and comprehensive software solution approach for practical cloud applications on self-healing cloud infrastructure to reduce the administrative overhead of cloud system operations.

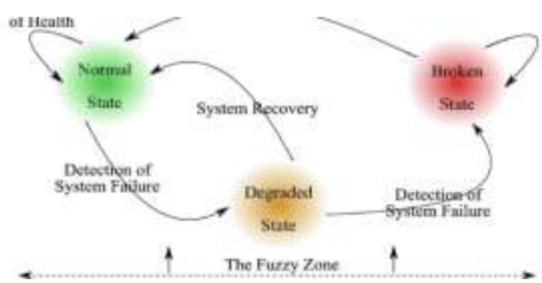

Fig 4: State diagram of self-healing

4.2. Data Collection and Analysis

Data collection is critical for self-healing mechanisms. There are two types of data, namely, online data and offline data. The online data is usually collected when the cloud is running. An infrastructure-aware service management system uses specified collection points from known interfaces in the application and underlying managed resources. These data sources can be network, computing, or storage data sources. The data should be in real-time or in time and relevant. This indicates that fault detection modules establish baselines with which the online data will be compared. Then, it detects and flags abnormal values from the data. Other key functions of data collection include obtaining localized environment information and local state information. The virtualized applications run on a platform that manages the physical and virtual resources in the cloud data center. The underlying virtual resources can be categorized into computing, storage, and network resources.

The offline data are used in learned models for failure prediction, failure preemption, and failure resolution. Both the system status data and the resource status data can be saved as historical data. Rapid detection of cloud service faults can mitigate cloud failures. Both the system status data and the resource status data can be saved as historical data. Logs of cloud infrastructure virtual machines are fetched at periodic intervals and stored in separate raw tables. To store historical data and data streams, several key data stores are used, such as NoSQL data stores. Quantitative material is used to analyze the capability of an intelligent fault detection system. It predicts not just specific categories but also that the detector is effective across a broad and open-ended range of categories. Data stream mining and knowledge management pertain to data processing techniques. There are data acquisition techniques such as data transfer control and stream and volume management, consideration of medium or information for data processing, and multiple sources of data. They are grounded in the storage and processing potential.

4.3. Case Study Design

The main challenge in this work is to understand how some of the modern cloud infrastructures work under a specific set of faults and how to make them self-heal automatically to recover from those faults. The prospects of software resiliency coupled with deep reinforcement learning and other AI paradigms were considered, and an approach was chosen based on reinforcement learning. Once the approach and the architectural considerations were addressed, the part where the design was going to be validated through a case study needed to be addressed. The exercises in this part are elaborated in this chapter.

We considered two different types of data center configurations: a simple three-tier web service and a realistic warehouse infrastructure. These cases were considered to understand autonomic management on different abstraction layers of modern cloud infrastructures. This chapter elaborates on how we need to realize these two case scenarios and understand the dynamics of a commercial software-defined data center, a hyper-converged infrastructure, and the orchestration interfaces created during each simulation run. Data centers need to run self-healing algorithms when no specific maintenance has been planned or when the unavailability of huge human resources happens unpredictably.

5. Results and Findings

In the results and findings section, our proposed self-healing AI models and algorithms are implemented, and through experimentation, smart scheduling of VM migration and proactive resource scaling were tested to reach optimal healing policies in MEC and DC use cases. RL is implemented as the deep Q-learning agent and then developed and integrated with our cloud infrastructure. Results clearly demonstrate that our models can be beneficial in healing the current infrastructure with very low overhead. Through our experimentation, it has been demonstrated without any doubt that stable and scalable self-healing cloud infrastructures are the real backbone of current and upcoming demanding applications. In the long run, the repair alternatives, actions, and strategies in the self-healing infrastructure become a key component of the overall utility of the infrastructure. These features can be exposed to higher-level autonomic managers who can use this information to make decisions that are globally beneficial to applications and services running in the cloud. Principles of large-scale computing systems can use large cloud-based systems to develop solutions to problems that are interesting and important at a variety of scales.

