

ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC BÁCH KHOA

BÙI THANH KHIẾT

**GIẢI PHÁP NHẬN DẠNG VÀ XỬ LÝ LỖI TRONG
HẠ TẦNG ĐIỆN TOÁN Đám MÂY**

Ngành: Khoa học máy tính

Mã ngành: 62480101

TÓM TẮT LUẬN ÁN TIẾN SĨ

TP. HỒ CHÍ MINH - NĂM 2022

Công trình được hoàn thành tại **Trường Đại học Bách Khoa – ĐHQG-HCM**

Người hướng dẫn 1: PGS.TS Trần Công Hùng

Người hướng dẫn 2: PGS.TS Phạm Trần Vũ

Phản biện độc lập:

Phản biện độc lập:

Phản biện:

Phản biện:

Phản biện:

Luận án sẽ được bảo vệ trước Hội đồng đánh giá luận án họp tại

.....
.....

vào lúc giờ ngày tháng năm

Có thể tìm hiểu luận án tại thư viện:

- Thư viện Trường Đại học Bách Khoa – ĐHQG-HCM
- Thư viện Đại học Quốc gia Tp.HCM
- Thư viện Khoa học Tổng hợp Tp.HCM

DANH MỤC CÔNG TRÌNH ĐÃ CÔNG BỐ

Tạp chí quốc tế

- [CT1]. Thanh Khiet Bui*, Cong Hung Tran, Tran Vu Pham. “V2PFQL: A proactive fault tolerance approach for cloud-hosted applications in cloud computing environment”. IET Control Theory & Applications 1–25 (2022). <https://doi.org/10.1049/cth2.12324>, ISI-Q1 (SCIE), Impact Factor (2021):2,67.
- [CT2]. Bui, Khiet Thanh*, Len Van Vo, Canh Minh Nguyen, Tran Vu Pham, and Hung Cong Tran. “A fault detection and diagnosis approach for multi-tier application in cloud computing.” Journal of Communications and Networks (JCN), pp.399-414, Vol.22, No.5, October 2020, ISSN 1229-2370 Print/eISSN 1976-5541 Online, 1229-2370/19/\$10.00 c 2020 KICS, DOI: 10.1109/JCN.2020.000023, ISI-Q1 (SCIE), Impact Factor (2021): 3,908.
- [CT3]. Cong Hung Tran, Thanh Khiet Bui*, Tran Vu Pham. “Virtual machine migration policy for multi-tier application in cloud computing based on Q-Learning algorithm”. Computing (2022), 104(6), 1285-1306. DOI:10.1007/s00607-021-01047-0, ISI-Q2 (SCIE), Impact Factor (2021): 2.42.
- [CT4]. Bui Khiet Thanh*, Hung Dac Ho, Tran Vu Pham, and Hung Cong Tran. “Virtual machines migration game approach for multi-tier application in infrastructure as a service cloud computing.” IET Network_The Institution of Engineering and Technology 2020, ISSN 2047-4954, Online ISSN 2047-4962, August 2020, Volume 9, Issue 6, November 2020, p. 326 – 337, doi: 10.1049/iet-net.2019.0204, ISI-Q2 (ESCI).

Kỹ yếu hội nghị quốc tế

- [CT5]. Khiet T. Bui*, Linh V. Nguyen, Tai V. Tran, Tran-Vu Pham, Hung C. Tran, “A load balancing VMs migration approach for multi-tier application in cloud computing based on Fuzzy set and Q-Learning algorithm”. In: Kumar R., Quang N.H., Kumar Solanki V., Cardona M., Pattnaik P.K. (eds) Research in Intelligent and Computing in Engineering. Advances in Intelligent Systems and Computing, vol 1254. Springer, Singapore, 2021 (pp. 617-628).
- [CT6]. Khiet Thanh Bui*, Tran Vu Pham, Hung Cong Tran, “A Load Balancing Game Approach for VM Provision Cloud Computing Based on Ant Colony Optimization.”, ICCASA 2016, © ICST Institute for Computer Sciences, Social Informatics and Telecommunications Engineering 2017 P. Cong Vinh et al.

(Eds.): ICCASA 2016, LNICST 193, pp. 52–63, 2017. DOI: 10.1007/978-3-319-56357-2_6, Springer.

ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC BÁCH KHOA

BÙI THANH KHIẾT

**GIẢI PHÁP NHẬN DẠNG VÀ XỬ LÝ LỖI TRONG
HẠ TẦNG ĐIỆN TOÁN Đám Mây**

Ngành: Khoa học máy tính

Mã ngành: 62480101

TÓM TẮT LUẬN ÁN TIẾN SĨ

TP. HỒ CHÍ MINH - NĂM 2022

Công trình được hoàn thành tại **Trường Đại học Bách Khoa – ĐHQG-HCM**

Người hướng dẫn 1: PGS.TS Trần Công Hùng

Người hướng dẫn 2: PGS.TS Phạm Trần Vũ

Phản biện độc lập:

Phản biện độc lập:

Phản biện:

Phản biện:

Phản biện:

Luận án sẽ được bảo vệ trước Hội đồng đánh giá luận án họp tại

.....
.....

vào lúc giờ ngày tháng năm

Có thể tìm hiểu luận án tại thư viện:

- Thư viện Trường Đại học Bách Khoa – ĐHQG-HCM
- Thư viện Đại học Quốc gia Tp.HCM
- Thư viện Khoa học Tổng hợp Tp.HCM

DANH MỤC CÔNG TRÌNH ĐÃ CÔNG BỐ

Tạp chí quốc tế

- [CT1]. Thanh Khiet Bui*, Cong Hung Tran, Tran Vu Pham. “V2PFQL: A proactive fault tolerance approach for cloud-hosted applications in cloud computing environment”. IET Control Theory & Applications 1–25 (2022). <https://doi.org/10.1049/cth2.12324>, ISI-Q1 (SCIE), Impact Factor (2021):2,67.
- [CT2]. Bui, Khiet Thanh*, Len Van Vo, Canh Minh Nguyen, Tran Vu Pham, and Hung Cong Tran. “A fault detection and diagnosis approach for multi-tier application in cloud computing.” Journal of Communications and Networks (JCN), pp.399-414, Vol.22, No.5, October 2020, ISSN 1229-2370 Print/eISSN 1976-5541 Online, 1229-2370/19/\$10.00 c 2020 KICS, DOI: 10.1109/JCN.2020.000023, ISI-Q1 (SCIE), Impact Factor (2021): 3,908.
- [CT3]. Cong Hung Tran, Thanh Khiet Bui*, Tran Vu Pham. “Virtual machine migration policy for multi-tier application in cloud computing based on Q-Learning algorithm”. Computing (2022), 104(6), 1285-1306. DOI:10.1007/s00607-021-01047-0, ISI-Q2 (SCIE), Impact Factor (2021): 2.42.
- [CT4]. Bui Khiet Thanh*, Hung Dac Ho, Tran Vu Pham, and Hung Cong Tran. “Virtual machines migration game approach for multi-tier application in infrastructure as a service cloud computing.” IET Network_The Institution of Engineering and Technology 2020, ISSN 2047-4954, Online ISSN 2047-4962, August 2020, Volume 9, Issue 6, November 2020, p. 326 – 337, doi: 10.1049/iet-net.2019.0204, ISI-Q2 (ESCI).

Kỹ yếu hội nghị quốc tế

- [CT5]. Khiet T. Bui*, Linh V. Nguyen, Tai V. Tran, Tran-Vu Pham, Hung C. Tran, “A load balancing VMs migration approach for multi-tier application in cloud computing based on Fuzzy set and Q-Learning algorithm”. In: Kumar R., Quang N.H., Kumar Solanki V., Cardona M., Pattnaik P.K. (eds) Research in Intelligent and Computing in Engineering. Advances in Intelligent Systems and Computing, vol 1254. Springer, Singapore, 2021 (pp. 617-628).
- [CT6]. Khiet Thanh Bui*, Tran Vu Pham, Hung Cong Tran, “A Load Balancing Game Approach for VM Provision Cloud Computing Based on Ant Colony Optimization.”, ICCASA 2016, © ICST Institute for Computer Sciences, Social Informatics and Telecommunications Engineering 2017 P. Cong Vinh et al.

(Eds.): ICCASA 2016, LNICST 193, pp. 52–63, 2017. DOI: 10.1007/978-3-319-56357-2_6, Springer.

