

TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN
KHOA CÔNG NGHỆ THÔNG TIN

Lê Thị Thùy Dương -
Nguyễn Dương Tuấn Phương

PHÂN TÍCH HIỆU NĂNG QUY TRÌNH
TỪ DỮ LIỆU BẰNG PHƯƠNG PHÁP
XEM XÉT NHIỀU ĐỐI TƯỢNG

KHÓA LUẬN TỐT NGHIỆP CỬ NHÂN
CHƯƠNG TRÌNH CHÍNH QUY

Tp. Hồ Chí Minh, tháng 06/2024

TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN
KHOA CÔNG NGHỆ THÔNG TIN

Lê Thị Thùy Dương - 20120063
Nguyễn Dương Tuấn Phương - 20120166

PHÂN TÍCH HIỆU NĂNG QUY TRÌNH
TỪ DỮ LIỆU BẰNG PHƯƠNG PHÁP
XEM XÉT NHIỀU ĐỐI TƯỢNG

KHÓA LUẬN TỐT NGHIỆP CỬ NHÂN
CHƯƠNG TRÌNH CHÍNH QUY

GIÁO VIÊN HƯỚNG DẪN

ThS. Trần Trung Kiên

TS. Nguyễn Ngọc Thảo

Tp. Hồ Chí Minh, tháng 06/2024

Nhận xét hướng dẫn

Nhận xét phản biện

Lời cảm ơn

Lời đầu tiên cho phép chúng em gửi lời tri ân và lời cảm ơn sâu sắc đến thầy Trần trung Kiên và cô Nguyễn Ngọc Thảo đã tận tình chỉ bảo, định hướng và hướng dẫn cho chúng em trong suốt quá trình thực hiện đề tài. Cảm ơn những lời nhắc nhở chân thành và sự kiên nhẫn của thầy và cô dành cho chúng em trong suốt thời gian vừa qua, chúng em rất biết ơn vì điều đó, nhờ những góp ý của thầy và cô giúp chúng em hoàn thành tốt hơn, kỹ càng hơn trong việc thực hiện khóa luận tốt nghiệp lần này.

Chúng em cũng xin gửi lời cảm ơn chân thành đến quý thầy cô trường Đại học Khoa học Tự nhiên, ĐHQG-HCM đặc biệt là quý thầy cô Khoa Công nghệ Thông tin đã tận tình giảng dạy chúng em trong thời gian học tập tại trường, dạy cho chúng em những kiến thức quý giá không những là nền tảng cho việc nghiên cứu khóa luận tốt nghiệp mà còn là hành trang cho bản thân sau này.

Lời cuối cùng, chúng em xin kính chúc quý thầy/cô dồi dào sức khỏe, luôn bình an và hạnh phúc trong cuộc sống.

Chúng em xin chân thành cảm ơn!

TP. Hồ Chí Minh, ngày 30 tháng 6 năm 2024

Nhóm sinh viên thực hiện

Lê Thị Thùy Dương

Nguyễn Dương Tuấn Phương



ĐỀ CƯƠNG KHÓA LUẬN TỐT NGHIỆP

**PHÂN TÍCH HIỆU NĂNG QUY TRÌNH TỪ
DỮ LIỆU BẰNG PHƯƠNG PHÁP XEM XÉT
NHIỀU ĐỐI TƯỢNG**

*(Analyzing process performance from data using object-centric
method)*

1 THÔNG TIN CHUNG

Người hướng dẫn:

- ThS. Trần Trung Kiên
- TS. Nguyễn Ngọc Thảo (Khoa Công nghệ Thông tin)

Nhóm sinh viên thực hiện:

1. Lê Thị Thùy Dương (MSSV: 20120063)
2. Nguyễn Dương Tuấn Phương (MSSV: 20120166)

Loại đề tài: Nghiên cứu.

Thời gian thực hiện: Từ 01/2024 đến 06/2024.

2 NỘI DUNG THỰC HIỆN

2.1 Giới thiệu về đề tài

Hiện nay, khai thác quy trình từ dữ liệu đang trở thành một xu hướng phổ biến trong các doanh nghiệp, được áp dụng nhằm mục đích tăng cường chất lượng của hoạt động kinh doanh. Và điều này đã dẫn đến sự xuất hiện của bài toán có ý nghĩa quan trọng là bài toán phân tích hiệu năng của quy trình nhằm tối ưu hóa quy trình.

Bài toán phân tích hiệu năng quy trình từ dữ liệu được phát biểu như sau:

- Cho đầu vào là dữ liệu log của một quy trình cần phân tích hiệu năng.
- Yêu cầu: Mô hình hóa được quy trình từ dữ liệu đầu vào, tính toán các chỉ số hiệu năng (ví dụ như thời gian chờ trung bình, thời gian thực thi trung bình, ...) cho từng bước trong quy trình và hiển thị các chỉ số này cùng với mô hình quy trình tương ứng.

Giải quyết được bài toán này sẽ đem đến một ứng dụng tiềm năng là hỗ trợ các doanh nghiệp, nhà đầu tư, và người quản lý trong việc có thể hiểu rõ hơn về quy trình hoạt động nghiệp vụ của họ. Đồng thời, họ có thể phát hiện các điểm thiếu hiệu quả trong quy trình vận hành, cũng như nhận biết các biến thể sai lệch so với thiết kế ban đầu. Những phát hiện này có thể giúp họ điều chỉnh lại quy trình làm việc, phân bổ tài nguyên hiệu quả hơn. Từ đó giảm chi phí, tối ưu hóa hoạt động, nâng cao chất lượng, tăng sự cạnh tranh và thích nghi nhanh chóng với sự biến đổi trong môi trường kinh doanh.

Tuy nhiên, bài toán cũng tồn tại một số khó khăn và thách thức nhất định. Đầu tiên là yêu cầu về chất lượng của dữ liệu, điều này đóng vai trò quan trọng trong khai thác quy trình, đặc biệt là với bài toán phân tích hiệu năng. Để đảm bảo hiệu quả và các kết quả phân tích hiệu năng chính xác thì đòi hỏi cần có nguồn dữ liệu đầy đủ và chất lượng cao. Sự không chính xác hay thiếu sót trong dữ liệu có

thể dẫn đến kết quả sai lệch về hiệu năng của quy trình, từ đó tạo ra những nhận thức sai lệch từ kết quả đó. Ngoài ra, các phương pháp tiếp cận truyền thống để mô hình hóa quy trình thường không thể hiện đầy đủ những gì thực tế xảy ra bên trong của quy trình. Với các quy trình phức tạp với nhiều hoạt động và đối tượng phức tạp, khó có thể thu được những hiểu biết hữu ích từ các mô hình quy trình sử dụng phương pháp truyền thống.

Trong thời gian gần đây, một phương pháp tiếp cận đạt được kết quả tốt trong bài toán phân tích hiệu năng quy trình đó là sử dụng phương pháp xem xét nhiều đối tượng (Object-Centric). Đây sẽ là phương pháp tiếp cận mà chúng em chọn để tìm hiểu cho đề tài.

2.2 Mục tiêu đề tài

- Hiểu rõ bối cảnh nghiên cứu về bài toán phân tích hiệu năng quy trình từ dữ liệu (hiện nay, đã xuất hiện những phương pháp tiếp cận nào để giải quyết vấn đề này? Mỗi phương pháp có ý tưởng, kết quả, cũng như ưu và nhược điểm như thế nào?). Từ đó chọn ra một phương pháp tốt (ứng với một bài báo uy tín), có tiềm năng phát triển trong tương lai và cũng khả thi để hoàn thành trong thời gian của khóa luận để tìm hiểu sâu.
- Hiểu rõ về lý thuyết của phương pháp tiếp cận đã lựa chọn (trên cơ sở hiểu rõ lý thuyết nền tảng về khai thác quy trình).
- Cài đặt lại thuật toán theo phương pháp tiếp cận đã chọn để ra được các kết quả trong bài báo tương ứng; có thể tiến hành thêm một số thí nghiệm ngoài bài báo để thấy rõ hơn về ưu/nhược điểm của thuật toán theo phương pháp tiếp cận được sử dụng.
- Trên cơ sở đã hiểu rõ thuật toán của phương pháp tiếp cận, nếu còn thời gian thì có thể xem xét các cải tiến có thể có.
- Rèn luyện được các kỹ năng: Tư duy logic, lên kế hoạch, làm việc nhóm, trình

bày, ...

2.3 Phạm vi của đề tài

- Để giải quyết bài toán phân tích hiệu năng quy trình, việc cần thiết nhất là sử dụng dữ liệu thực tế như một nguồn thông tin cơ sở. Trong nghiên cứu này, chúng em sẽ sử dụng đầu vào là dữ liệu log từ các quy trình thực tế, thường được sử dụng phổ biến trong các bài toán về quy trình. Cụ thể là bộ dữ liệu về quy trình đăng ký khoản vay của một viện Tài chính Hà Lan, được cung cấp bởi Hội thảo Quốc tế về Business Process Intelligence (BPI) lần thứ 13 diễn ra vào năm 2017.
- Đề tài sẽ tập trung vào việc hiểu và tái cài đặt thuật toán từ một bài báo uy tín. Bên cạnh đó, chúng em cũng có kế hoạch thực hiện các thí nghiệm bổ sung ngoài phạm vi của bài báo để có cái nhìn rõ ràng hơn về ưu/nhược điểm của thuật toán. Chúng em quyết định giới hạn phạm vi của đề tài vì hai lý do chính: (i) việc hiểu thuật toán và các kiến thức nền tảng cũng như tái cài đặt thuật toán mất rất nhiều thời gian, và (ii) chúng em nhận thấy rằng, chỉ thông qua việc hiểu rõ thuật toán và các kiến thức nền tảng, chúng em mới có thể đạt được các cải tiến thực sự trong tương lai và áp dụng thuật toán cho các bài toán khác. Tất nhiên, trong quá trình làm khóa luận, nếu có đủ thời gian, chúng em sẽ xem xét và thử nghiệm các cải tiến; tuy nhiên, điều này không phải là mục tiêu chính.