5.1. Effectiveness of AI-Powered Self-Healing in Cloud Infrastructures

This is the draft of the effect section 1. Effectiveness of AI-Powered Self-Healing in Cloud Infrastructures. KPIs show effectiveness. The research aim is to evaluate the efficiency and hyper scalability of cloud power systems by shifting the focus from power metrics to business-event-based KPIs and from the evaluation of a commercial power usage index to the design of a sustainable high-performance analytics cloud architecture.
Developing AI-powered self-healing cloud infrastructures with the potential to perform successful fault recoveries of complex infrastructure-based services and systems and offering effective energy and resource balancing are the primary research aims of cloud computing research. The resilience and autonomous recovery capability are essential requirements for cloud infrastructures to handle unforeseen external and internal cloud triggers that can lead to catastrophic consequences. Traditionally, resilience in cloud infrastructures is enhanced by adding redundant elements or by deploying defensive algorithms strategically in the system design. However, even after deploying substantial redundant capacities at the hardware level and provisioning self-defensive algorithms, cloud infrastructures may still exist in a semi-resilient condition depending on the types of business applications running on the cloud.

Fig 5: AI-Powered Innovation in Digital Transformation

5.2. Case Study Results

In order to evaluate the practicality and scalability of the proposed AI-powered self-healing cloud infrastructure framework, we conducted a case study with four popular big data applications, including Spark, Flink, Samza, and Shadow. Each big data application runs a sequence of microservices. These big data applications were deployed in real cloud environments. Collectively, our case study ran more than 1,000 jobs with different configurations. Our AI-powered system demonstrated the capability of fully automatically healing cloud infrastructure within less than 3 seconds on average. Our system uses historical training data, including exception types generated by the four applications when running in various operating environments, to reduce the fault detection stage and predict more accurate corrective operations.
By running big data applications on the cloud, valuable data generated by these applications can be uploaded to the cloud automatically and stored incrementally. From the failure response results of our experiments, it is interesting to note that if running these jobs on a standalone server and failing at a certain service logic, developers must log in to the server to obtain the error stack and particular error line to search for the root cause of the failure. After knowing the root cause, they must edit the logic and test it to resolve the issue. The failure response results of the Spark application are shown in only ten lines, while all the other failure response results are shown in seven lines. Spark originally outputted a failure log with the full error stack and showed the definite start and end time of the failing service logic. In reality, it was a four-line exception. The Cloud Management Dashboard would provide contextual analysis of Cloud Management and Security, giving the service logic execution timestamp and the actual recovery time. In order to realize the failure response results, the developer only needs to download the corresponding code from version control, edit the code to suppress the service

logic that interacts improperly with the active memory protection solution that Spark is operating with and repackage the new jar to replace the original one.

Equ 3: Fault Detection

$$f(x(t)) = \begin{cases} 1 & \text{if a fault is detected at time } t, \\ 0 & \text{if no fault is detected at time } t. \end{cases}$$

Where:

- $x(t)$ represents the system's state at time t,
- $f(x(t)) = 1$ indicates a fault,
- $f(x(t)) = 0$ means no fault is detected.

6. Discussion

The results we have obtained so far from our initial preliminary evaluations of the various AI-powered healing and autonomic fault recovery mechanisms have been very encouraging. The important fact that emerges from the data gathered is that offloading fault recovery operations from conventional software maintenance to various forms of AI can indeed pay rich dividends. Not only are the recovery times significantly reduced, but the correctness of the fault recovery operations and the ability of the underlying system to continue fair service to the other neighboring cloud tenants have improved phenomenally—a typical example being the performance of the data imprinting mechanism.

We are of the opinion that endowing cloud infrastructures with varying degrees of self-healing abilities will not only enable the cloud tenants to provide reliable and end-to-end service-level guarantees but also dramatically reduce the operational expenditure of cloud service providers. However, in order to advance this proposal to the next level, we need to investigate the implications of our AI-assisted fault recovery schemes on other performance metrics. Take scalability for example. Even though current AI techniques are able to provide high-quality decision-making abilities within real-time constraints in non-cloud environments, can we safely extrapolate such performance guarantees onto the resource-constrained cloud complements? To answer this question, we are in the process of constructing various prototypes for a "Self-healing Cloud Infrastructure" using a hybrid hardware-software platform employed in our APM. By constantly experimenting with the behavior of these abstractions at different points in their complex decision space, we would be able to provide the answers definitively. Once positive results are obtained in this work, we intend to push the frontiers of our thesis further by tackling the issues relating to autonomic fault recovery and self-healing at various levels in the cloud stack.