CHƯƠNG 1 GIỚI THIỆU

1.1 Giới thiệu

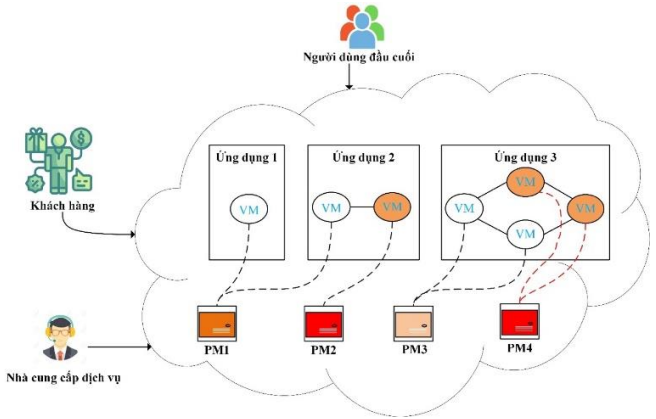
Điện toán đám mây (ĐTĐM) ngày càng trở nên phổ biến, mang lại sự tiện lợi thiết thực, giúp người dùng dễ dàng triển khai các ứng dụng một cách linh hoạt. Kiến trúc của ĐTĐM có thể chia thành hạ tầng vật lý bao gồm các thành phần vật lý như máy chủ, hệ thống lưu trữ, hệ thống mạng, v.v. và tầng trừu tượng tương ứng gồm dịch vụ hạ tầng (Infrastructure as a Service, viết tắt IaaS), dịch vụ nền tảng (Platform as a Service, viết tắt PaaS) dịch vụ phần mềm (Software as a Service, viết tắt SaaS) được triển khai trên hạ tầng vật lý [1]. Hiện nay, các dịch vụ công nghệ thông tin đa số dựa trên các nguồn tài nguyên, cơ cấu vận hành, lưu trữ, phân phối và xử lý thông tin trên ĐTĐM. Thay vì người dùng sử dụng một hoặc nhiều máy chủ vật lý (Physical machine, viết tắt PM), người dùng có thể sử dụng các máy ảo (Virtual machine, viết tắt VM) của dịch vụ cơ sở hạ tầng từ IaaS hay sử dụng dịch vụ nền tảng gồm các API dùng để phát triển các ứng dụng trên một nền tảng công nghệ cụ thể từ PaaS hay sử dụng dịch vụ phần mềm đa phần được cung cấp dưới dạng ứng dụng web và được truy cập từ xa từ SaaS. Thêm vào đó, tính co giãn cùng với mô hình tính phí thuê tài nguyên theo dạng sử-dụng-bao-nhiều-trả-bấy-nhiều (pay-as-you-go) đã góp phần cho ĐTĐM được sử dụng rộng rãi vì khách hàng chỉ trả phí trên tài nguyên, dịch vụ đã sử dụng. Năm 2018, Việt Nam đạt 41/100 điểm và trở thành nước đứng thứ 14 trong bảng xếp hạng “Chỉ số sẵn sàng cho Điện toán đám mây tại khu vực châu Á - Thái Bình Dương”. Trước đó, mức tăng chi tiêu cho ĐTĐM ở Việt Nam giai đoạn 2010-2016 là 64,4% - cao nhất ASEAN. Điều này cho thấy mô hình ĐTĐM đang trở nên phổ biến và bắt đầu chiếm ưu thế hơn so với mô hình CNTT truyền thống. Theo khảo sát và đánh giá của Viettel IDC, khi sử dụng dịch vụ ĐTĐM, doanh nghiệp có thể tiết kiệm tới 40% chi phí đầu tư ban đầu, rút ngắn thời gian triển khai từ 4 - 6 tuần triển khai và loại bỏ hoàn toàn chi phí nhân sự vận hành bảo trì hệ thống so với việc tự đầu tư tại doanh nghiệp. Lỗi có thể xảy ra ở bất kỳ một tầng cụ thể của ĐTĐM và nó sẽ ảnh hưởng lên tầng trên nó. Ví

dụ, nếu lỗi xảy ra ở hệ điều hành tại tầng PaaS có thể dẫn đến các ứng dụng trên SaaS có thể bị lỗi. Trong khi đó nếu lỗi xảy ra ở ổ cứng của máy chủ vật của hạ tầng vật lý xảy ra lỗi, nó sẽ ảnh hưởng lên tầng IaaS và tiếp tục dẫn đến lỗi sẽ xảy ở hệ điều hành của tầng PaaS và tiếp tục ảnh hưởng đến lỗi xảy ra ở ứng dụng của tầng SaaS. Từ đó, phát hiện các lỗi phân cứng điển hình và phát triển các kỹ thuật kháng lỗi tương ứng là một vấn đề cấp thiết. Trong thực tế, vấn đề xây dựng hệ thống kháng lỗi trên hạ tầng ĐTĐM nhằm tăng độ sẵn sàng và giảm tổn thất có thể xảy ra do lỗi trong hệ thống là một trong những thách thức lớn. Từ đó, luận án này tập trung vào nghiên cứu và đề xuất giải pháp kháng lỗi chủ động cho hạ tầng ĐTĐM.

1.2 Tính cấp thiết của luận án

Các khung kháng lỗi tập trung vào hai vấn đề chính là ngăn chặn lỗi và xử lý lỗi. Những phương pháp trong kháng lỗi thụ động phổ biến trong giới nghiên cứu cho đến nay, tuy nhiên, do sự tiến bộ vượt bậc của học máy, trí tuệ nhân tạo, thiết bị ngày càng trở nên thông minh hơn làm gia tăng phạm vi nghiên cứu về kháng lỗi chủ động. Các khung kháng lỗi ngày càng được mong đợi thông minh hơn để đưa ra các chiến lược khác nhau cho các ngữ cảnh khác nhau của lỗi trong hệ thống nhằm kháng được các dạng lỗi khác nhau. Thêm vào đó, việc quản lý và khai thác tài nguyên hiệu quả hiện cũng là một trong các hướng nghiên cứu mở của ĐTĐM và cần được xem xét trong vấn đề kháng lỗi. Một cơ chế của điều phối dịch vụ linh hoạt trong ĐTĐM hướng đến kháng lỗi là rất cần thiết. Nói một cách khác, cần xây dựng một khung kháng lỗi đảm bảo cả độ sẵn sàng cao cũng như quản lý, khai thác tài nguyên hiệu quả. Ví dụ trong Hình 1.1 minh họa trường hợp máy chủ vật lý PM2 và PM4 trên hạ tầng ĐTĐM bị lỗi. Điều đó dẫn đến Ứng dụng 2 và Ứng dụng 3 sẽ gặp lỗi vì các máy ảo chạy những ứng dụng này đang được triển khai trên PM2 và PM4. Theo đó, khách hàng – người thuê VM trên dịch vụ hạ tầng ĐTĐM để triển khai ứng dụng/dịch vụ phục vụ người dùng đầu cuối sẽ bị ảnh hưởng trực tiếp. Do vậy, nhà cung cấp dịch vụ cần phải có giải pháp phát hiện lỗi trên các PM cũng như đưa ra phương án di trú VM đến các PM an toàn khác nhằm tránh lỗi sắp xảy ra trên các PM đáng ngờ cũng

như đảm bảo việc quản lý khai thác tài nguyên trên các PM một cách hiệu quả. Cụ thể như việc di trú các VM vào máy PM1 và PM3 có thể dẫn đến tình trạng không cân bằng về mức độ sử dụng giữa các tài nguyên trong PM và có thể gây ra phân mảnh tài nguyên. Ví dụ, tài nguyên tại một thời điểm của PM được xem xét ở ba khía cạnh là CPU, MEMORY, DISK, nếu một PM có mức độ sử dụng của CPU là 90% trong khi đó lượng mức độ sử của MEMORY và DISK lần lượt là 50% và 10% điều này sẽ dẫn đến sự mất cân bằng về mức độ sử dụng giữa các tài nguyên trong PM và gây ra sự phân mảnh, lãng phí tài nguyên.



Hình 1. 1 Ví dụ trường hợp lỗi máy chủ vật lý trên hạ tầng điện toán đám mây.

Việc phát hiện lỗi thường dựa vào dữ liệu từ các bộ điều khiển, nhật ký sự kiện, hồ sơ hệ thống và bằng cách đo đạt, quan sát các biến, trạng thái của hệ thống. Sau khi đo đạt, quan sát cần có quá trình phân tích, đánh giá các biến và trạng thái của hệ thống cũng như kết hợp với kiến thức của chuyên gia. Mô hình toán học và thống kê là các phương pháp nổi bật được sử dụng cho phát hiện lỗi. Có nhiều đề xuất để giải quyết vấn đề này như phương pháp dựa trên mật độ [12, 13], phương pháp dựa trên mạng nơ-ron nhân tạo, phương pháp dựa trên nhân (kernel) [14], phương pháp dựa trên khai phá dữ liệu [15-17]. Tuy nhiên, việc phát hiện lỗi trên hạ tầng ĐTĐM gặp một số khó khăn, thách thức sau:

- Việc thu thập các thông số của hệ thống từ các tầng như mạng, phần cứng, hệ điều hành, máy ảo, tầng nền tảng, tầng ứng dụng với qui mô hàng ngàn nút xử lý trong hạ tầng ĐTĐM gặp khó khăn.