2.4 Cách tiếp cận dự kiến

Dưới đây chúng em sẽ trình bày một số phương pháp để giải quyết bài toán phân tích hiệu năng quy trình mà chúng em đã tìm hiểu được cho đến thời điểm hiện tại, cũng như là phương pháp mà chúng em dự kiến sẽ chọn để tập trung tìm hiểu sâu.

- Một nghiên cứu đáng chú ý trong nhóm các phương pháp truyền thống là “Performance analysis of business processes through process mining” [1]. Nghiên cứu này bắt đầu bằng việc tạo ra một mô hình quy trình dạng Petri Net từ dữ liệu log. Tiếp theo, các sự kiện trong dữ liệu log được phát lại (replay) trên mô hình quy trình này và sau đó các chỉ số hiệu năng được tính toán và ghi lại. Tuy nhiên, điểm yếu của nghiên cứu này và của nhóm các phương pháp truyền thống là giả định rằng mỗi sự kiện trong dữ liệu log chỉ thuộc về một trường hợp duy nhất của việc thực hiện quy trình cho một đối tượng [2]. Và trong thực tế, một sự kiện có thể thuộc về nhiều trường hợp thực hiện quy trình của nhiều đối tượng khác nhau. Điều này đòi hỏi chúng ta sẽ cần phải tiền xử lý dữ liệu log trước tiên, để làm cho mỗi sự kiện chỉ thuộc về đúng một trường hợp thực hiện quy trình của một đối tượng khi muốn áp dụng các phương pháp truyền thống. Việc này có thể sẽ làm cho mô hình quy trình không phản ánh được thực tế một cách chính xác, và do đó có thể sẽ làm cho kết quả tính toán các chỉ số hiệu năng không được chính xác. Ngoài ra, các chỉ số hiệu năng liên quan đến việc tương tác giữa các đối tượng cũng không được xem xét tới.
- Nhận thấy nhược điểm của các phương pháp truyền thống, bài báo “OPerA: Object-Centric Performance Analysis” [3] đã đề xuất một phương pháp phân tích hiệu năng quy trình từ dữ liệu mà có xem xét nhiều đối tượng. Cách làm của phương pháp này tương tự như bài báo [1], nhưng thay vì sử dụng mô hình quy trình dạng Petri Net, phương pháp sẽ sử dụng Object-Centric Petri Net [4]. Bằng cách này, mô hình quy trình có khả năng phản ánh thực tế chính xác hơn, từ đó cho ra kết quả tính toán các chỉ số hiệu năng của quy trình sẽ chính xác hơn. Ngoài ra, phương pháp [3] cũng đề xuất thêm các chỉ số hiệu năng liên quan đến sự tương tác của các đối tượng.

Với những gì đã trình bày ở trên, chúng em quyết định sẽ tập trung tìm hiểu sâu

phương pháp xem xét nhiều đối tượng được đề xuất bởi [3] để giải quyết bài toán phân tích hiệu năng quy trình từ dữ liệu.

2.5 Kết quả dự kiến của đề tài

- Tổng hợp đầy đủ, phân tích rõ ràng, cụ thể về các phương pháp phân tích hiệu năng quy trình.
- Cài đặt lại được từ đầu phương pháp được đề xuất trong bài báo [3].
- Có được các kết quả thí nghiệm cho thấy việc tự cài đặt cho ra được các kết quả tương tự như bài báo [3].
- Có được các kết quả thí nghiệm ngoài bài báo [3] để thấy rõ hơn về ưu/nhược điểm của phương pháp.
- Nếu có thời gian, có thể nghiên cứu cải tiến phương pháp và có được các kết quả thí nghiệm tương ứng.

2.6 Kế hoạch thực hiện

Từ ngày	Đến ngày	Nội dung thực hiện	Phân công
01/01/2024	07/01/2024	Thống nhất ý kiến, xác định các mục tiêu, yêu cầu và lên kế hoạch thực hiện khóa luận.	Dương, Phương
08/01/2024	22/01/2024	Tìm hiểu về tình hình nghiên cứu của bài toán phân tích hiệu năng quy trình.	Dương, Phương
23/01/2024	30/01/2024	Tìm hiểu về các phương pháp tiếp cận xem xét nhiều đối tượng cho bài toán phân tích hiệu năng quy trình.	Dương, Phương

31/01/2024	06/02/2024	Phân tích ưu/khuyết điểm của phương pháp tiếp cận và kết luận.	Dương, Phương
07/02/2023	21/02/2024	Tìm hiểu về các bài báo có liên quan đến bài toán và phương pháp tiếp cận đã chọn, chọn được bài báo chính để tiến hành tìm hiểu và nghiên cứu.	Dương, Phương
22/02/2024	29/02/2024	Đọc hiểu bài báo chính và tìm hiểu thêm các bài báo mới có liên quan đến bài toán.	Dương, Phương
01/03/2024	14/03/2024	Đọc và hiểu sâu hơn về bài báo chính và đọc hiểu thêm các bài báo đã chọn lọc.	Dương, Phương
15/03/2024	31/03/2024	Tìm hiểu về thuật toán được sử dụng cho bài báo và các công trình liên quan.	Dương, Phương
01/04/2024	25/04/2024	Dựa trên ý tưởng các công trình hiện có và kiến thức đã tìm hiểu được, xây dựng và cài đặt lại thuật toán giải quyết bài toán của khóa luận.	Dương, Phương
25/04/2024	07/05/2024	So sánh, nhận xét, đánh giá kết quả và tinh chỉnh thuật toán cho phù hợp.	Dương, Phương
08/05/2024	22/05/2024	Bổ sung và hoàn thành viết cuốn luận.	Dương, Phương

23/05/2024	31/05/2024	Hoàn tất viết cuốn luận, tiến hành làm slide thuyết trình cho khóa luận.	Dương, Phương
01/06/2024	07/06/2024	Kiểm tra lại cuốn luận, mã nguồn và các tài nguyên liên quan.	Dương, Phương

Tài liệu

- [1] P. T. G. Hornix, T.-I. A. J. M. M. Weijters, W.-D. P. M. E. D. Bra, W.-A. M. Voorhoeve, and P. H. Stramproy, *Performance Analysis of Business Processes through Process Mining*. Master's thesis of P.T.G Hornix at Eindhoven University of Technology, 2007.
- [2] J. N. Adams, D. Schuster, S. Schmitz, G. Schuh, and W. M. P. van der Aalst, *Defining Cases and Variants for Object-Centric Event Data*. International Conference on Process Mining, 2022.
- [3] G. Park, J. N. Adams, and W. M. P. van der Aalst, *OPerA: Object-Centric Performance Analysis*. Springer International Publishing, 2022.
- [4] W. M. P. van der Aalst and A. Berti, *Discovering Object-Centric Petri Nets*. Fundamenta Informaticae, 2020.

XÁC NHẬN
CỦA NGƯỜI HƯỚNG DẪN
(Ký và ghi rõ họ tên)



Trần Trung Kiên



Nguyễn Ngọc Thảo

TP. Hồ Chí Minh, ngày 4 tháng 4 năm 2024
NHÓM SINH VIÊN THỰC HIỆN
(Ký và ghi rõ họ tên)



Lê Thị Thùy Dương



Nguyễn Dương Tuấn Phương

Mục lục

1	Giới thiệu	1
1.1	Phát biểu bài toán	1
1.2	Giới thiệu các phương pháp giải quyết bài toán và phương pháp mà khóa luận tập trung tìm hiểu	4
2	Kiến thức nền tảng	8
2.1	Dữ liệu event log	8
2.1.1	Dữ liệu event log theo chuẩn OCEL (Object-Centric Event Log)	8
2.1.2	Tiền xử lý dữ liệu event log để có thể áp dụng phương pháp truyền thống	11
2.2	Phân tích hiệu năng quy trình từ dữ liệu event log bằng phương pháp truyền thống	14
2.2.1	Bước 1 - Tìm mô hình quy trình truyền thống từ dữ liệu event log	14
2.2.2	Bước 2 - Tính các kết quả trung gian liên quan đến chỉ số hiệu năng từ mô hình quy trình và dữ liệu event log	21
2.2.3	Bước 3 - Tính các chỉ số hiệu năng từ các kết quả trung gian và mô hình quy trình	22
3	Phân tích hiệu năng quy trình từ dữ liệu event log bằng phương pháp xem xét nhiều đối tượng	25

3.1	Vấn đề gặp phải của phương pháp truyền thống	26
3.2	Ý tưởng của phương pháp xem xét nhiều đối tượng để giải quyết vấn đề gặp phải của phương pháp truyền thống . . .	29
3.3	Chi tiết về các bước thực hiện của phương pháp xem xét nhiều đối tượng	31
3.3.1	Bước 1 - Tìm mô hình quy trình có xem xét nhiều đối tượng từ dữ liệu event log	31
3.3.2	Bước 2 - Tính các kết quả trung gian liên quan đến chỉ số hiệu năng từ mô hình quy trình và dữ liệu event log	34
3.3.3	Bước 3 - Tính các chỉ số hiệu năng từ các kết quả trung gian và mô hình quy trình	37
4	Thí nghiệm	43
4.1	Các thiết lập thí nghiệm	43
4.1.1	Tập dữ liệu	43
4.1.2	Rút trích dữ liệu	45
4.2	Thí nghiệm 1 - So sánh kết quả cài đặt của khóa luận với bài báo gốc	46
4.3	Thí nghiệm 2 - Cải thiện kết quả của phương pháp xem xét nhiều đối tượng ở thí nghiệm 1	51
5	Tổng kết và hướng phát triển	54
5.1	Tổng kết	54
5.2	Hướng phát triển	56
	Tài liệu tham khảo	57