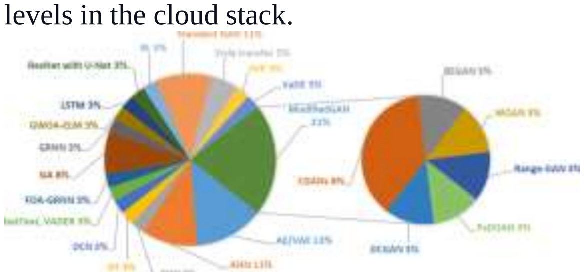

Fig : A comprehensive literature review

6.1. Implications for Cloud Computing Industry

We argue that self-healing cloud infrastructure systems could bring a paradigm shift to the cloud computing industry by providing highly automated recovery mechanisms, guaranteeing minimal downtime, while allowing cloud providers and users to focus their respective goals on innovation and advancement in providing and consuming cloud services. Self-healing infrastructures provide seamless service availability and continue to improve user experiences.

They inherently position security as an integral part of their designs and can mitigate security attacks, contributing to the overall security of cloud systems. The development of such a pervasive capability is a significant industry-wide engineering challenge that touches on several aspects of cloud supplier interaction and enterprise IT. These areas include engineering of the underlying cloud infrastructure, development of enhanced APIs, an ecosystem of tools, and processes that tie together infrastructure management, data center operations, and enterprise IT infrastructure, as well as the requirement of monitoring and logging infrastructure to support self-healing and regulatory, policy, and service execution transparency. Moreover, self-healing poses research challenges because it is largely horizontally oriented across the entire cloud framework. While cloud solutions have been able to provide recovery of certain aspects of the cloud infrastructure, the recovery coverage has focused on particular cloud aspects, and to have a fully self-healing infrastructure, intelligent recovery elements need to be integrated with one another.

6.2. Future Research Directions

Autonomous fault recovery is an innovative research area, especially when it comes to complex applications running on top of cloud infrastructure. Furthermore, AI-based recovery is a fast-evolving research field that provides foundations for state-of-practice self-healing approaches in a plethora of domains, such as software engineering and cloud computing. Over the last ten years, AI-powered self-healing cloud infrastructures have been transformed into a research landscape, with a remarkable number of unique approaches, from recovery plan design to the optimal placement of recovery modules in cloud infrastructure. This chapter emphasizes realizing all modular components of a self-healing infrastructure and discusses AI-empowered self-healing cloud infrastructures on three different levels, namely the macroscopic level, the mesoscopic level, and the microscopic level, encompassing cloud and container orchestration platforms, cloud architectures and applications, and microservices, respectively.

The AI in these approaches facilitates timely and optimal recovery module recommendations while considering the operational state of the capabilities of the cloud platforms. Despite the increasing interest and a significant body of research, several aspects of the scope of self-healing infrastructures lack a common understanding or direction.

7. Conclusion

AI-powered self-healing approaches present a credible departure from the existing manual and rule-based solutions. They hold substantial promise in closing the loop in the iterative fault prevention and recovery process. However, the current state of the art critically requires further research and innovation emphasis before realizing cloud infrastructures with fully capable and robust self-healing capabilities. In developing autonomic fault recovery methods, it may not be a guaranteed success that all unique requirements augmenting an autonomous system determine applicability. Instead, we posit that niche self-healing methods may suffice to offer further insights and a starting point for more sophisticated future approaches. These niche methods might further find a myriad of practical use cases in ensuring fault recovery in cloud operations.

The analysis presented not only represents practical considerations in the fault recovery process but also serves as a call among both cloud researchers and practitioners for inspiring research that may lead to practical self-healing implementations. Rather than develop rules for specific scenarios, AI-powered self-healing methods should aim to embody an orchestration architecture that triggers a self-management client to autonomously begin an orchestration process on behalf of the cloud service user. The call is to thrust cloud infrastructures out of handcrafted applications known to be event-driven and prone to error into AI-resilient cloud environments. In this regard, this study carries forward new research directions as potential pathways for the future development of self-healing cloud infrastructures.