- Các hệ thống ĐTĐM thương mại cho phép người vận hành hệ thống thiết lập thông số cụ thể để giám sát và vận hành hệ thống ví dụ như công cụ giám sát của các công ty hàng đầu thế giới như Tivoli của IBM, OpenView của HP, CloudWatch của Amazon, SLA-Based Google App của Google [18]. Sau đó, hệ thống sẽ đưa ra những cảnh báo khi các thông số hệ thống vượt ngưỡng đã thiết lập từ trước. Tuy nhiên, việc thiết lập hàng ngàn thông số ngưỡng cho các thông số hạ tầng gặp khó khăn và phụ thuộc vào kinh nghiệm của người vận hành.
- Trong các ứng dụng thực tế, việc thu thập tất cả dữ liệu lỗi rất khó vì rất tốn kém để làm cho hệ thống hoạt động trong điều kiện lỗi. Thêm vào đó, các dạng lỗi của hệ thống rất đa dạng nên khó có thể kết hợp tất cả các loại lỗi trong thực tế để tạo ra bộ dữ liệu huấn luyện chứa tất cả các mẫu của lỗi. Dữ liệu lỗi đa số có độ lệch đáng kể so với dữ liệu bình thường và nó được tạo ra theo một cơ chế khác so với các đối tượng dữ liệu bình thường. Từ đó, sẽ có ít dữ liệu lỗi trong tập dữ liệu huấn luyện.
- Phát hiện lỗi thông qua so sánh các giá trị đo được của các thuộc tính trong hệ thống với những giá trị đo được trong quá trình chuẩn không lỗi. Phương pháp phát hiện thay đổi dựa trên phân lớp thường được sử dụng. Theo đó, kỹ thuật phân lớp là nhằm phân loại một đối tượng chưa biết trở thành một đối tượng thuộc loại đối tượng được định nghĩa trước. Tuy nhiên, khi đối tượng chưa biết không thuộc bất kỳ loại nào đã được định nghĩa thì đây được coi là một khó khăn lớn trong phân loại đối tượng.

Từ góc nhìn của người dùng, ĐTĐM chủ yếu cung cấp cơ sở hạ tầng và các ứng dụng về công nghệ thông tin dưới dạng dịch vụ có khả năng mở rộng được. Nhờ vào công nghệ ảo hóa, người dùng được cung cấp cơ sở hạ tầng như mạng, máy chủ, CPU, bộ nhớ, không gian lưu trữ và các tài nguyên tính toán khác thông qua máy ảo (Virtual machine, viết tắt VM). Tài nguyên ảo hóa được

tổ chức một cách linh động vì lợi ích của các ứng dụng và phần mềm. Ngoài việc cho phép tăng hoặc giảm số lượng VM một cách linh hoạt, công nghệ ảo cho phép di trú toàn bộ hệ thống của VM (gồm vi xử lý, bộ nhớ, lưu trữ, tài nguyên mạng, hệ điều hành, các ứng dụng liên quan) từ PM này sang PM khác [25-27]. Từ đó, đem lại nhiều lợi ích cho việc quản lý tài nguyên ĐTĐM như hợp nhất máy chủ, tối thiểu thời gian gián đoạn, quản lý năng lượng, quản lý lưu lượng mạng, bảo trì trực tuyến, quản lý năng lượng và kháng lỗi [28, 29]. Ví dụ, tình huống hạ tầng ĐTĐM có một số PM bị quá tải, trong khi những PM khác đang chạy không tải; hoặc nếu một PM bị lỗi, tất cả các VM trên đó sẽ bị ảnh hưởng; v.v. được giải quyết cùng với sự ra đời của một công nghệ di trú VM [30]. Việc di trú VM là một trong những công việc điều phối tài nguyên trên ĐTĐM. Theo đó, việc điều phối tài nguyên thường dựa trên các thông tin giám sát như tải CPU, bộ nhớ, thông tin mạng, v.v. cũng như các thiết lập cấu hình được yêu cầu từ khách hàng. Mỗi giải pháp điều phối tài nguyên đã được thiết kế với các mục tiêu cụ thể như tập trung vào kiến trúc ứng dụng, hiệu quả khai thác tài nguyên hay khía cạnh chất lượng dịch vụ với các tiêu chí đánh giá như độ chính xác dự đoán, tuân thủ SLA hoặc chi phí tài nguyên. Việc phân phối VM vào PM cần đảm bảo sử dụng tài nguyên trên PM hiệu quả hướng đến nhà cung cấp dịch vụ hoặc khách hàng hoặc cả hai [31-35]. Một số thách thức cho vấn đề điều phối tài nguyên như sau:

- Mối quan hệ mục tiêu của nhà cung cấp dịch vụ và khách hàng có thể mâu thuẫn với nhau. Môi trường ĐTĐM có dữ liệu phân tán, đòi hỏi có khả năng mở rộng, khả năng đáp ứng yêu cầu người dùng cao, tức đòi hỏi cao về chất lượng dịch vụ (Quality of Service, viết tắt QoS). Trong khi đó, việc duy trì chất lượng dịch vụ thường phụ thuộc vào hai loại của thỏa thuận mức dịch vụ (Service-Level Agreement, viết tắt SLA): SLA tài nguyên là hợp đồng giữa nhà cung cấp và khách hàng – chủ sở hữu ứng dụng triển khai trên hạ tầng đám mây, SLA ứng dụng là hợp đồng giữa khách hàng và người dùng đầu cuối. Cả hai loại SLA này thường được trộn lẫn, có nghĩa nhà cung cấp cần phải tuân thủ SLA tài nguyên để thỏa mãn SLA ứng dụng;

- Đa số tài nguyên vật lý trong môi trường ĐTĐM không đồng nhất với nhau đồng thời yêu cầu tài nguyên từ khách hàng thường cũng không đồng nhất với nhau. Các yêu cầu tài nguyên không đồng nhất có thể gây ra hiện tượng phân mảnh tài nguyên dẫn đến việc lãng phí tài nguyên. Để sử dụng tài nguyên một cách hiệu quả cần sử dụng tối ưu nguồn tài nguyên bằng cách giảm độ lãng phí không gian tài nguyên trong môi trường đa tài nguyên [36];
- Việc điều phối tài nguyên tối ưu là rất cần thiết trong việc sử dụng hiệu quả tài nguyên trong cơ sở hạ tầng ĐTĐM, bài toán tối ưu dạng này thường thuộc lớp NP-Hard hoặc NP-Complete [37]. Việc tìm ra lời giải tối ưu cho các bài toán này cho các hệ thống song song lớn nhất cũng không thể hoàn thành được trong giới hạn thời gian cho phép. Giải pháp cho vấn đề này thường dựa trên đặc tính cụ thể của từng bài toán từ đó áp dụng các thuật toán như vét cạn (exhaustive algorithm), xác định (deterministic algorithm) [38] hoặc meta-heuristic. Trong thực nghiệm, hầu như các thuật toán xác định tốt hơn các giải thuật vét cạn. Tuy nhiên các thuật toán xác định lại không hiệu quả trong môi trường dữ liệu phân tán [39]. Trong khi đó, ĐTĐM là môi trường có dữ liệu phân tán, đòi hỏi có khả năng mở rộng, khả năng đáp ứng yêu cầu người dùng cao.

Để thiết kế khung kháng lỗi chủ động cho dịch vụ hạ tầng ĐTĐM hiệu quả, đạt được mức độ tự động hóa cao, luận án tiếp cận mô hình vòng lặp điều khiển MAPE-K của hệ thống tự trị [49] gồm giám sát (M), phân tích (A), lên kế hoạch (P), thực thi (E), tri thức (K) nhằm giải quyết vấn đề. Theo đó, thông tin nhận được từ thành phần giám sát (M) được sử dụng trong giai đoạn phân tích (A) để phát hiện lỗi trên PM và lập kế hoạch (P) để ước tính các yêu cầu của hệ thống và lên kế hoạch cho hành động di trú VM. Nhà cung cấp dịch vụ sau đó thực hiện (E) các hành động đã được đưa ra trong phần lập kế hoạch (P). Việc lập kế hoạch được xem xét như một chính sách sự kiện - điều kiện - hành động (ECA - Event Condition Action), để thực hiện và nhanh chóng tính toán, hoặc có

dạng hàm tối ưu một số tính năng nhất định của các hệ thống được quản lý. Kế hoạch dựa vào tri thức sẵn có để có thể đạt được mức quản lý một cách tự trị. Thêm vào đó, tri thức có thể được hình thành khi vận hành hệ thống hoặc có thể được phát triển bằng cách rút trích luật từ dữ liệu của quá trình giám sát. Để hiện thực khung kháng lỗi này cần giải quyết các câu hỏi nghiên cứu sau:

- **Câu hỏi 1:** *Có điều gì bất thường đang xảy ra trên máy chủ vật lý hay không? Nếu có nó có thể dẫn đến lỗi hay không? Những thông số hiệu năng nào của máy chủ vật lý liên quan đến lỗi?*
- **Câu hỏi 2:** *Nếu máy chủ vật lý được xác định có lỗi xảy ra, cần có phương án di trú máy ảo như thế nào để tránh lỗi?*

1.3 Đóng góp chính của luận án

Đóng góp chính của luận án bao gồm:

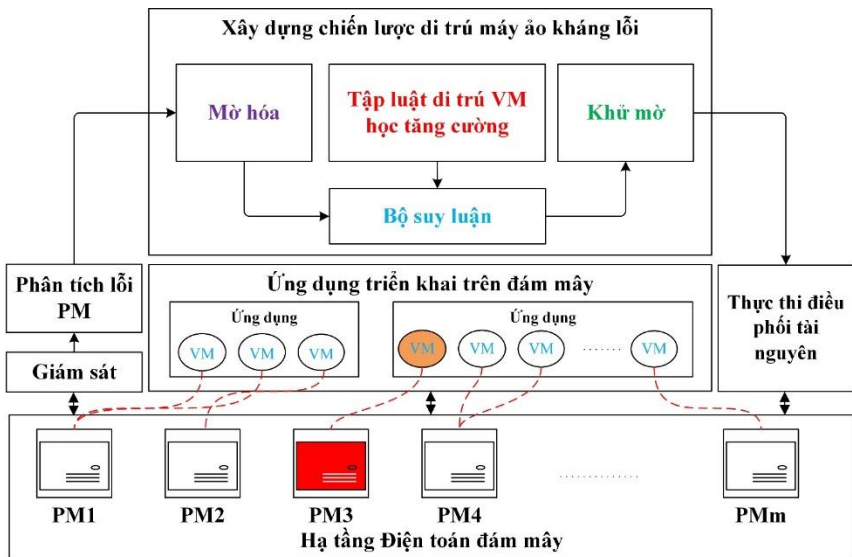
- Xây dựng khung kháng lỗi chủ động cho hạ tầng ĐTĐM dựa trên cấu trúc vòng lặp MAPE-K của hệ thống tự trị gồm thành phần giám sát, phân tích lỗi PM, xây dựng chiến lược chiến lược di trú VM kháng lỗi có khả năng học tăng cường, thực thi điều phối tài nguyên. Kết quả nghiên cứu được công bố trong công trình [CT1][CT4] và được trình bày trong Chương 2 của luận án.
- Đề xuất mô hình phát hiện lỗi trên PM trên hạ tầng ĐTĐM của dựa trên chỉ số vận hành bất thường. Chỉ số vận hành bất thường được xác định từ giá giá trị biên quyết định của mô hình Fuzzy One Class Support Vector Machine (FOCSVM) – là sự kết hợp của logic mờ và OCSVM để giảm ảnh hưởng nhiễu xuất hiện trong tập dữ liệu huấn luyện. Bằng cách sử dụng logic mờ để tính toán các hệ số phạt của mô hình OCSVM nhằm cải thiện hoạt động linh hoạt trong thời gian thực thi cũng như tận dụng kiến thức của chuyên gia. Đề xuất phương pháp phát hiện lỗi, có tên là EWMA-FOCSVM, dựa trên sự theo dõi biến động đột ngột của giá trị biên quyết định trong mô hình FOCSVM bằng biểu đồ Exponentially Weighted Moving Average (EWMA). Các mẫu dữ liệu giám sát được dán nhãn bình thường/lỗi bằng cách sử dụng mô hình EWMA-FOCSVM trong thời gian thực để tạo thành bộ dữ liệu huấn luyện có

nhân cho vấn đề phân tích những thông số hiệu năng của máy chủ vật lý liên quan đến lỗi. Việc phân tích những thông số hiệu năng của máy chủ vật lý liên quan đến lỗi được đưa về bài toán lựa chọn đặc trưng và được giải quyết bằng cách sử dụng mô hình RFE-RF – là sự kết hợp mô hình Recursive Feature Elimination và Random Forest. Các thông số đáng ngờ được xác định thông qua việc xếp hạng thuộc tính của tập dữ liệu. Kết quả nghiên cứu được công bố trong công trình [CT1][CT2] và được trình bày trong Chương 3 của luận án.

- Đề xuất mô hình xây dựng chiến lược di trú máy ảo kháng lỗi dựa trên điều khiển mờ học tăng cường Fuzzy Q-Learning. Việc điều khiển di trú VM để tránh ảnh hưởng từ máy chủ vật lý sắp bị lỗi đảm bảo PM sau tiếp nhận VM có chỉ số vận hành bất thường thấp cũng như mức độ sử dụng giữa các tài nguyên trong PM được cân bằng. Thêm vào đó, để nâng cao khả năng thực thi của bộ điều khiển di trú VM kháng lỗi, thành phần tập luật được cập nhật theo cơ chế học tăng cường ngay cả khi bắt đầu hệ thống với tập luật chưa đầy đủ. Đề xuất giải thuật V2PFQL cho việc điều khiển di trú VM kháng lỗi dựa trên Fuzzy Q-Learning. Một trong những sức mạnh của hệ suy diễn mờ là khả năng chuyên đổi tri thức của con người thành một các luật trực quan dưới dạng NẾU-THÌ. Tuy nhiên, trong quá trình thiết kế hệ suy diễn mờ, người thiết kế có thể gặp vấn đề khó khăn khi định nghĩa tập luật như có không thể thiết kế tập luật vì không có sẵn tri thức của vấn đề, hoặc chỉ có thể định nghĩa một phần của tập luật, hoặc có thể định nghĩa tập luật nhưng không hiệu quả do dư thừa tập luật hoặc do tập luật không chắc chắn (đúng trong một số trường hợp như lại sai trong một số trường hợp khác). Để giải quyết vấn đề này, luận án đề xuất thuật toán huấn luyện tập cho vấn đề di trú máy ảo, được đặt tên V2PFQL-AS, dựa trên sự kết hợp giữa thuật toán V2PFQL và Hệ kiến để hoàn thiện tập luật trong giai đoạn thiết kế hệ suy diễn mờ. Luận án đánh giá hiệu quả của V2PFQL sau khi cập nhật tri thức từ kết quả của quá trình huấn luyện theo thuật toán V2PFQL-AS. Giá trị hàm mục tiêu của bài toán di trú VM kháng lỗi của thuật toán V2PFQL được so sánh với giải thuật RoundRobin (RR); giải thuật tối ưu đàn kiến Inverse Ant System (iAS), giải

thuật hệ kiến Ant System (AS), giải thuật Max-Min Ant System (MMAS), giải thuật tối ưu bầy đàn Particle swarm optimization (PSO), giải thuật luyện kim Simulated Annealing (SA). Kết quả nghiên cứu được công bố trong [CT1][CT3][CT4] [CT5][CT6] công bố trong công trình được trình bày trong Chương 4 của luận án.

CHƯƠNG 2 PHÁT HIỆN VÀ XỬ LÝ LỖI TRONG HẠ TẦNG ĐIỆN TOÁN Đám MÂY



Hình 2. 1 Kiến trúc khung kháng lỗi chủ động cho hạ tầng ĐTĐM.

Mục tiêu của khung kháng lỗi chủ động cho hạ tầng ĐTĐM giúp hệ thống hạ tầng thực thi đúng bằng cách tránh các lỗi tiềm ẩn thông qua biện pháp ngăn chặn. Trạng thái hệ thống được theo dõi liên tục dựa trên thông tin giám sát đại lượng hiệu năng như CPU, bộ nhớ, thông tin mạng, v.v nhằm phát hiện lỗi. Từ kết quả phát hiện lỗi, hệ thống cần có hành động để ngăn chặn lỗi xảy ra. Kỹ thuật di trú VM giúp tăng khả năng chủ động di chuyển VM trên các PM đáng ngờ. Luận án xây dựng khung kháng lỗi chủ động cho dịch vụ hạ tầng ĐTĐM dựa trên vòng lặp MAPE-K của hệ thống tự trị gồm thành phần giám sát, phân tích lỗi PM, xây dựng chiến lược di trú VM kháng lỗi có khả năng

học tăng cường, thực thi điều phối tài nguyên. Theo đó, hai thành phần quan trọng quyết định sự thành công của khung kháng lỗi gồm thành phần phân tích lỗi PM và thành phần xây dựng chiến lược di trú máy ảo kháng lỗi. Hai thành phần này được tập trung nghiên cứu xuyên suốt của luận án. Hình 2.8 mô tả kiến trúc khung kháng lỗi chủ động cho hạ tầng ĐTĐM.

CHƯƠNG 3 PHÂN TÍCH LỖI MÁY CHỦ VẬT LÝ TRÊN HẠ TẦNG ĐIỆN TOÁN Đám Mây

3.1 Giới thiệu

Trong khuôn khổ của luận án, phân tích lỗi PM bao gồm việc phát hiện và phân tích thông số hiệu năng của máy chủ vật lý liên quan đến lỗi của PM khi vận hành. Cụ thể, câu hỏi đặt ra trong phần nghiên cứu này là *“Có điều gì bất thường đang xảy ra trên PM hay không? Nếu có nó có thể dẫn đến lỗi hay không? Và những thông số hiệu năng nào của máy chủ vật lý liên quan đến lỗi?”*

Trong môi trường ĐTĐM, sự dao động của các giá trị quyết định cần phải được xác định liệu chúng có ổn định hay không. Cảnh báo sẽ được đưa ra khi các thông số hệ thống được giám sát không nhất quán với những giá trị quá khứ. Ví dụ, các PM trên hạ tầng ĐTĐM thường được kiểm tra đầy đủ, chúng hoạt động bình thường trong hầu hết các trường hợp. Thông qua giám sát trạng thái hệ thống và xem như bỏ qua các lỗi vật lý thì hầu như các PM là hoạt động bình thường, tuy nhiên khi có một số lượng lớn thay đổi đột ngột khối lượng truy cập làm ảnh hưởng đến thông số hệ thống hoặc bị “hacker” tấn công thì sẽ gây ra sự bất thường cho hệ thống và được thông báo là có lỗi hoặc đưa ra cảnh báo khi giá trị thông số của hệ thống hiện tại không phù hợp với quá khứ. Do vậy, cần phải có chỉ số biểu diễn diễn sự bất thường của PM cũng như cần một biểu đồ kiểm soát để theo dõi và tính toán sự ổn định của giá trị quyết định. Luận án đề xuất chỉ số vận hành bất thường của PM được tính toán dựa vào các thông số về đại lượng hiệu năng của PM như chỉ số về CPU, RAM, DISK, NETWORKS, .v.v. Để giải quyết vấn đề có nhiều trong tập huấn luyện, các yếu tố phạt thích nghi được đề xuất sử dụng hàm thuộc của logic mờ cho các điểm dữ liệu huấn