Danh sách hình

1.1	Ví dụ về công cụ ProM trong khai thác quy trình.	2
2.1	Ví dụ về dữ liệu event log theo quy chuẩn của OCEL. Mỗi hàng (trừ tiêu đề) sẽ đại diện cho một sự kiện (event). . .	9
2.2	Ví dụ về một mô hình quy trình Petri Nets mô tả quá trình chuyển đổi đơn giản của đèn giao thông.	16
2.3	Ví dụ minh họa về quá trình kích hoạt, tiêu thụ và sản xuất mã thông báo (token) trên Petri Net.	17
2.4	Ví dụ minh họa mô hình quy trình Petri Net có thành phần chuyển tiếp im lặng (silent transitions).	18
2.5	Ví dụ về thực hiện phát lại dữ liệu event log trên Petri Net.	23
3.1	Mô hình quy trình xét nghiệm máu của đối tượng <i>Test</i> và <i>Sample</i>	28
3.2	Tổng quan phương pháp xem xét nhiều đối tượng.	30
3.3	Ví dụ về phát lại (replay) dữ liệu event log lên mô hình quy trình OCPN.	36
3.4	Tổng hợp các độ đo hiệu năng.	39
3.5	Ví dụ minh họa các độ đo hiệu năng theo thời gian của sự kiện <i>Conduct Test</i> trong một quy trình xét nghiệm máu. .	42
4.1	Kết quả phân tích hiệu năng quy trình dựa trên cài đặt khóa luận khi chạy thử nghiệm trên tập dữ liệu “BPI 2017” . . .	52

4.2	Kết quả phân tích hiệu năng quy trình của tác giả tập dữ liệu “BPI 2017” khi áp dụng phương pháp truyền thống . .	53
4.3	Kết quả phân tích hiệu năng quy trình của tác giả tập dữ liệu “BPI 2017” khi áp dụng phương pháp xem xét nhiều đối tượng	53

Danh sách bảng

1.1	Dữ liệu event log của một quy trình xét nghiệm máu. . . .	5
2.1	Dữ liệu event log minh họa	13
2.2	Dữ liệu event log sau khi “phân tách” cho loại đối tượng “Order”	14
2.3	Dữ liệu event log sau khi “phân tách” cho loại đối tượng “Item”	14
3.1	Dữ liệu event log của quy trình xét nghiệm máu đối với đối tượng Test.	27
3.2	Dữ liệu event log của quy trình xét nghiệm máu của đối tượng Sample.	27

Tóm tắt

Trong quy trình vận hành của một công ty hay tổ chức ngày nay, việc nhận biết những vấn đề trong quy trình và tối ưu hóa quy trình là việc làm không thể thiếu. Giảm thiểu thời gian thừa trong quy trình là một mục tiêu quan trọng để tối ưu hóa hiệu suất và tăng lợi nhuận cho công ty và doanh nghiệp. Vì vậy trong khóa luận này, nhóm chúng em sẽ trình bày về phương pháp “xem xét nhiều đối tượng” (Object-Centric Petri Net) để giải quyết bài toán phân tích hiệu năng quy trình từ dữ liệu event log. Phương pháp này cho thấy sự phù hợp hơn với quy trình có nhiều đối tượng liên quan và giúp tính toán chính xác các chỉ số hiệu năng của quy trình hơn so với phương pháp truyền thống.

Chương 1

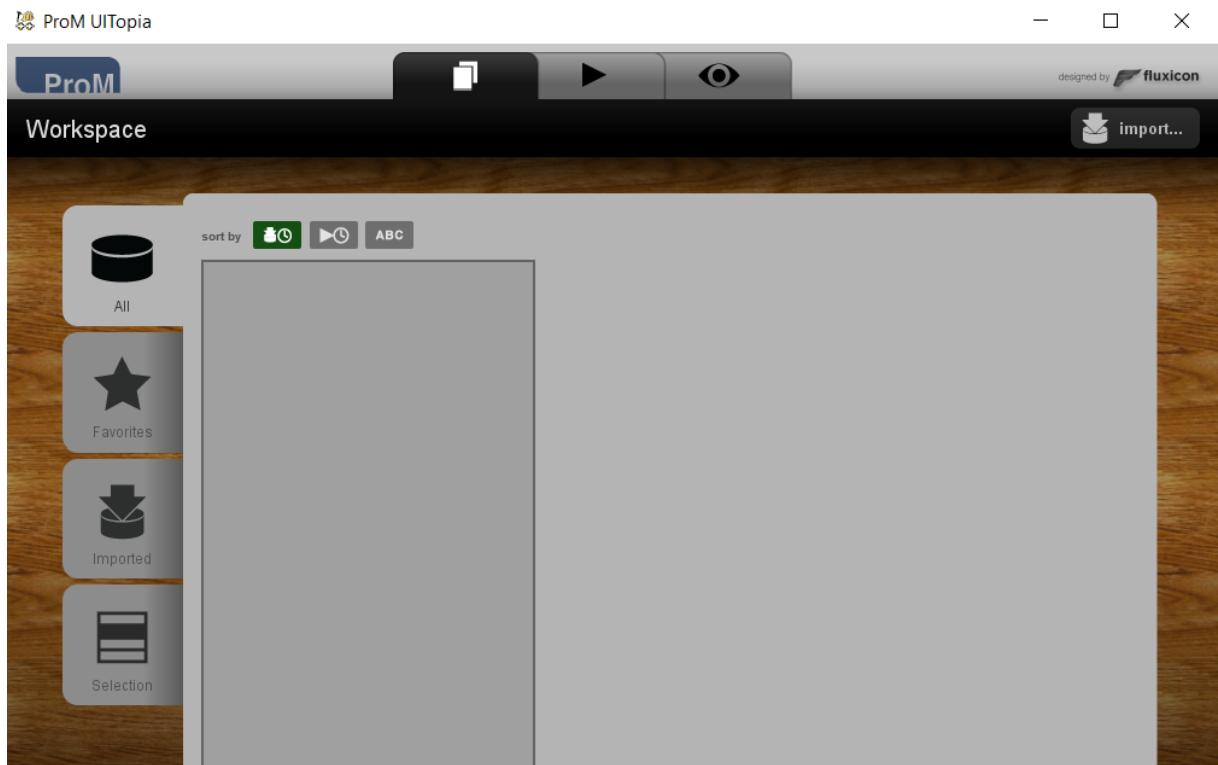
Giới thiệu

Trong chương này, đầu tiên chúng em sẽ phát biểu về bài toán phân tích hiệu năng quy trình từ dữ liệu cũng như ý nghĩa và thách thức của nó đối với cuộc sống hiện nay. Sau đó, chúng em sẽ giới thiệu về các phương pháp đã được đề xuất để giải quyết bài toán và phương pháp mà chúng em chọn để tập trung tìm hiểu. Cuối cùng, chúng em sẽ trình bày về các thí nghiệm đã thực hiện và hướng phát triển của khóa luận.

1.1 Phát biểu bài toán

Hiện nay, trong môi trường kinh doanh ngày càng cạnh tranh gay gắt, việc áp dụng phân tích dữ liệu để tối ưu hoá quy trình đã trở thành một xu hướng thiết thực và phổ biến. Đây không chỉ là nỗ lực nâng cao chất lượng hoạt động kinh doanh mà còn là yếu tố quyết định giúp các doanh nghiệp vượt qua các thách thức, đáp ứng nhanh chóng với sự biến đổi của thị trường và nhu cầu ngày càng cao của khách hàng. Bài toán phân tích hiệu năng quy trình không chỉ đơn thuần là một vấn đề kỹ thuật, mà còn là một phương tiện quan trọng để tối ưu hóa các quy trình trong tổ chức. Việc đánh giá và cải thiện hiệu suất của các quy trình mang lại lợi ích rất lớn, từ việc giảm thiểu lãng phí tài nguyên đến tăng cường năng suất lao động và cải thiện trải nghiệm của khách hàng. Điều này không chỉ giúp

doanh nghiệp tiết kiệm chi phí mà còn nâng cao khả năng cạnh tranh và sự bền vững trên thị trường hiện nay. Trong thực tế, đã có nhiều phương pháp để phân tích hiệu năng quy trình dựa trên dữ liệu event log, một trong số đó chính là khai thác quy trình (Process Mining). Đây là phương pháp phổ biến nhất để phân tích hiệu năng quy trình từ dữ liệu event log. Process Mining sử dụng các kỹ thuật và công cụ để tự động phân tích và khai thác dữ liệu từ event log để hiểu rõ các quy trình kinh doanh, tìm ra các mẫu, các quy luật và cải thiện hiệu năng quy trình. Các công cụ như Disco, Celonis, Fluxicon và ProM (Hình 1.1) đều là các phần mềm chuyên dụng trong lĩnh vực khai thác quy trình (Process Mining). Chúng được sử dụng để phân tích và cải thiện hiệu năng của quy trình trong các tổ chức và doanh nghiệp.



Hình 1.1: Ví dụ về công cụ ProM trong khai thác quy trình.

Bài toán phân tích hiệu năng quy trình từ dữ liệu được phát biểu như sau:

- Cho đầu vào là dữ liệu event log của một quy trình cần phân tích

hiệu năng.

- Yêu cầu: Mô hình hóa được quy trình từ dữ liệu đầu vào, tính toán các chỉ số hiệu năng (ví dụ như thời gian chờ trung bình, thời gian thực thi trung bình, ...) cho từng bước trong quy trình và hiển thị các chỉ số này cùng với mô hình quy trình tương ứng.

Giải quyết được bài toán này sẽ đem đến một ứng dụng tiềm năng là hỗ trợ các doanh nghiệp, nhà đầu tư, và người quản lý trong việc có thể hiểu rõ hơn về quy trình hoạt động nghiệp vụ của họ. Đồng thời, họ có thể phát hiện các điểm thiếu hiệu quả trong quy trình vận hành, cũng như nhận biết các biến thể sai lệch so với thiết kế ban đầu. Những phát hiện này có thể giúp họ điều chỉnh lại quy trình làm việc, phân bổ tài nguyên hiệu quả hơn. Từ đó giảm chi phí, tối ưu hóa hoạt động, nâng cao chất lượng, tăng sự cạnh tranh và thích nghi nhanh chóng với sự biến đổi trong môi trường kinh doanh.