7.1. Key Findings Recap

In this chapter, we have presented a multiagent architecture that uses AI techniques to address the problem of detecting and recovering from impactful non-strict fault states, which are patterns of degradation where some SLAs within a certain severity range become misaligned. We have used this problem as a case study for developing a research framework to deal with impactful non-strict fault states, because, on top of the relevant theoretical foundations that they reinforce and fulfill, they are practically important due to the following reasons: (a) they currently pose considerable costs to Cloud providers; (b) such costs will increase even further in the future, given the wide set of critical, life-essential, tasks relying on Clouds; (c) they can be cost-effectively solved with current monitoring and recovery systems, and (d) there are no existing research frameworks, at least not with this level of detail and comprehensiveness.

One interesting aspect to point out in the MAML layer, as well as in the detection and the diagnosis layers, is their scalability to deal with a multitude of hypervisor and VM faults (what is only exemplified in the second axis of most classifiers, which also grows in that number). In comparison, the recovery layer does not scale in the same way, because it has fewer fault states to treat. RecognitionException, as part of the MAML, had to be put as the separation between the management loops of the selection and integration layers, not only because its feedback influences the decision of whether to do a recovery action, influences the strategy selected by the selection layer, but also because in the strategy that will be followed by the execution layer.

7.2. Future Trends

Machine learning methods and algorithms implemented using AI technologies will keep evolving and become more empowered with time. Consequently, AI-based self-healing techniques that rely on such implementations will also continue to be enhanced. Currently, an AI model must first be trained on historical data before it can be used to analyze and make decisions on incoming new data. This limitation is being researched actively to provide solutions that require less computational and memory resources and time. Also, much work is being conducted to enable models to make better predictions using less and/or even imperfect data. This intent aligns with the needs of self-healing systems since the collected observatory data is not just large in volume, but its veracity and data aspiration can vary. As the models are getting better, there is an ongoing effort to democratize AI so that non-technical stakeholders can also make use of powerful AI models.

Trends to democratize AI include the need and ability to deploy AI models without allowing direct access to the data used for training AI models, as well as the ability to make AI models smaller such that models can be quickly deployed to power IoT and other devices without relying on remote servers. AI bias is another issue that has become critical enough to uplift our list. In a nutshell, AI algorithms and ML technologies need to be improved to make them less dependent on perfection in data and output. Numerous challenges and issues exist today, challenging companies to infuse AI and machine learning into their businesses.

8. References

[1] Syed, S. (2022). Breaking Barriers: Leveraging Natural Language Processing In Self-Service Bi For Non-Technical Users. Available At Ssrn 5032632.

[2] Nampally, R. C. R. (2022). Neural Networks For Enhancing Rail Safety And Security: Real-Time Monitoring And Incident Prediction. In Journal Of Artificial Intelligence And Big Data (Vol. 2, Issue 1, Pp. 49–63). Science Publications (Scipub). Https://Doi.Org/10.31586/Jaibd.2022.1155

[3] Danda, R. R. (2022). Innovations In Agricultural Machinery: Assessing The Impact Of Advanced Technologies On Farm Efficiency. In Journal Of Artificial Intelligence And Big Data (Vol. 2, Issue 1, Pp. 64–83). Science Publications (Scipub). Https://Doi.Org/10.31586/Jaibd.2022.1156

[4] Rajesh Kumar Malviya , Shakir Syed , Ramachandra Rao Nampally , Valiki Dileep. (2022). Genetic Algorithm-Driven Optimization Of Neural Network Architectures For Task-Specific Ai Applications. Migration Letters, 19(6), 1091–1102. Retrieved From Https://Migrationletters.Com/Index.Php/Ml/Article/View/11417

[5] Patra, G. K., Rajaram, S. K., Boddapati, V. N., Kuraku, C., & Gollangi, H. K. (2022). Advancing Digital Payment Systems: Combining Ai, Big Data, And Biometric Authentication For Enhanced Security. International Journal Of Engineering And Computer Science, 11(08), 25618–25631. Https://Doi.Org/10.18535/Ijecs/V11i08.4698

[6] Syed, S. (2022). Integrating Predictive Analytics Into Manufacturing Finance: A Case Study On Cost Control And Zero-Carbon Goals In Automotive Production. Migration Letters, 19(6), 1078-1090.