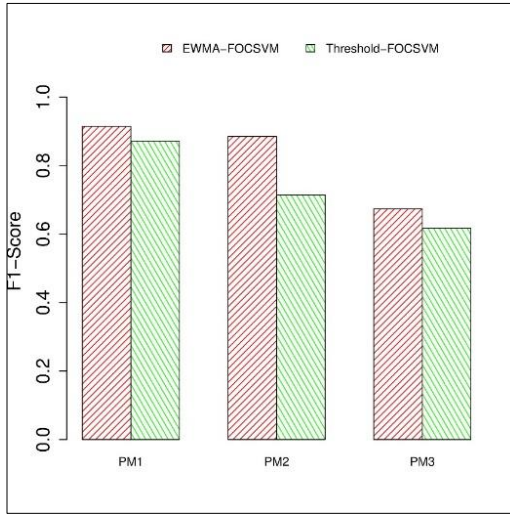
luyện cho mô hình OCSVM (được đặt tên là Fuzzy One-class Support Vector Machine, viết tắt FOCSVM). Các yếu tố hình phạt thích nghi dựa trên hàm liên thuộc mờ biểu diễn khoảng cách của mỗi quan hệ giữa một điểm dữ liệu và trung tâm của của tập dữ liệu huấn luyện. Các chỉ số hiệu năng được xem là đầu vào cho việc tính giá trị biên quyết định của mô hình FOCSVM. Dựa trên chỉ số vận hành bất thường, việc phát hiện lỗi vận hành của PM được xác định thông qua biểu đồ kiểm soát trung bình trượt có trọng số hàm mũ (Exponentially Weighted Moving Average, viết tắt EWMA), mô hình phát hiện lỗi được đặt tên EWMA-FOCSVM. Trong kiểm soát chất lượng thống kê, biểu đồ kiểm soát trung bình trượt có trọng số hàm mũ EWMA được sử dụng để giám sát giá trị biên quyết định. EWMA là một kỹ thuật phân tích tuần tự để phát hiện các thay đổi trong kiểm soát chất lượng thống kê. EWMA có chi phí tính toán thấp phát hiện thay đổi đột ngột [24]. Hơn thế nữa, EWMA không đòi hỏi tri thức cho việc phát hiện lỗi dựa trên ngưỡng. Lỗi có thể là nguyên nhân gây ra sự biến động đột ngột của thông số hệ thống [24]. Các lỗi dẫn đến việc tăng dao động của các thông số hệ thống, do đó, việc xác định các dao động của thông số là rất quan trọng để tìm ra nguồn gây lỗi. Theo đó, mỗi mẫu dữ liệu giám sát sẽ chứa nhiều loại thông tin khác của thông số hệ thống, các biến động bất thường của các thông số này trước và sau khi xảy ra lỗi có thể giúp chúng ta định vị lỗi đang xảy ra tại thông số hệ thống nào đó. Từ đó, định vị các thông số đáng ngờ liên quan đến nguyên nhân gây ra lỗi có thể được chuyển về vấn đề lựa chọn thuộc tính. Vấn đề phân tích thông số hiệu năng của máy chủ vật lý liên quan đến lỗi được đưa về bài toán lựa chọn đặc trưng với tập dữ liệu huấn luyện được gắn nhãn dựa trên đầu ra của giai đoạn phát hiện lỗi. Sau khi phát hiện lỗi, dữ liệu giám sát trực tuyến được trước khi phát hiện lỗi gắn nhãn là Normal và các trường hợp dữ liệu sau khi phát hiện lỗi là Fault. Để giải quyết vấn đề phân tích thông số hiệu năng của máy chủ vật lý liên quan đến lỗi, luận án đề xuất áp dụng phương pháp Recursive Feature Elimination (RFE) kết hợp với thuật toán Rừng ngẫu nhiên (RF) trên mỗi lần lặp được áp dụng [122, 123] để xây dựng mô hình phân tích thông số hiệu năng của máy chủ vật lý liên quan đến lỗi, được đặt tên REF-RF.

3.2 Đánh giá mô hình trên dữ liệu Google Cluster Trace

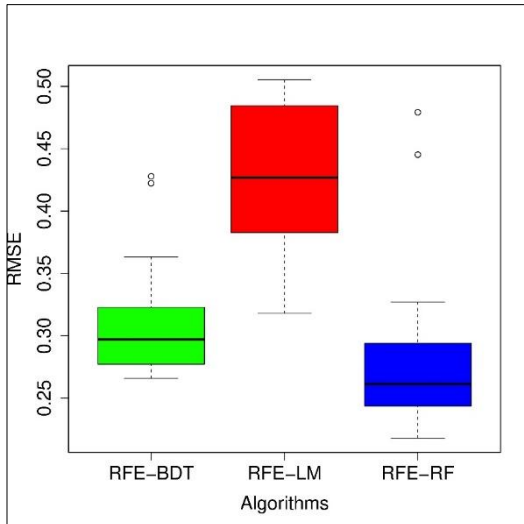
Google cluster là một tập hợp các PM được kết nối cụm bằng thông cao. Mỗi ô là một tập các máy chủ được xem như một cụm và quản lý bởi hệ thống quản lý cụm Borg [140]. Dữ liệu theo dõi khối lượng công việc do Google công bố chứa một số bảng theo dõi trạng thái của PM, công việc và nhiệm vụ trong khoảng thời gian khoảng 29 ngày cho một cụm trong tổng số 12.453 máy chủ vật lý [141]. Trong đó, dữ liệu về sự kiện kiện tác vụ có hơn hơn 100 triệu bản ghi (khoảng 17GB chưa nén) và kéo theo nhật ký sử dụng tác vụ có khoảng 1 tỷ bản ghi (178GB chưa nén), các thông số tài nguyên cho mỗi tác vụ được thu thập vào khoảng 5 phút/lần. Sîrbu và công sự [130] đã xử lý dữ liệu lớn Google Cluster Trace (GCT) này trên nền tảng BigQuery¹ để cho ra bộ dữ liệu gồm 12 thuộc tính cơ bản gồm bảy thuộc tính liên quan đến tác vụ gồm evicted, failed, finished normally, killed và lost và năm thuộc tính đo tải của máy chủ vật lý gồm CPU, Memory, Disk Time, cycles per instruction(CPI) và memory accesses per instruction (MAI) trong khoảng thời gian 5 phút/lần lấy mẫu. Trong khuôn khổ của Luận án, Luận án trích xuất dữ liệu đã được xử lý từ nghiên cứu của Sîrbu và các đồng nghiệp với năm thuộc tính đo tải của PM. Để đánh giá mô hình, luận án sử dụng một phần nhỏ dữ liệu lấy từ nghiên cứu của Sîrbu và công sự với ba máy chủ được ký hiệu là PM1, PM2, PM3.

Hình 3.16 trình bày độ chính xác của mô hình phát hiện lỗi EWMA-FOCSVM được đối sánh với mô hình Threshold-FOCSVM. Kết quả cho thấy độ chính xác của mô hình EWMA-FOCSVM cao hơn mô hình Threshold-FOCSVM.

¹ <https://cloud.google.com/bigquery>



Hình 3. 1 So sánh F1-Score giữa mô hình EWMA-FOCSVM và mô hình Threshold-FOCSVM trên dữ liệu GCT



Hình 3. 2 Kết quả RMSE trung bình của thuật toán RFE-BDT, RFE-LM, RFE-RF trên dữ liệu GCT

Mô hình phân tích thông số hiệu năng của máy chủ vật lý liên quan đến lỗi RFE-RF được so sánh với mô hình RFE kết hợp với Bagged Decision Trees (RFE-BDT) và mô hình RFE kết hợp với Linear Regression (RFE-LM). Hình 3.20 biểu diễn Kết quả RMSE trung bình của thuật toán RFE-RF, RFE-BDT, RFE-LM và cho thấy RMSE trung bình của RFE-RF thấp hơn RFE-BDT và RFE-LM.