Tuy nhiên, bài toán cũng tồn tại một số khó khăn và thách thức nhất định. Đầu tiên là yêu cầu về chất lượng của dữ liệu, điều này đóng vai trò quan trọng trong khai thác quy trình, đặc biệt là với bài toán phân tích hiệu năng. Để đảm bảo hiệu quả và các kết quả phân tích hiệu năng chính xác thì đòi hỏi cần có nguồn dữ liệu đầy đủ và chất lượng cao. Sự không chính xác hay thiếu sót trong dữ liệu có thể dẫn đến kết quả sai lệch về hiệu năng của quy trình, từ đó tạo ra những nhận thức sai lệch từ kết quả đó. Ngoài ra, các phương pháp tiếp cận truyền thống để mô hình hóa quy trình thường không thể hiện đầy đủ những gì thực tế xảy ra bên trong của quy trình. Với các quy trình phức tạp với nhiều hoạt động và đối tượng phức tạp, khó có thể thu được những hiểu biết hữu ích từ các mô hình quy trình sử dụng phương pháp truyền thống.

Trong thời gian gần đây, một phương pháp tiếp cận đạt được kết quả tốt trong bài toán phân tích hiệu năng quy trình đó là sử dụng phương pháp xem xét nhiều đối tượng (Object-Centric). Đây sẽ là phương pháp tiếp cận mà chúng em chọn để tìm hiểu cho đề tài.

1.2 Giới thiệu các phương pháp giải quyết bài toán và phương pháp mà khóa luận tập trung tìm hiểu

Để giải quyết bài toán phân tích hiệu năng quy trình từ dữ liệu event log, chúng ta có thể sử dụng phương pháp truyền thống như “Performance analysis of business processes through process mining” [5]. Nghiên cứu này bắt đầu bằng việc tạo ra một mô hình quy trình dạng Petri Net (chương 2) từ dữ liệu event log. Tiếp theo, các sự kiện trong dữ liệu event log được phát lại (replay) trên mô hình quy trình này và sau đó các chỉ số hiệu năng được tính toán và ghi lại. Tuy nhiên, điểm yếu của nghiên cứu này và của nhóm các phương pháp truyền thống là giả định rằng mỗi sự kiện trong dữ liệu event log chỉ thuộc về một trường hợp duy nhất của việc thực hiện quy trình cho một đối tượng [2]. Và trong thực tế, một sự kiện có thể thuộc về nhiều trường hợp thực hiện quy trình của nhiều đối tượng khác nhau. Trong trường hợp này, đòi hỏi chúng ta sẽ cần phải tiền xử lý dữ liệu event log trước tiên, để làm cho mỗi sự kiện chỉ thuộc về đúng một trường hợp thực hiện quy trình của một đối tượng khi muốn áp dụng các phương pháp truyền thống. Việc này có thể sẽ làm cho mô hình quy trình không phản ánh được thực tế một cách chính xác, và do đó có thể sẽ làm cho kết quả tính toán các chỉ số hiệu năng không được chính xác. Ngoài ra, các chỉ số hiệu năng liên quan đến việc tương tác giữa các đối tượng cũng không được xem xét tới.

Bảng 1.1 là một ví dụ về dữ liệu event log cho đầu vào của bài toán phân tích hiệu năng quy trình. Đây là dữ liệu event log của một mô hình quy trình xét nghiệm máu đơn giản. Khi một người cần xét nghiệm máu, họ cần trải qua các bước như sau:

- Chuẩn bị xét nghiệm (Prepare test): đây là bước để xác định được những thông tin cơ bản cho quy trình xét nghiệm máu (ví dụ như thông tin của bệnh nhân, các loại xét nghiệm,...).

- Lấy mẫu (Take sample): quá trình này cần lấy 2 mẫu máu từ bệnh nhân. Trong một số xét nghiệm, yêu cầu có nhiều mẫu máu (ví dụ như đo nồng độ thuốc có trong máu), dự phòng cho trường hợp bị thất lạc hoặc có thể phải thực hiện xét nghiệm lại khi có kết quả bất thường,... Ngoài ra, khi bệnh nhân cần kiểm tra nhiều loại xét nghiệm cùng lúc cũng cần lấy nhiều mẫu máu hơn.
- Tiến hành xét nghiệm (Conduct test): tiến hành xử lý các yêu cầu xét nghiệm dành cho bệnh nhân. Đây là sự kiện có sự liên quan của nhiều đối tượng (Test và Sample).
- Công bố kết quả (Publish test): tổng hợp và trả về kết quả xét nghiệm.
- Chuyển mẫu (Transfer sample) là bước đưa các mẫu máu sang bộ phận khác để tiếp tục xử lý, sau khi hoàn thành xét nghiệm.
- Xử lý mẫu (Clear sample): các mẫu máu có thể được lưu trữ lại cho các mục đích khác như kiểm tra lại, nghiên cứu hoặc giám định. Nếu không cần thiết, mẫu máu cũng có thể được tiêu hủy để đảm bảo an toàn và tuân thủ các quy định về vệ sinh và môi trường.

Event ID	Activity	Test	Sample	Time start	Time end
e1	Prepare test	T_1		0:00:00	0:00:15
e2	Take sample		S_1	0:02:00	0:02:30
e3	Take sample		S_2	0:02:30	0:02:45
e4	Conduct test	T_1	S_1, S_2	0:03:00	0:04:00
e5	Publish test	T_1		0:04:30	0:05:00
e6	Transfer sample		S_1, S_2	0:04:45	0:05:30
e7	Clear sample		S_1	0:05:30	0:05:45
e8	Clear sample		S_2	0:05:45	0:06:40

Bảng 1.1: Dữ liệu event log của một quy trình xét nghiệm máu.

Với phương pháp truyền thống, cách làm của phương pháp này chính là đưa dữ liệu đầu vào về đúng với giả định ban đầu (tiền xử lý dữ liệu).

Đầu tiên, những sự kiện có liên quan đến nhiều đối tượng như Conduct test sẽ được tách thành những sự kiện chỉ liên quan đến một đối tượng.

Nhận thấy nhược điểm của các phương pháp truyền thống, bài báo “OPerA: Object-Centric Performance Analysis” [6] đã đề xuất một phương pháp phân tích hiệu năng quy trình từ dữ liệu mà có xem xét nhiều đối tượng. Cách làm của phương pháp này tương tự như bài báo [5], nhưng thay vì sử dụng mô hình quy trình dạng Petri Net, phương pháp sẽ sử dụng Object-Centric Petri Net [1]. Bằng cách này, mô hình quy trình có khả năng phản ánh thực tế chính xác hơn, từ đó cho ra kết quả tính toán các chỉ số hiệu năng của quy trình sẽ chính xác hơn. Ngoài ra, phương pháp [6] cũng đề xuất thêm các chỉ số hiệu năng liên quan đến sự tương tác của các đối tượng. Đây chính là phương pháp tối ưu nhất chúng em đã tìm hiểu được cho đến thời điểm hiện tại, cũng như là phương pháp mà chúng em dự kiến sẽ chọn để tập trung tìm hiểu sâu nhằm giải quyết yêu cầu bài toán phân tích hiệu năng quy trình từ dữ liệu.

Đối với các phần còn lại của khoá luận chúng em sẽ trình bày như sau:

- Chương 2: Trình bày những kiến thức nền tảng về dữ liệu event log và chi tiết các bước phân tích hiệu năng quy trình từ dữ liệu event log bằng phương pháp truyền thống.
- Chương 3: Trình bày về phương pháp xem xét nhiều đối tượng để giải quyết bài toán phân tích hiệu năng quy trình từ dữ liệu event log; đây là phần chính của khoá luận. Trong phần này gồm có 3 phần nhỏ:
 - Vấn đề gặp phải của phương pháp truyền thống.
 - Ý tưởng của phương pháp xem xét nhiều đối tượng để giải quyết vấn đề gặp phải của phương pháp truyền thống.
 - Chi tiết về các bước thực hiện của phương pháp xem xét nhiều đối tượng.

- Chương 4: Trình bày các thí nghiệm để đánh giá mức độ hiệu quả của phương pháp xem xét nhiều đối tượng trong việc phân tích hiệu năng quy trình ở các bộ dữ liệu thực tế.
- Cuối cùng, kết luận và hướng phát triển được trình bày ở chương 5.

Chương 2

Kiến thức nền tảng

Chương này sẽ trình bày những kiến thức nền tảng để có thể hiểu rõ và sâu sắc hơn về phương pháp chính mà khóa luận tập trung tìm hiểu (sẽ được trình bày ở chương 3). Đầu tiên, chúng em trình bày về dữ liệu event log, bao gồm: chuẩn OCEL của dữ liệu event log, và cách tiền xử lý dữ liệu event log để có thể áp dụng phương pháp phân tích truyền thống. Sau đó, chúng em trình bày về phương pháp truyền thống để giải quyết bài toán phân tích hiệu năng quy trình; phương pháp này sẽ có 3 bước thực hiện: (1) tìm mô hình quy trình truyền thống từ dữ liệu event log, (2) tính các kết quả trung gian liên quan đến chỉ số hiệu năng từ mô hình quy trình và dữ liệu event log, (3) tính các chỉ số hiệu năng từ các kết quả trung gian và mô hình quy trình.