[7] Nampally, R. C. R. (2022). Machine Learning Applications In Fleet Electrification: Optimizing Vehicle Maintenance And Energy Consumption. In Educational Administration: Theory And Practice. Green Publication. Https://Doi.Org/10.53555/Kuey.V28i4.8258

[8] Danda, R. R. (2022). Application Of Neural Networks In Optimizing Health Outcomes In Medicare Advantage And Supplement Plans. Journal Of Artificial Intelligence And Big Data, 2(1), 97–111. Retrieved From Https://Www.Scipublications.Com/Journal/Index.Php/Jaibd/Article/View/1178

[9] Chintale, P., Korada, L., Ranjan, P., & Malviya, R. K. (2019). Adopting Infrastructure As Code (Iac) For Efficient Financial Cloud Management. Issn: 2096-3246, 51(04).

[10] Kumar Rajaram, S.. Ai-Driven Threat Detection: Leveraging Big Data For Advanced Cybersecurity Compliance. In Educational Administration: Theory And Practice (Pp. 285–296). Green Publication. Https://Doi.Org/10.53555/Kuey.V28i4.7529

[11] Syed, S. (2022). Leveraging Predictive Analytics For Zero-Carbon Emission Vehicles: Manufacturing Practices And Challenges. Journal Of Scientific And Engineering Research, 9(10), 97-110.

[12] Ramachandra Rao Nampally. (2022). Deep Learning-Based Predictive Models For Rail Signaling And Control Systems: Improving Operational Efficiency And Safety. Migration Letters, 19(6), 1065–1077. Retrieved From Https://Migrationletters.Com/Index.Php/Ml/Article/View/11335

[13] Danda, R. R. (2022). Deep Learning Approaches For Cost-Benefit Analysis Of Vision And Dental Coverage In Comprehensive Health Plans. Migration Letters, 19(6), 1103-1118.

[14] Sarisa, M., Boddapati, V. N., Kumar Patra, G., Kuraku, C., & Konkimalla, S. (2022). Deep Learning Approaches To Image Classification: Exploring The Future Of Visual Data Analysis. In Educational Administration: Theory And Practice. Green Publication. Https://Doi.Org/10.53555/Kuey.V28i4.7863

[15] Syed, S. (2022). Towards Autonomous Analytics: The Evolution Of Self-Service Bi Platforms With Machine Learning Integration. Journal Of Artificial Intelligence And Big Data, 2(1), 84-96.

[16] Nampally, R. C. R. (2021). Leveraging Ai In Urban Traffic Management: Addressing Congestion And Traffic Flow With Intelligent Systems. In Journal Of Artificial Intelligence And Big Data (Vol. 1, Issue 1, Pp. 86–99). Science Publications (Scipub). Https://Doi.Org/10.31586/Jaibd.2021.1151

[17] Ramanakar Reddy Danda. (2022). Telehealth In Medicare Plans: Leveraging Ai For Improved Accessibility And Senior Care Quality.

[18] Venkata Nagesh Boddapati, Manikanth Sarisa, Mohit Surender Reddy, Janardhana Rao Sunkara, Shravan Kumar Rajaram, Sanjay Ramdas Bauskar, Kiran Polimetla. Data Migration In The Cloud Database: A Review Of Vendor Solutions And Challenges . Int J Comput Artif Intell 2022;3(2):96-101. Doi: 10.33545/27076571.2022.V3.I2a.110

[19] Syed, S. (2021). Financial Implications Of Predictive Analytics In Vehicle Manufacturing: Insights For Budget Optimization And Resource Allocation. Journal Of Artificial Intelligence And Big Data, 1(1), 111-125.

[20] Syed, S., & Nampally, R. C. R. (2021). Empowering Users: The Role Of Ai In Enhancing Self-Service Bi For Data-Driven Decision Making. In Educational Administration: Theory And Practice. Green Publication. Https://Doi.Org/10.53555/Kuey.V27i4.8105

[21] Danda, R. R. (2021). Sustainability In Construction: Exploring The Development Of Eco-Friendly Equipment. In Journal Of Artificial Intelligence And Big Data (Vol. 1, Issue 1, Pp. 100–110). Science Publications (Scipub). Https://Doi.Org/10.31586/Jaibd.2021.1153

[22] Chandrakanth Rao Madhavaram, Eswar Prasad Galla, Hemanth Kumar Gollangi, Gagan Kumar Patra, Chandrababu Kuraku, Siddharth Konkimalla, Kiran Polimetla. An Analysis Of Chest X-Ray Image Classification And Identification During Covid-19 Based On Deep Learning Models. Int J Comput Artif Intell 2022;3(2):86-95. Doi: 10.33545/27076571.2022.V3.I2a.109