CHƯƠNG 4 XÂY DỰNG CHIẾN LƯỢC DI TRÚ MÁY ẢO KHÁNG LỖI TRÊN HẠ TẦNG ĐIỆN TOÁN Đám Mây

4.1 Giới thiệu

Việc xây dựng chiến lược kháng lỗi chủ động được hiện thực dựa trên ba kỹ thuật gồm tự Self-Healing (tự điều chỉnh), Preemptive migration (di trú ưu tiên) và Software Rejuvenation (làm tươi phần mềm). Theo đó, kỹ thuật di trú ưu tiên phù hợp với việc xây dựng chiến lược kháng lỗi chủ động cho dịch vụ hạ tầng ĐTDĐ. Từ đó, nó đem lại nhiều lợi ích cho các trung tâm dữ liệu như cân bằng tải, bảo trì trực tuyến, quản lý năng lượng, và kháng lỗi [28, 29]. Việc phân phối VM vào PM cần đảm bảo sử dụng tài nguyên trên PM hiệu quả hướng đến nhà cung cấp dịch vụ hoặc khách hàng hoặc cả hai [31-35]. Câu hỏi đặt ra trong phần nghiên cứu này là “*Nếu máy chủ vật lý được xác định có lỗi xảy ra, cần có phương án di trú máy ảo như thế nào để tránh lỗi?*”. Trong khuôn khổ của luận án, vấn đề di trú VM được xem xét các khía cạnh (i) mức độ sử dụng cân bằng giữa các tài nguyên trong PM nhằm tránh sự quá tải ở bất kỳ thông số tài nguyên nào của PM có thể dẫn đến hiệu suất bị giảm sút và gây ra lỗi, (ii) giải pháp di trú VM kháng lỗi đảm bảo PM sau tiếp nhận VM có chỉ số vận hành bất thường thấp nhằm giảm nguy cơ gây ra lỗi, (iii) đảm bảo một VM chỉ được di trú đến một PM.

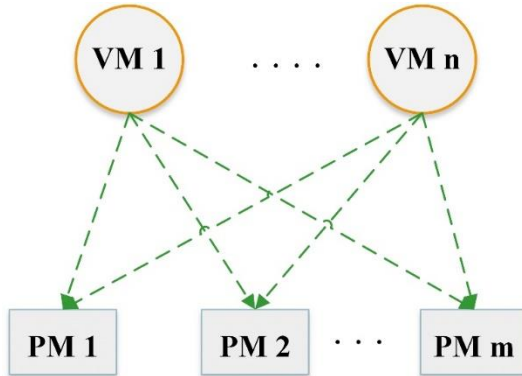
Chiến lược di trú VM được xem như tri thức bên trong của bộ điều khiển đám mây sao cho chúng có khả năng học trong môi trường thực thi. Luận án xây dựng một bộ điều khiển di trú VM kháng lỗi mà trong đó tập luật di trú VM có

khả năng học tăng cường theo cơ chế MAPE-K. Thuật toán điều khiển di trú V2PFQL được xây dựng dựa trên thuật toán học tăng cường mờ Fuzzy Q-Learning [47, 120, 121]. Một trong những sức mạnh của hệ suy diễn mờ là khả năng chuyển đổi tri thức của con người thành một các luật trực quan dưới dạng NẾU-THÌ. Tuy nhiên, trong quá trình thiết kế hệ suy diễn mờ, người thiết kế có thể gặp vấn đề khó khăn khi định nghĩa tập luật như có không thể thiết kế tập luật vì không có sẵn tri thức của vấn đề, hoặc chỉ có thể định nghĩa một phần của tập luật, hoặc có thể định nghĩa tập luật nhưng không hiệu quả do dư thừa tập luật hoặc do tập luật không chắc chắn (đúng trong một số trường hợp như lại sai trong một số trường hợp khác). Để giải quyết vấn đề này, luận án đề xuất thuật toán huấn luyện tập luật cho vấn đề di trú máy ảo, được đặt tên V2PFQL-AS, dựa trên sự kết hợp giữa thuật toán V2PFQL và Hệ kiến [142, 143] để hoàn thiện tập luật trong giai đoạn thiết kế hệ suy diễn mờ. Luận án đánh giá hiệu quả của bộ điều khiển di trú VM thông qua hai pha gồm pha huấn luyện tập luật di trú VM và pha khai thác tập luật di trú VM. Trong pha huấn luyện tập luật di trú, các thông số điều khiển của thuật toán V2PFQL-AS được xem xét gồm hệ số chiết khấu, hệ số khám phá/khai thác và hệ số học của quá trình học tăng cường. Kết quả tập luật di trú VM từ pha huấn luyện được sử dụng như giá trị đầu vào cho thuật toán V2PFQL. Trong pha khai thác tập luật di trú VM, thuật toán điều khiển di trú VM kháng lỗi V2PFQL được đánh giá thông qua giá trị hàm phần thưởng được tính dựa trên mức độ cân bằng tải giữa các tài nguyên một PM và mức độ bất thường của PM. Luận án cũng so sánh kết quả hàm mục tiêu của thuật toán V2PFQL trong bài toán di trú VM kháng lỗi trong với các thuật toán RoundRobin (RR), Inverse Ant System (iAS) [143], Ant System (AS), Max-Min Ant System (MMAS)[144], Simulated Annealing (SA) [145], Particale Swarm Optimization (PSO)[146].

4.2 Thuật toán V2PFQL

Giả sử một trung tâm dữ liệu cung cấp dịch vụ hạ tầng đám mây cho nhiều ứng dụng. Mỗi ứng dụng triển khai trên cụm nhiều VM. Quá trình di trú VM sẽ được thực thi ngay khi phát hiện có một hoặc nhiều PM bị lỗi thông qua

thành phần phát hiện lỗi. Để không mất tính tổng quát, tại thời điểm đó dịch vụ hạ tầng có n VM cần được di trú đến m PM an toàn. Để biểu diễn sự không đồng nhất về mặt tài nguyên của hệ thống, k loại tài nguyên ví dụ (CPU, MEMORY, DISK) được xem xét cho cả PM và VM. Vấn đề đặt ra làm sao triển khai n VM lên m PM an toàn một cách hợp lý. Có thể mô hình hóa bài toán di trú VM trên ĐTĐM dưới dạng đồ thị có hướng (Directed Acyclic Graph, viết tắt DAG) [147-149] $G(V,E)$ với trong đó V là tập đỉnh thể hiện các công việc, E là tập các cạnh có hướng thể hiện mối quan hệ phụ thuộc giữa các đỉnh như trong Hình 4.2.



Hình 4. 1 Đồ thị có hướng cho bài toán di trú máy ảo

Thuật toán 4.1. Điều khiển di trú VM học tăng cường V2PFQL

Đầu vào:

Hệ số học: η

Hệ số chiết khấu: γ

Hệ số khám phá/khai thác: ϵ

Đầu ra: Giải pháp di trú VM

- 1: Khởi tạo giá trị $q[i, j] = 0, i \in \mathcal{S}, j \in \mathcal{A}$
- 2: Tính toán hành động cho trạng thái hiện tại:
- 3: $a = \text{random} \{j \in \mathcal{A}\}$ với xác suất ϵ
- 4: $a = \text{agr} \max_j q[i, j]$ với xác suất $1 - \epsilon$
- 5: Xấp xỉ hàm Q trạng thái s_t theo Công thức (4.33)
- 6:
$$Q(s_t, a) = \frac{\sum_{i=1}^L \theta_i(s_t) \times q[i, a]}{\sum_{i=1}^L \theta_i(s_t)}$$
- 7: Thực thi hành động a và hệ thống chuyển lên trạng thái s_{t+1}
- 8: Tính toán Q cho trạng thái mới theo Công thức (4.34)

- 9:
$$Q(s_{t+1}) = \frac{\sum_{i=1}^L \theta_i(s_{t+1}) \times \max_j [i + 1, j]}{\sum_{i=1}^L \theta_i(s_{t+1})}$$
- 10: Cập nhật giá trị q theo Công thức (4.29)
- 11:
$$q[i, a] = (1 - \eta)q[i, a] + \eta[r_{t+1} + \gamma Q(s_{t+1}) - Q(s_t, a)]$$
- 12: Lặp lại bước 1 cho trạng thái mới
-

Với việc mở rộng thuật toán Q-Learning, hàm Q tính toán theo hệ suy diễn mờ trong thuật toán Fuzzy Q-Learning để quản lý cặp trạng thái-hành động mà trong đó hàm Q được biểu diễn bởi một hệ suy diễn mờ. Giả sử thời điểm t hệ thống có trạng thái s_t (theo Định nghĩa 4.1) bộ điều khiển có tập hành động khả thi $\mathcal{A} = \{PM_1, PM_2, \dots, PM_m\}$ là tập PM an toàn để triển khai VM cần di trú. Đối với mỗi luật, cần phải chọn một trong các hành động \hat{a} tương ứng với mức độ phù hợp dựa trên giá trị q (được tính từ hàm Q), được biểu diễn như sau:

R_i : **NẾU** $s_i = s_t \{S(PM_1) \text{ là } p_1, \dots, S(PM_m) \text{ là } p_p, S(VM_j) \text{ là } v_k \}$ **THÌ**

$$\hat{a} = PM_1 \text{ với } q_{i1} \text{ hoặc}$$

$$\hat{a} = PM_2 \text{ với } q_{i2} \text{ hoặc} \quad (4.1)$$

⋮

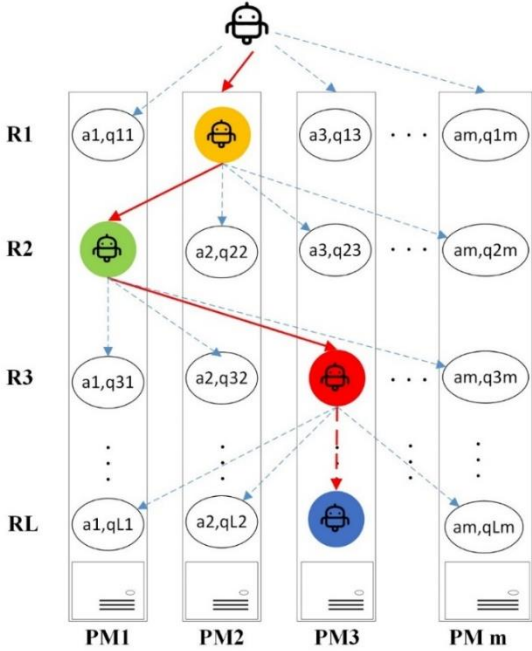
$$\hat{a} = PM_m \text{ với } q_{im}$$

trong đó R_i là tập mờ trong luật thứ i , $S(PM_m)$ là trạng thái của PM thứ m (có giá trị là p_p), $S(VM_j)$ là dạng của VM thứ j (có giá trị là v_k), \hat{a} là biến hành động đầu ra, q_{im} là giá trị q biểu diễn mức độ phù hợp khi chọn PM thứ m cho việc di trú VM thứ j khi trạng thái của toàn bộ hệ thống là s_i .