2.1 Dữ liệu event log

2.1.1 Dữ liệu event log theo chuẩn OCEL (Object-Centric Event Log)

Theo thuật ngữ điện toán, sự kiện (event) là bất kỳ hành động hoặc sự xuất hiện quan trọng nào mà hệ thống phần mềm có thể nhận ra và ghi nhận. Mỗi hành động thực hiện trong hệ thống sẽ để lại một dấu vết kỹ

thuật số trên hệ thống lưu trữ, dấu vết này được gọi là event log và tập hợp các sự kiện được trích xuất từ cơ sở dữ liệu hỗ trợ việc thực hiện quy trình này sẽ được gọi là dữ liệu nhật ký sự kiện hay dữ liệu event log. Dữ liệu event log bao gồm danh sách các sự kiện được ghi lại theo thứ tự thời gian, cung cấp một bức tranh chi tiết và liên tục về các hoạt động diễn ra trong hệ thống. Quá trình rút trích dữ liệu event log từ cơ sở dữ liệu là một bước quan trọng để thu thập và phân tích thông tin này, giúp theo dõi, giám sát và tối ưu hóa hiệu suất hệ thống, cũng như hỗ trợ việc phát hiện và khắc phục sự cố một cách hiệu quả. Trong bài báo “OPerA: Object-Centric Performance Analysis” [6], nhóm tác giả cũng đã đưa ra một số quy chuẩn liên quan đến OCEL [4] - một cách thể hiện mới của dữ liệu event log trong hệ thống thông tin nhằm đảm bảo tính hợp lý và hiệu quả trong việc phân tích hiệu năng quy trình. Những quy chuẩn này quy định về những thông tin cần có cho dữ liệu event log để có thể áp dụng được cả phương pháp truyền thống (không xem xét nhiều đối tượng) lẫn phương pháp hiện đại (có xem xét nhiều đối tượng).

id	activity	timestamp	item	order	package	prepaid-amount	weight
e_1	<i>place order</i>	2020-07-09 08:20:01.527+01:00	$\{ i_1, i_2 \}$	$\{ o_1 \}$		200.0	
e_2	<i>check availability</i>	2020-07-09 08:21:01.527+01:00	$\{ i_1 \}$				10.0
e_3	<i>load package</i>	2020-07-09 08:22:01.527+01:00			$\{ p_1 \}$		

Hình 2.1: Ví dụ về dữ liệu event log theo quy chuẩn của OCEL. Mỗi hàng (trừ tiêu đề) sẽ đại diện cho một sự kiện (event).

Một event log theo quy chuẩn của OCEL sẽ bao gồm các thành phần chính là sự kiện (event) và đối tượng (object). Các thành phần này thường liên quan đến một quy trình kinh doanh cụ thể, chẳng hạn như quy trình Order-to-Cash (O2C) hoặc Purchase-to-Pay (P2P).

- Sự kiện (Event): Mỗi sự kiện trong OCEL đại diện cho một bản ghi thực thi của một bước trong một quy trình kinh doanh cơ bản. Nói cách khác, sự kiện là một bản ghi, một minh họa của việc thực thi một hoạt động cụ thể xảy ra tại một thời điểm nhất định trong quy

trình.

- Định danh (Identifier): Mỗi sự kiện sẽ được liên kết với một định danh duy nhất, giúp xác định và phân biệt sự kiện này với các sự kiện khác trong dữ liệu event log.
 - Hoạt động (Activity): Đây là hành động cụ thể diễn ra trong quá trình thực thi quy trình. Hoạt động cho biết sự kiện đó đại diện cho bước nào trong quy trình kinh doanh.
 - Dấu thời gian (Timestamp): Dấu thời gian biểu thị thời điểm cụ thể khi sự kiện diễn ra. Nó giúp theo dõi trình tự thời gian của các sự kiện trong quy trình.
- Đối tượng (Object): Đối tượng trong OCEL là các thực thể liên quan đến sự kiện và quy trình kinh doanh. Đối tượng có thể là các yếu tố như đơn hàng, khách hàng, sản phẩm, hóa đơn, v.v. Mỗi đối tượng sẽ có một vai trò cụ thể trong quy trình và có thể liên quan đến nhiều sự kiện khác nhau.

Trong hàng đầu tiên của hình 2.1, với sự kiện (event) có mã định danh (identifier) e_1 được liên kết với hoạt động (activity) là “place order” vào thời điểm “2020-07-09 08:20:01.527+01:00”. Việc thực thi của một sự kiện (event) nó không chỉ liên quan đến các thông tin nêu trên mà sẽ còn bao gồm một tập hợp các đối tượng (object) liên quan đến sự kiện (event) đó. Trong trường hợp của sự kiện (event) này, có ba đối tượng (o_1 , i_1 , và i_2) tham gia vào quá trình thực thi.

Ngoài ra một sự kiện (event) có thể chứa nhiều thuộc tính sự kiện. Mỗi thuộc tính sự kiện đều có tên thuộc tính và giá trị thuộc tính tương ứng. Tên thuộc tính và giá trị thuộc tính này đều liên quan đến các loại thuộc tính cụ thể. Ví dụ, sự kiện (event) e_1 có một thuộc tính “prepaid-amount” (tên thuộc tính), và giá trị của thuộc tính này là “200.0”.

Như đã thể hiện trong hình 2.1, dữ liệu event log theo quy chuẩn OCEL còn tồn tại các mối quan hệ nhiều-nhiều giữa các đối tượng và các sự kiện

với nhau. Nhiều đối tượng có thể tham gia vào một sự kiện, ví dụ ta có thể thấy ở hình 2.1 các đối tượng như o_1 , i_1 , và i_2 đều cùng tham gia vào sự kiện e_1 . Và ngược lại thì một đối tượng cũng có thể tham gia vào nhiều sự kiện, ví dụ ở hình 2.1, đối tượng i_1 đều tham gia vào cả hai sự kiện là e_1 và e_2 .

2.1.2 Tiền xử lý dữ liệu event log để có thể áp dụng phương pháp truyền thống

Dữ liệu event log có thể được xem là điểm khởi đầu cơ bản để áp dụng các kỹ thuật khai thác quy trình. Với dữ liệu event log truyền thống, các sự kiện (event) được đặc trưng bởi một định danh trường hợp, chỉ đến một trường hợp duy nhất của quy trình. Mỗi sự kiện cũng bao gồm một hoạt động, một dấu thời gian cho biết thời điểm sự kiện diễn ra, và các thuộc tính thông tin bổ sung khác. Tiêu chuẩn XES, dựa trên XML, được thiết lập cho các dữ liệu event log truyền thống này. Tiêu chuẩn XES yêu cầu event log phải chọn một khái niệm trường hợp duy nhất, có nghĩa là mỗi sự kiện chỉ liên quan đến một đối tượng ứng với một trường hợp thực hiện quy trình.

Với dữ liệu event log đơn giản, khi mỗi sự kiện chỉ liên quan đến một đối tượng trong một trường hợp thực hiện quy trình, chúng ta có thể áp dụng ngay các phương pháp truyền thống để phân tích. Tuy nhiên, trong các trường hợp đặc biệt với các quy trình phức tạp hơn, dữ liệu event log không thể giảm xuống thành một trường hợp duy nhất. Với các quy trình phức tạp này, một sự kiện có thể liên quan đến nhiều đối tượng, chẳng hạn như khi một đơn hàng chứa nhiều mặt hàng và gói hàng khác nhau. Sự tương tác phức tạp giữa các đối tượng này khiến việc áp dụng tiêu chuẩn XES trở nên không khả thi, vì nó yêu cầu mỗi sự kiện chỉ có một đối tượng liên quan. Kết quả là, các phương pháp truyền thống dựa trên dữ liệu event log đơn giản sẽ gặp khó khăn trong việc phân tích các quy trình này một cách chính xác và toàn diện. Do đó, không thể áp dụng khái niệm

mỗi sự kiện chỉ có một đối tượng ứng với một trường hợp thực hiện quy trình, và việc chọn một loại đối tượng làm khái niệm trường hợp duy nhất có thể dẫn đến các vấn đề hội tụ và phân kỳ trong mô hình hóa quy trình.

Vậy nên, với những dữ liệu event log không đáp ứng được yêu cầu về tính đơn nhất của trường hợp thực hiện quy trình, cần phải tiền xử lý dữ liệu để đảm bảo mỗi sự kiện chỉ liên quan đến một trường hợp duy nhất. Điều này giúp đảm bảo rằng các phương pháp phân tích truyền thống có thể được áp dụng hiệu quả và chính xác, cung cấp cái nhìn sâu sắc và toàn diện về các quy trình kinh doanh phức tạp.

Một thao tác phổ biến được sử dụng để thực hiện quá trình tiền xử lý dữ liệu event log chính là thao tác “phân tách” (flattening). Thao tác này giúp chuyển đổi dữ liệu event log theo hướng nhiều đối tượng có thể trở thành thành dữ liệu event log truyền thống, dựa trên việc chọn một loại đối tượng cụ thể. Quá trình này bao gồm việc tách biệt các sự kiện phức tạp thành nhiều sự kiện đơn lẻ hoặc tạo ra các trường hợp mới để đảm bảo rằng mỗi sự kiện chỉ liên quan đến một loại đối tượng duy nhất. Dưới đây là các bước cụ thể của quá trình này:

- Xác định các loại đối tượng
 - Đầu tiên, ta cần xác định tất cả các loại đối tượng xuất hiện trong dữ liệu event log. Việc xác định này bao gồm việc phân loại các đối tượng theo các tiêu chí cụ thể để đảm bảo rằng mỗi loại đối tượng được định danh một cách rõ ràng và chính xác.
- Tạo bản ghi sự kiện phẳng cho mỗi loại đối tượng: Sau khi đã xác định các loại đối tượng, ta tiến hành tạo bản ghi sự kiện phẳng cho từng loại đối tượng. Quá trình này bao gồm các bước:
 - Duyệt qua tất cả các sự kiện: Ta sẽ duyệt qua tất cả các sự kiện trong dữ liệu event log ban đầu. Mỗi sự kiện sẽ được kiểm tra để xác định xem nó có liên quan đến loại đối tượng cụ thể hay không.

- Xác định sự kiện liên quan: Đối với mỗi sự kiện, xác định xem sự kiện đó có liên quan đến loại đối tượng cụ thể hay không. Nếu sự kiện có liên quan, nó sẽ được giữ lại trong bản ghi sự kiện phẳng.
- Loại bỏ thông tin không liên quan: Các thông tin liên quan đến các đối tượng khác sẽ bị loại bỏ khỏi sự kiện. Điều này giúp đảm bảo rằng mỗi bản ghi sự kiện phẳng chỉ chứa các sự kiện liên quan đến một loại đối tượng duy nhất.

Kết quả của quá trình phân tách là một bản ghi sự kiện phẳng chỉ chứa các sự kiện liên quan đến một loại đối tượng duy nhất. Mỗi loại đối tượng sẽ có một bản ghi sự kiện phẳng riêng biệt, đảm bảo tính tập trung và rõ ràng.