4.3 Thuật toán V2PFQL-AS

Giả sử hệ suy diễn mờ có tập luật $L = \{R_1, R_2, \dots, R_L\}$ và tập hành động $\mathcal{A} = \{a_1, a_2, \dots, a_m\}$. Việc lựa chọn một trong các hành động tương ứng với mức độ phù hợp dựa trên giá trị q (được tính từ hàm Q) cho mỗi luật như trong Công thức (4.34) được xem như bài toán tổ hợp và Hệ kiến được sử dụng để tìm ra phương án tối ưu cục bộ hoặc toàn cục. Ý tưởng được trình bày trong Hình 4.5 tại mỗi bước tương ứng với một luật trong tập L con kiến có thể chọn một hành

động tập \mathcal{A} . Cụ thể, tại bước khởi tạo ban đầu cho luật R_1 , hành động a_2 được chọn tương ứng với PM2 cho luật R_1 , lúc này (a_2, q_{12}) tương ứng với một nút trên đồ thị biểu diễn giá trị q của luật R_1 khi chọn hành động a_2 . Tiếp theo, con kiến sẽ đi chuyển qua hết các nút của đồ thị tương ứng ta có vết mùi của con kiến. Đối với mỗi luật, nút được chọn bởi con kiến giúp hoàn thiện một phần thông tin hệ quả của luật. Việc lựa chọn một phần thông tin hệ quả của luật dựa trên các dựa trên sự kết hợp giữa vết mùi và luật.



Hình 4. 2 Sơ đồ mối quan hệ giữa tập luật, hành động, giá trị q và vết mùi.

4.4 Đánh giá thuật toán điều khiển di trú máy ảo

Hệ điều khiển di trú VM được triển khai trên trung tâm ĐTĐM cung cấp dịch vụ hạ tầng và được phát triển dựa trên công cụ mô phỏng CloudSim. Việc đánh giá hệ điều khiển di trú VM được chia thành pha huấn luyện tập luật và pha khai thác tập luật. Ban đầu, pha huấn luyện tập luật được tạo ra bởi thuật toán V2PFQL-AS. Sau đó, trong pha khai thác luật, bộ điều khiển di trú VM sẽ thực thi thuật toán V2PFQL dựa trên tập luật đã thu được ở pha huấn luyện tập luật.

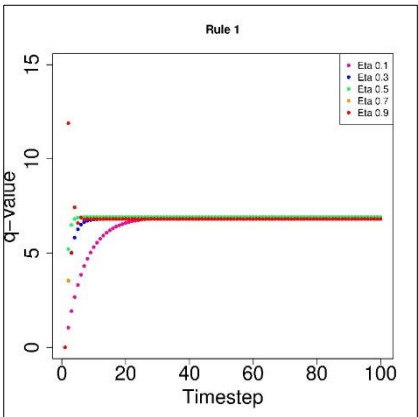
Trong quá trình thực thi, thành phần cơ sở luật của bộ điều khiển được cập nhật liên tục. Để đánh giá hiệu quả của thuật toán di trú VM, giả sử tại mỗi thời điểm rời rạc, bộ phân tích sẽ đánh giá trạng thái của hệ thống để từ đó đưa ra các hành động di trú VM cụ thể.

4.4.1 Đánh giá thuật toán huấn luyện tập luật di trú máy ảo V2PFQL-AS

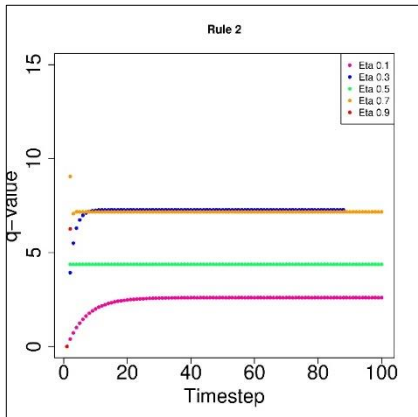
Để đánh giá thuật toán huấn luyện tập luật, các tham số ảnh hưởng đến quá trình học tăng cường được xem xét gồm hệ số học η , hệ số chiết khấu γ , hệ số khám phá/khai thác ε trong Công thức (4.29) dựa trên sự hội tụ giá trị của hàm Q (hay còn gọi là q-value).

4.4.1.1 Điều chỉnh hệ số học

Đầu tiên, cố định hệ số chiết khấu $\gamma = 0.5$, hệ số khám phá/khai thác $\varepsilon = 0.5$, sau đó điều chỉnh hệ số học $\eta \in [0.1, 0.9]$. Trong Hình 4.6 biểu diễn quá trình hội tụ của q-value của bốn luật theo các giá trị hệ số học η . Ví dụ, như trong Hình 4.6 (a) của luật Rule1, ban đầu giá trị q-value khác nhau nhưng từ khoảng thời gian timestep=30 trở về sau thì hội tụ và cùng giá trị q-value.



(a) Giá trị q-value của luật 1 (Rule1)

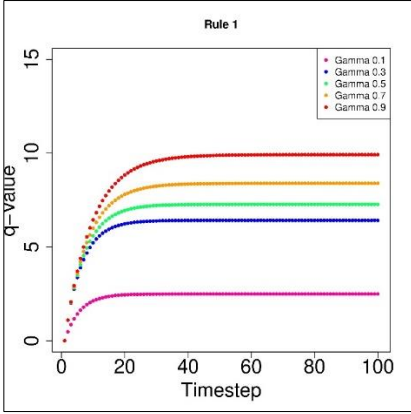


(b) Giá trị q-value của luật 2 (Rule2)

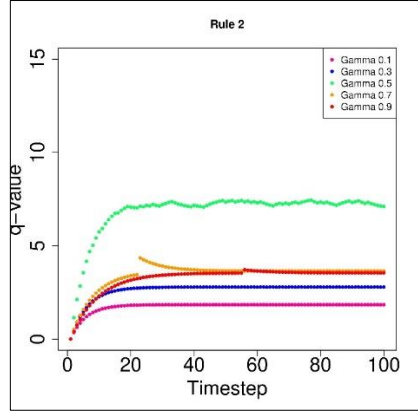
Hình 4. 3 Kết quả giá trị q của 4 luật theo hệ số học $\eta \in [0.1, 0.9]$ với hệ số chiết khấu $\gamma = 0.5$, hệ số khám phá/khai thác $\varepsilon = 0.5$

4.4.1.2 Điều chỉnh hệ số chiết khấu

Sự ảnh hưởng của hệ số chiết khấu được đánh giá thông qua việc điều chỉnh giá trị hệ số chiết khấu $\gamma \in [0.1, 0.9]$ và chọn hệ số khám phá/khai thác $\varepsilon = 0.5$, hệ số học $\eta = 0.1$. Hình 4.7 biểu diễn quá trình hội tụ của q-value của bốn luật theo các giá trị hệ số chiết khấu γ .



(a) Giá trị q-value của luật 1 (Rule1)

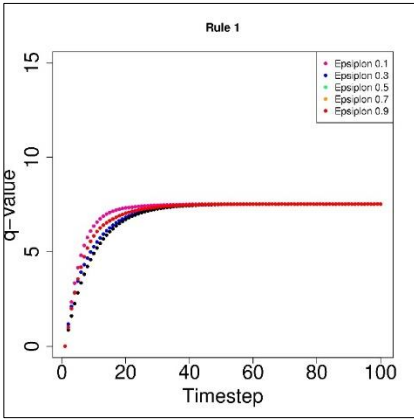


(b) Giá trị q-value của luật 2 (Rule2)

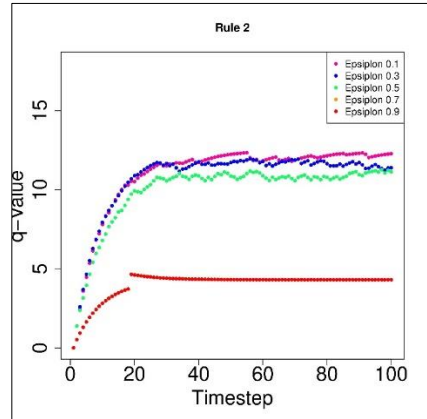
Hình 4. 4 Kết quả giá trị q-value của bốn luật theo hệ số chiết khấu $\gamma \in [0.1, 0.9]$ và hệ số khám phá/khai thác $\varepsilon = 0.5$, hệ số học $\eta = 0.1$

4.4.1.3 Điều chỉnh hệ số khám phá/khai thác

Sự ảnh hưởng của hệ số khám phá/khai thác được đánh giá thông qua việc điều chỉnh giá trị hệ khám phá/khai thác $\varepsilon \in [0.1, 0.9]$, chọn hệ số chiết khấu $\gamma = 0.9$, hệ số học $\eta = 0.1$. Hình 4.8 biểu diễn quá trình hội tụ của q-value của bốn luật theo các giá trị hệ số khám phá/khai thác ε .