Để minh họa quá trình phân tách dữ liệu sự kiện, hãy xem xét một ví dụ cụ thể với dữ liệu sự kiện liên quan đến đơn hàng (Order) và mặt hàng (Item). Giả sử chúng ta có một dữ liệu event log như sau:

Event ID	Activity	Timestamp	Order	Item
e1	place order	2024-01-01 08:20:00	o_1	i_1, i_2
e2	check availability	2024-01-01 08:21:00		i_1
e3	ship item	2024-01-01 08:23:00	o_1	i_1

Bảng 2.1: Dữ liệu event log minh họa

Để tạo một bản ghi sự kiện phẳng cho loại đối tượng “Order”, chúng ta sẽ duyệt qua tất cả các sự kiện và giữ lại các sự kiện liên quan đến “Order”. Các thông tin liên quan đến đối tượng khác sẽ bị loại bỏ. Kết quả sẽ là một nhật ký sự kiện chỉ chứa các sự kiện liên quan đến “Order”. Sau khi phân tách cho loại đối tượng “Order”, ta thu được dữ liệu event log như sau:

Tương tự, để tạo bản ghi sự kiện phẳng cho loại đối tượng “Item”, ta sẽ giữ lại các sự kiện liên quan đến “Item” và loại bỏ các thông tin khác. Kết quả sẽ là:

Event ID	Activity	Timestamp	Order
e1	place order	2024-01-01 08:20:00	o_1
e3	ship item	2024-01-01 08:23:00	o_1

Bảng 2.2: Dữ liệu event log sau khi “phân tách” cho loại đối tượng “Order”

Event ID	Activity	Timestamp	Item
e1	place order	2024-01-01 08:20:00	i_1
e1	place order	2024-01-01 08:20:00	i_2
e2	check availability	2024-01-01 08:21:00	i_1
e3	ship item	2024-01-01 08:23:00	i_1

Bảng 2.3: Dữ liệu event log sau khi “phân tách” cho loại đối tượng “Item”

Quá trình phân tách dữ liệu sự kiện giúp tạo ra các bản ghi sự kiện chỉ chứa các sự kiện liên quan đến một loại đối tượng duy nhất. Ví dụ trên minh họa cách dữ liệu sự kiện ban đầu được tách ra thành các bản ghi sự kiện riêng biệt cho “Order” và “Item”. Điều này cho phép chúng ta áp dụng các kỹ thuật phân tích truyền thống dễ dàng hơn, vì mỗi bản ghi sự kiện chỉ liên quan đến một loại đối tượng duy nhất.

2.2 Phân tích hiệu năng quy trình từ dữ liệu event log bằng phương pháp truyền thống

2.2.1 Bước 1 - Tìm mô hình quy trình truyền thống từ dữ liệu event log

Mô hình quy trình Petri Nets là một ứng dụng chuyên biệt của mạng Petri trong việc mô hình hóa và phân tích các quy trình. Có thể hiểu đơn giản, Petri Net là một mạng lưới tĩnh và không thay đổi, được sử dụng để biểu diễn trạng thái và sự chuyển đổi trong các hệ thống phức tạp. Về cấu

trúc cơ bản, một mạng Petri sẽ bao gồm các thành phần sau:

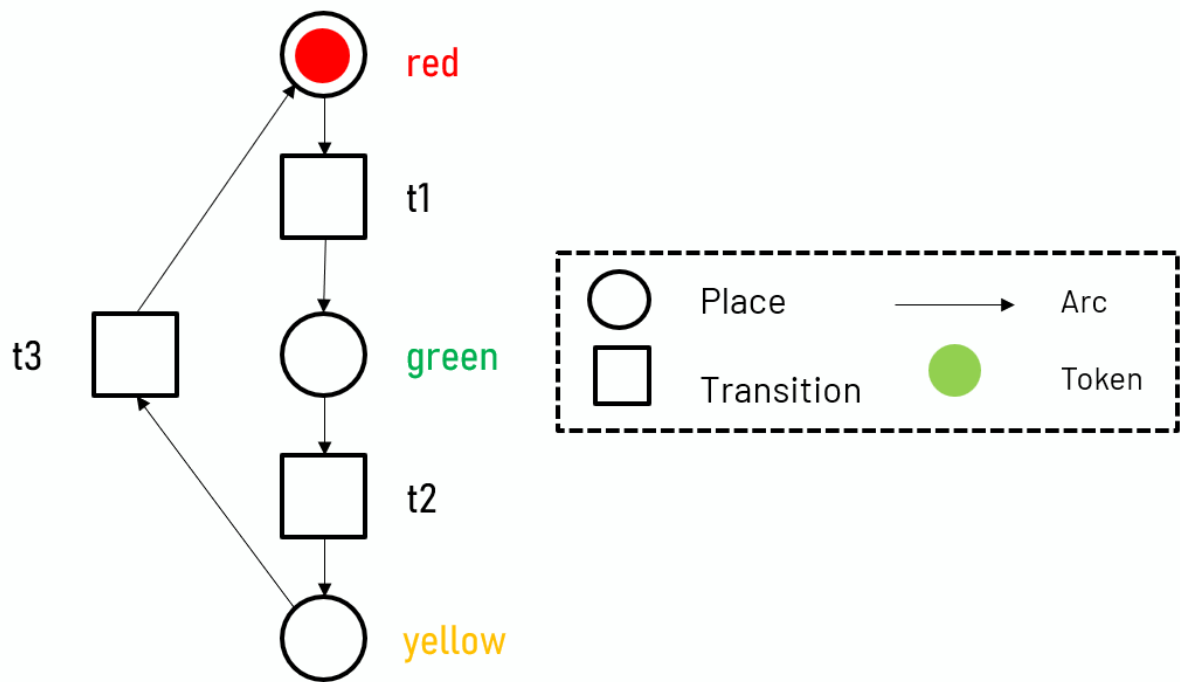
- Places (Điểm): Đại diện cho các trạng thái hoặc các đối tượng liên quan đến quy trình. Places có thể chứa các mã thông báo (tokens) và có thể được phân biệt với nhau bằng màu sắc.
- Transitions (Chuyển tiếp): Đại diện cho các sự kiện hoặc quá trình chuyển đổi trạng thái trong quy trình.
- Tokens (Mã thông báo): Đại diện cho các tài nguyên hoặc thông tin di chuyển trong các quy trình. Các tokens có thể di chuyển từ điểm (places) này sang điểm (places) khác. Một Petri Net có thể chứa một hoặc nhiều mã thông báo (tokens).
- Arcs (Cung): Kết nối giữa place và transition, xác định sự chuyển động của tokens.

Trạng thái trong một Petri Nets được gọi là “đánh dấu” (marking). Do đó, điểm “đánh dấu” ban đầu được xem là trạng thái khởi đầu của quy trình. Như ở hình 2.2, nếu đèn giao thông có trạng thái khởi đầu là “red”, chúng ta sẽ đặt một token tại vị trí “red”.

Về nguyên lý hoạt động, Petri Net tuân theo các quy tắc quy định việc kích hoạt sự chuyển tiếp bên trong mạng:

- Kích hoạt Transition: Một quá trình chuyển tiếp (transition) sẽ được kích hoạt (enable) nếu có đủ mã thông báo (tokens) tại mỗi điểm (places) đầu vào.
- Tiêu thụ và sản xuất Tokens: Khi một quá trình chuyển tiếp được kích hoạt, nó có thể tiêu thụ mã thông báo (tokens) từ các điểm (places) đầu vào và sản xuất ra mã thông báo (tokens) tại các điểm (places) đầu ra.

Về quy tắc đầu tiên: một chuyển tiếp (transition) sẽ được kích hoạt khi mỗi vị trí điểm (place) đầu vào của nó có chứa một mã thông báo (token). Ví

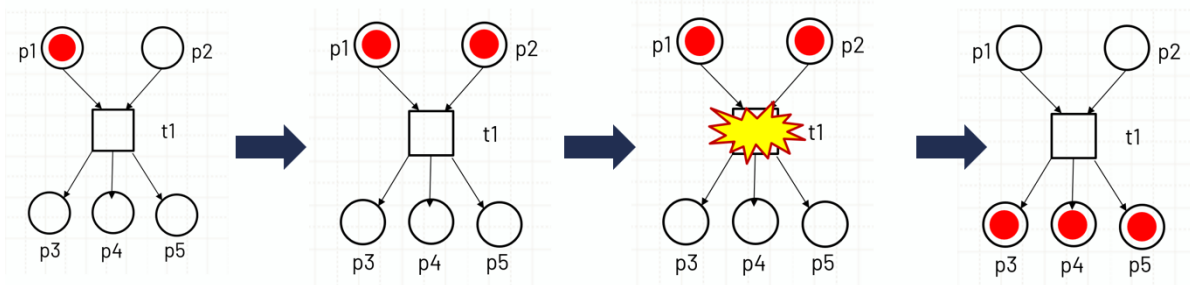


Hình 2.2: Ví dụ về một mô hình quy trình Petri Nets mô tả quá trình chuyển đổi đơn giản của đèn giao thông.

dụ cụ thể trong hình 2.3, nếu vị trí điểm (place) $p1$ có chứa một mã thông báo (token) và vị trí điểm (place) $p2$ trống, thì chuyển tiếp (transition) $t1$ sẽ không thể được kích hoạt (enable). Tuy nhiên, khi chúng ta thêm một mã thông báo (token) vào vị trí điểm (place) $p2$, chuyển tiếp (transition) $t1$ sẽ được kích hoạt (enable) ngay lập tức.

Các chuyển tiếp (transition) đã được kích hoạt có khả năng thực hiện quá trình tiêu thụ và sản xuất mã thông báo (token), quá trình này còn được gọi là thực thi (firing). Tiêu thụ có thể được hiểu là sự xuất hiện hoặc thực hiện của một chuyển tiếp (transition). Khi một chuyển tiếp (transition) được thực hiện, nó sẽ tiêu thụ một mã thông báo (token) từ mỗi vị trí điểm (place) đầu vào của nó và sản xuất một mã thông báo (token) cho mỗi vị trí điểm (place) đầu ra của nó. Điều này có nghĩa là quá trình tiêu thụ và sản xuất sẽ diễn ra đồng thời.