(a) Giá trị q-value của luật 1 (Rule1)

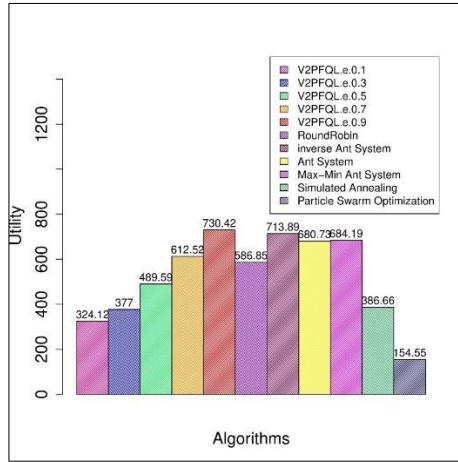


(b) Giá trị q-value của luật 2 (Rule2)

Hình 4. 5 Kết quả giá trị q-value theo hệ khám phá/khai thác $\epsilon \in [0.1,0.9]$, hệ số chiết khấu $\gamma = 0.9$, hệ số học $\eta = 0.1$.

4.4.2 Đánh giá thuật toán điều khiển di trú máy ảo V2PFQL

Xét một cách tổng quát, bài toán di trú VM kháng được biểu diễn theo dạng có một n VM cần được di trú vào m PM, theo đó, sau khi di trú VM hệ thống đảm bảo tối thiểu mức độ cân bằng tải giữa các tài nguyên trong từng PM, đảm bảo tối thiểu mức độ bất thường của từng PM, tránh sự quá tải bất kỳ PM nào có thể dẫn đến hiệu suất bị giảm sút và đảm bảo một VM chỉ được di trú đến một PM. Trong khi đó thuật toán V2PFQL là thuật toán điều khiển theo thời t mà tại mỗi thời điểm chỉ có một VM được di trú đến một PM an toàn. Do vậy, để đánh giá kết quả hàm mục tiêu của bài toán di trú VM kháng lỗi của thuật toán V2PFQL cần xem xét tại thời điểm di trú xong n VM vào m PM. Trong khuôn khổ của luận án, việc so sánh kết quả hàm mục tiêu của bài toán di trú VM kháng lỗi của thuật toán V2PFQL với các thuật toán thuộc lớp meta-heuristic gồm thuật toán RR, iAS, AS, MMAS, SA, PSO. Để đánh giá khả năng khám phá/khai thác của thuật toán V2PFQL, các hệ số khám phá/khai thác được điều chỉnh $\epsilon \in [0.1,0.9]$ và chọn hệ số học $= 0.1$, hệ số chiết khấu $\gamma = 0.9$. Kết quả của thuật toán V2PFQL theo các cấu hình theo ϵ được ký hiệu lần lượt từ V2PFQL.e.0.1 đến V2PFQL.e.0.9.



Hình 4. 6 Kết quả giá trị hàm mục tiêu (Utility) của các thuật toán cho bài toán di trú máy ảo kháng lỗi.

Hình 4.11 biểu diễn kết quả giá trị hàm mục tiêu (Utility) của các thuật toán cho bài toán di trú VM kháng lỗi, theo đó, kết quả cho thấy kết quả của thuật toán V2PFQL.e.0.9 tương ứng với $\varepsilon = 0.9$ cho kết quả tốt nhất. Điều này cho thấy, với việc lựa chọn hành động dựa trên hành động mới cho thấy khả năng khám phá tốt của thuật toán V2PFQL.

CHƯƠNG 5 TỔNG KẾT

5.1 Tóm tắt

Để giải quyết Câu hỏi 1, luận án đề xuất một phương pháp máy vector hỗ trợ một lớp mờ (FOCSVM) để phát hiện bất thường. Bộ dữ liệu thực tế luôn tồn tại nhiều vì các lý do như lỗi dụng cụ, lỗi định dạng và lấy mẫu không biểu diễn. Để khắc phục những khiếm khuyết này, trong nghiên cứu, sự kết hợp của logic mờ và OCSVM (có tên FOCSVM) được đề xuất để cải thiện việc phát hiện bất thường khi có nhiều xuất hiện trong tập dữ liệu. Đối với vấn đề phát hiện lỗi, biểu đồ kiểm soát trung bình trượt có trọng số hàm mũ (Exponentially Weighted

Moving Average, viết tắt EWMA) được sử dụng để xác định những thay đổi đột ngột nếu có bất kỳ lỗi nào xảy ra, được đặt tên EWMA-FOCSVM. Và sau đó, vấn đề chẩn đoán lỗi được đưa về bài toán lựa chọn đặc trưng với tập dữ liệu huấn luyện được gắn nhãn bởi thành phần phát hiện lỗi EWMA-FOCSVM. Để giải quyết vấn đề phân tích thông số hiệu năng của máy chủ vật lý liên quan đến lỗi, phương pháp Recursive Feature Elimination (RFE) kết hợp với thuật toán Rừng ngẫu nhiên (Random Forest, viết tắt RF) trên mỗi lần lặp được áp dụng để xây dựng mô hình chẩn đoán lỗi (có tên REF-RF).

Để giải quyết Câu hỏi 2, cần xây dựng chiến lược di trú VM kháng lỗi dựa trên công nghệ di trú VM. Công nghệ di trú VM cho phép di chuyển toàn bộ hệ thống của VM (gồm vi xử lý, bộ nhớ, lưu trữ, tài nguyên mạng, hệ điều hành, các ứng dụng liên quan) từ PM này sang PM khác. Trong khuôn khổ của luận án, chiến lược di trú VM được xem như tri thức bên trong của bộ điều khiển đám mây sao cho chúng có khả năng học trong môi trường thực thi. Luận án xây dựng một bộ điều khiển di trú VM mà trong đó tập luật di trú VM có khả năng học tăng cường theo cơ chế MAPE-K. Thuật toán điều khiển di trú máy ảo V2PFQL được xây dựng dựa trên thuật toán học tăng cường mờ Fuzzy Q-Learning. Thêm vào đó, để nâng cao khả năng thực thi của bộ điều khiển di trú VM, thành phần tập luật có khả năng học tăng cường để hoàn thiện tập luật trong suốt thời gian thực. Thuật toán huấn luyện tập luật di trú máy ảo V2PFQL-AS được thiết kế dựa trên sự kết hợp giữa V2PFQL và Hệ kiến để hoàn thiện tập luật trong giai đoạn thiết kế hệ suy diễn mờ của bộ điều khiển di trú VM. Hiệu quả của giải thuật huấn luyện tập luật V2PFQL-AS được đánh giá bằng việc điều chỉnh hệ số học, hệ số chiết khấu, hệ số khám phá/khai thác dựa trên sự hội tụ của giá trị q -value. Giải thuật điều khiển di trú máy ảo V2PFQL được đánh giá và so sánh với giải thuật thuộc lớp meta-heuristic RR, iAS, AS, MMAS, SA, PSO.

5.2 Hướng nghiên cứu mở rộng

Từ các nghiên cứu và các kết quả đạt được, luận án đề nghị một số vấn đề và các hướng nghiên cứu tiếp theo như sau:

- Vấn đề 1: Luận án có thể tiếp tục nghiên cứu vấn đề xác định nguồn gốc gây ra lỗi. Trong Chương 3 chỉ mới khảo sát đến việc phân tích thông số hiệu năng của vật lý liên quan đến lỗi. Đây có thể là tiền đề để tiếp tục nghiên cứu vấn đề xác định nguồn gốc gây ra lỗi, từ đó có được mô hình phân tích lỗi máy chủ vật lý hoàn chỉnh và hiệu quả.
- Vấn đề 2: Luận án có thể tiếp tục nghiên cứu vấn đề xác định máy chủ vật lý tối ưu cho di trú máy ảo dựa trên thông số của hạ tầng mạng. Luận án chỉ dừng lại việc xác định máy chủ vật lý cho di trú máy ảo dựa trên thông số hiệu năng. Việc kết hợp giữa thông số hiệu năng với thông số của hạ tầng mạng sẽ giúp cho việc xác định máy chủ vật lý phù hợp, hiệu quả cho việc di trú máy ảo.
- Vấn đề 3: Luận án có thể tiếp tục đánh giá mô hình xây dựng chiến lược di trú máy ảo được đề xuất trong Chương 4 với các mô hình hiện có khác để có kết quả đánh giá toàn diện. Các tham số trong Công thức (4.29) ảnh hưởng đến quá trình học tăng cường gồm hệ số học η , hệ số chiết khấu γ , hệ số khám phá/khai thác ϵ của thuật toán điều khiển di trú máy ảo V2PFQL. Gần đây, việc nghiên cứu chiến lược khám phá/khai thác các thông số này được tiếp cận bằng các phương pháp học máy như Deep Learning.