Ví dụ, trở lại hình 2.3, nếu chuyển tiếp (transition) $t1$ được thực hiện, nó sẽ tiêu thụ các mã thông báo (token) từ các vị trí điểm (place) $p1$ và $p2$,

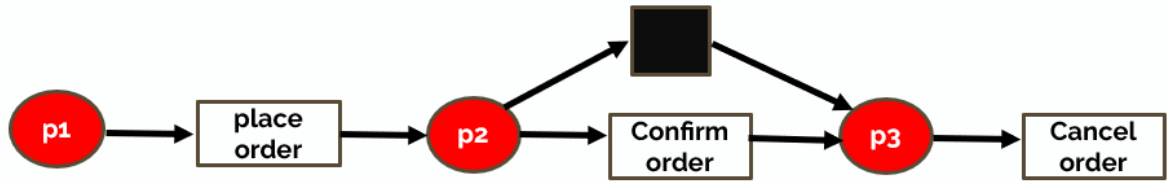


Hình 2.3: Ví dụ minh họa về quá trình kích hoạt, tiêu thụ và sản xuất mã thông báo (token) trên Petri Net.

và sau đó sản xuất các mã thông báo (token) tại ba vị trí điểm (place) đầu ra. Quá trình này minh họa cách các mã thông báo (token) di chuyển qua các vị trí điểm (place) và chuyển tiếp (transition) trong Petri Net, thể hiện cả sự tiêu thụ (tokens bị lấy đi từ các vị trí đầu vào) và sự sản xuất (tokens được đặt vào các vị trí đầu ra).

Một điểm đặc biệt giúp mô hình quy trình Petri Net trở nên linh hoạt và có khả năng mô hình hóa các quy trình thực tế một cách chính xác là khả năng phù hợp hoàn toàn (fit) với dữ liệu event log được đưa vào. Để đạt được điều này, mô hình Petri Net sử dụng một thành phần phụ là các chuyển tiếp im lặng (silent transitions). Chuyển tiếp im lặng (silent transitions) là các chuyển tiếp không đại diện cho bất kỳ hoạt động thực tế nào trong quy trình. Chúng thường không được ghi lại trong dữ liệu event log và không có sự kiện thực tế nào tương ứng với chúng. Vai trò của các chuyển tiếp im lặng (silent transitions) là giúp tăng khả năng biểu diễn của mô hình Petri Net, cho phép mô hình hóa các luồng điều khiển phức tạp như các điều kiện rẽ nhánh hoặc kết hợp các hoạt động mà không cần thêm các hoạt động thực tế vào mô hình. Điều này làm cho mô hình Petri Net linh hoạt hơn và có khả năng mô hình hóa các quy trình thực tế một cách hoàn toàn chính xác. Trong ngữ cảnh mô hình quy trình được mô hình hóa hoàn toàn từ dữ liệu event log, không phải là mô hình quy trình được thiết kế ban đầu trước khi thực hiện quy trình và tạo ra dữ liệu event log, nếu không có bộ lọc hay giới hạn đặc biệt nào, mô hình quy trình tìm

được thường sẽ phù hợp hoàn toàn với dữ liệu event log ban đầu.



Hình 2.4: Ví dụ minh họa mô hình quy trình Petri Net có thành phần chuyển tiếp im lặng (silent transitions).

Ở hình 2.4, chúng ta có thể thấy một mô hình quy trình Petri Net minh họa cho quá trình hủy đơn hàng. Trong trường hợp đầu tiên, một đơn hàng sau khi được đặt và đã xác nhận có thể bị hủy do một số lý do như thay đổi trong khối lượng hàng hóa, lỗi hệ thống, hoặc thay đổi chính sách. Khi đó, đơn hàng sẽ không được tiếp tục xử lý và kết thúc tại điểm hủy bỏ. Trong trường hợp thứ hai, một đơn hàng sau khi được đặt nhưng chưa được xác nhận cũng có thể bị hủy bỏ. Điều này có thể xảy ra do khách hàng thay đổi nhu cầu, nhập sai thông tin, hoặc thay đổi quyết định trước khi đơn hàng được xác nhận. Mô hình quy trình Petri Net này có thể khớp chính xác với cả hai trường hợp hủy đơn hàng, từ khi đơn hàng được đặt cho đến khi quá trình hủy hoàn tất.

Hiện nay, có rất nhiều kỹ thuật được sử dụng trong lĩnh vực khám phá quy trình nhằm tìm ra một mô hình quy trình phù hợp từ dữ liệu đầu vào. Trong số các kỹ thuật này, một trong những kỹ thuật nổi bật là kỹ thuật khai thác quy nạp (Inductive Mining). Inductive mining là một kỹ thuật khám phá quy trình được thiết kế để xây dựng một mô hình quy trình từ dữ liệu event log. Kỹ thuật này tập trung vào việc xây dựng các mô hình cấu trúc khối (block-structured models) có tính chất vừa hợp lệ vừa phù hợp với các hành vi quan sát được trong dữ liệu event log. Điểm mạnh của kỹ thuật này nằm ở khả năng xây dựng các mô hình có cấu trúc rõ ràng và dễ hiểu, đồng thời đảm bảo rằng mô hình có thể tái tạo lại tất cả các hành vi đã được quan sát. Inductive mining không chỉ nổi bật về tính chính xác

mà còn về tính linh hoạt và khả năng mở rộng. Phương pháp này có thể áp dụng cho nhiều loại dữ liệu event log khác nhau, từ các quy trình đơn giản đến phức tạp.

Thông thường thì kĩ thuật inductive mining sẽ có các bước chính như sau:

- Bước 1 - Tiền xử lý event log: Mục tiêu của bước này là chuẩn bị dữ liệu event log phù hợp để có thể áp dụng thuật toán. Đầu tiên ta cần xác định tất cả các hoạt động riêng biệt diễn ra từ dữ liệu event log ban đầu. Điều này có thể bao gồm việc làm sạch dữ liệu, loại bỏ các dữ liệu không hợp lệ, và xác định các hoạt động chính. Sau đó tạo các trace đại diện cho các chuỗi sự kiện. Ví dụ, trong dữ liệu event log có các trace như $t1 : a, b, c$, $t2 : a, c, b$, các hoạt động là a, b, c .
- Bước 2 - Phân chia ban đầu: Ở bước này ta sẽ tiến hành phân chia event log thành các tập con dựa trên các mẫu hành vi quan sát được. Việc này giúp đơn giản hóa việc phân tích và xác định cấu trúc quy trình. Sẽ có 4 phương thức phân chia như sau:
 - Phân chia theo tuần tự (Sequence): Nếu có một hoạt động luôn xảy ra trước các hoạt động khác, phân chia log dựa trên hoạt động này. Ví dụ, nếu a luôn xuất hiện đầu tiên trong tất cả các trace, phân chia log thành các phần bắt đầu từ a . Điều này giúp xác định rõ ràng các trình tự hoạt động.
 - Phân chia theo song song (Parallelism): Nếu có các hoạt động có thể diễn ra song song, phân chia log để phản ánh điều này. Ví dụ, nếu b và c có thể xảy ra đồng thời, phân chia log để bao gồm cả hai trường hợp. Điều này giúp nhận diện các hoạt động có thể thực hiện đồng thời, tăng hiệu suất quy trình.
 - Phân chia theo lựa chọn (Choice): Nếu có các hoạt động xảy ra luân phiên (chỉ một trong hai), phân chia log để phản ánh sự lựa chọn này. Ví dụ, nếu b và d là lựa chọn lẫn nhau, log sẽ

được chia thành các phần chứa b và các phần chứa d . Điều này giúp mô hình hóa các quyết định trong quy trình.

- Phân chia theo vòng lặp (Loop): Nếu có các hoạt động xảy ra lặp lại, nhận diện và phân chia log để phản ánh vòng lặp. Ví dụ, nếu c có thể lặp lại nhiều lần, log sẽ được chia để bao gồm các vòng lặp c . Điều này giúp xác định và quản lý các hoạt động lặp lại trong quy trình.
- Bước 3 - Phân rã đệ quy: Sau khi phân chia ban đầu tại bước 2, ta sẽ tiếp tục tiến hành phân chia các tập con thành các phần nhỏ hơn. Ta sẽ áp dụng logic phân chia tương tự: Đối với mỗi tập con được tạo ra từ bước phân chia ban đầu, tiếp tục áp dụng các phân chia theo tuần tự, song song, lựa chọn, và vòng lặp để phân rã log thành các phần không thể phân chia được nữa. Đến khi đạt được các phần không thể phân chia thêm (atomic parts) chỉ còn các hoạt động đơn lẻ thì quá trình phân rã hoàn tất. Việc này giúp tạo ra một mô hình chi tiết và đầy đủ của quy trình.
- Bước 4 - Xây dựng cây quy trình (Process Tree): Ta sẽ tiến hành kết hợp các phần đã phân chia thành một cấu trúc cây quy trình. Quá trình này sẽ bao gồm hai bước như sau:
 - Tạo nút và lá: Mỗi phần nhỏ hơn trong log được biểu diễn bằng các nút trong cây quy trình, với các lá đại diện cho các hoạt động và các nút nội tại đại diện cho các toán tử như tuần tự, song song, lựa chọn, và vòng lặp. Điều này giúp biểu diễn trực quan cấu trúc quy trình.
 - Kết hợp các nút: Kết hợp các nút và lá để tạo thành cây quy trình phản ánh toàn bộ cấu trúc quy trình. Quá trình này giúp xác định rõ ràng mối quan hệ giữa các hoạt động và cách chúng tương tác với nhau.

- Bước 5 - Chuyển đổi cây quy trình thành Petri Net: Cuối cùng ta sẽ tiến hành chuyển đổi cây quy trình thành một mô hình quy trình Petri net hoàn chỉnh. Đầu tiên ta sẽ tạo các điểm và các chuyển tiếp (places and transitions): Mỗi nút và lá trong cây quy trình được chuyển thành các điểm và chuyển tiếp tương ứng trong Petri net. Sau đó ta tiến hành kết nối các phần tử: Các cung (arcs) được tạo ra để kết nối các điểm và chuyển tiếp, mô tả luồng và kiểm soát hoạt động. Cuối cùng là xác minh tính hợp lệ để đảm bảo rằng Petri net kết quả không có deadlock và có thể tái tạo tất cả hành vi quan sát được từ dữ liệu event log. Điều này giúp đảm bảo mô hình quy trình chính xác và hiệu quả.

2.2.2 Bước 2 - Tính các kết quả trung gian liên quan đến chỉ số hiệu năng từ mô hình quy trình và dữ liệu event log

Để tính toán được các kết quả trung gian liên quan đến chỉ số hiệu năng từ mô hình quy trình và dữ liệu event log thì một phương pháp được sử dụng rộng rãi để thực hiện điều này chính là phương pháp phát lại (replay) dựa trên mã thông báo (token). Đầu vào của phương pháp sẽ là một mô hình quy trình Petri Net cùng với dữ liệu event log. Ý tưởng của phương pháp này là sẽ sử dụng các mã thông báo (tokens) để đối chiếu dữ liệu event log với mô hình quy trình, xác định được các lần sự kiện xuất hiện (event occurrences) và các lượt truy cập bằng mã thông báo (token visits) để từ đó tính toán kết quả trung gian phục vụ cho quá trình tính toán các chỉ số hiệu năng.

Đầu tiên, về các lần xuất hiện sự kiện (event occurrences), ta có thể hiểu các lần xuất hiện sự kiện sẽ biểu thị cho sự xuất hiện của một sự kiện liên quan đến một quá trình chuyển tiếp hay. Nói cách khác, chúng đại diện cho việc kích hoạt một chuyển tiếp với các mã thông báo (token) cụ thể từ các điểm (place) trong mô hình quy trình. Thành phần của một lần

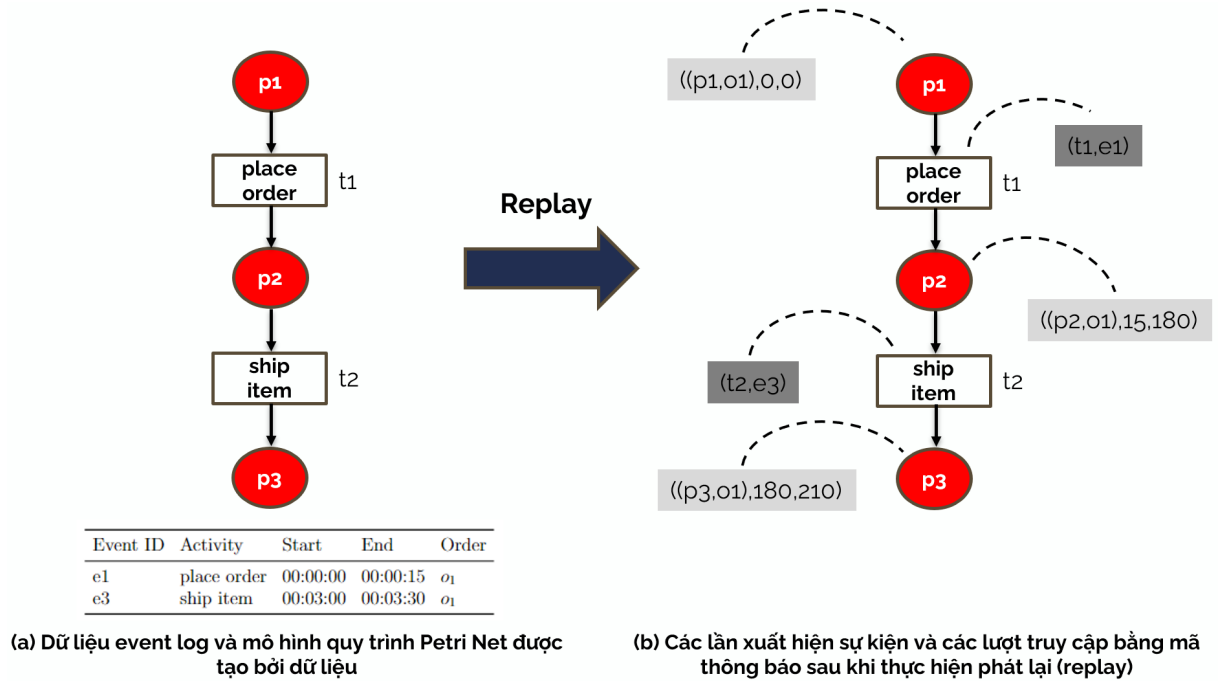
xuất hiện sự kiện sẽ là một cặp gồm hai thành phần chính: một chuyển tiếp và một sự kiện. Ví dụ ta có một lần xuất hiện sự kiện $(t1, e1)$ thì ta có thể hiểu rằng chuyển tiếp $t1$ được liên kết với sự kiện $e1$ tạo nên một lần xuất hiện sự kiện là $(t1, e1)$.

Về các lượt truy cập bằng mã thông báo (token visits), một lượt truy cập bằng mã thông báo mô tả sự di chuyển của một mã thông báo (token) tới một vị trí (place) tương ứng trong mô hình quy trình Petri Net. Nó ghi nhận lại thời điểm bắt đầu của việc truy cập, tức là thời điểm mà mã thông báo được tạo ra, và thời điểm kết thúc của việc truy cập, tức là thời điểm mà mã thông báo đã được sử dụng xong hoặc tiêu thụ. Một lượt truy cập bằng mã thông báo bao gồm hai thành phần chính: một mã thông báo (token) và cặp thông tin về thời gian bắt đầu và thời gian kết thúc. Ví dụ, ta có một lượt truy cập bằng mã thông báo là $((p2, o1), 15, 180)$. Điều này có nghĩa là mã thông báo $(p2, o1)$ được tạo ra tại vị trí $p2$ vào thời điểm 15 và bị tiêu thụ vào thời điểm 180.

Để thực hiện việc phát lại (replay) dữ liệu event log trên mô hình quy trình Petri Net, ta sẽ sử dụng hàm phát lại (replay function). Hàm phát lại sẽ thực hiện ánh xạ các sự kiện trong dữ liệu event log vào mô hình quy trình Petri Net. Quá trình này tạo ra các lần xuất hiện sự kiện và các lượt ghé thăm bằng mã thông báo bằng cách mô phỏng luồng di chuyển của các mã thông báo (token) qua Petri Net. Nhìn vào hình 2.5, ta có thể thấy được kết quả khi thực hiện một quá trình phát lại (replay). Những ô màu xám nhạt thể hiện cho những lượt ghé thăm bằng mã thông báo (token visits), trong khi những ô màu xám đậm sẽ tượng trưng cho những lần xuất hiện sự kiện (event occurrences).

2.2.3 Bước 3 - Tính các chỉ số hiệu năng từ các kết quả trung gian và mô hình quy trình

Trong bước này, chúng ta sẽ tiến hành tính toán các chỉ số hiệu năng dựa trên các kết quả trung gian. Các kết quả trung gian này bao gồm các



Hình 2.5: Ví dụ về thực hiện phát lại dữ liệu event log trên Petri Net.

lần xuất hiện của sự kiện (event occurrences) và các lượt truy cập bằng mã thông báo (token visits), được ghi nhận lại thông qua quá trình phát lại (replay) đã thực hiện ở bước 2. Việc phân tích chính xác các chỉ số hiệu năng là một bước quan trọng để đánh giá và cải thiện hiệu quả hoạt động của quy trình.

Các chỉ số hiệu năng sẽ được tính toán cho mỗi lần xuất hiện sự kiện (event occurrence). Trước khi thực hiện tính toán, chúng ta cần xem xét và liên kết mỗi lần xuất hiện sự kiện với lượt truy cập mã thông báo tương ứng. Việc này phải đảm bảo rằng lượt truy cập mã thông báo liên quan đến chuyển tiếp (transition) của lần xuất hiện sự kiện. Như ở hình 2.5, ta có thể thấy lượt truy cập mã thông báo liên quan đến lần xuất hiện sự kiện ($t2, e3$) sẽ là $((p2, o1), 15, 180)$ và nó sẽ liên quan đến chuyển tiếp (transition) là “ship item”.

Theo phương pháp truyền thống, ba chỉ số chính được sử dụng để đánh giá hiệu năng của một quy trình bao gồm thời gian lưu trú (sojourn time), thời gian chờ (waiting time) và thời gian thực thi (service time). Mỗi chỉ số này cung cấp một góc nhìn khác nhau về hiệu năng của quy trình và có

các phương pháp tính toán riêng biệt cụ thể như sau:

- Thời gian lưu trú (Sojourn Time): Thời gian lưu trú có thể được hiểu là tổng thời gian mà một đối tượng (ở đây sẽ là một mã thông báo) ở trong một địa điểm cụ thể trong hệ thống, từ lúc nó xuất hiện đến khi nó rời khỏi địa điểm đó. Thời gian lưu trú của một mã thông báo trong một địa điểm được tính bằng cách lấy dấu thời gian khi mã thông báo được tiêu thụ trừ đi dấu thời gian khi mã thông báo được tạo ra.
- Thời gian chờ (Waiting Time): Thời gian chờ là khoảng thời gian mà một đối tượng phải chờ đợi trước khi được xử lý hoặc thực thi trong hệ thống. Đây là thời gian mà mã thông báo (token) chưa được xử lý dù đã sẵn sàng. Thời gian chờ của một mã thông báo được tính bằng cách lấy dấu thời gian khi mã thông báo (token) thực sự bị tiêu thụ trừ đi dấu thời gian khi chuyển tiếp tiêu thụ mã thông báo được kích hoạt hoàn toàn.
- Thời gian thực thi (Service Time): Thời gian thực thi là khoảng thời gian mà một đối tượng thực sự được xử lý hoặc thực thi trong hệ thống. Đây là khoảng thời gian mà mã thông báo đang được hệ thống xử lý. Thời gian thực thi sẽ được tính là khoảng thời gian từ lúc mã thông báo thực sự bị tiêu thụ cho đến khi sự kiện liên quan đến mã thông báo kết thúc.

Chương 3

Phân tích hiệu năng quy trình từ dữ liệu event log bằng phương pháp xem xét nhiều đối tượng

Chương này trình bày về phương pháp xem xét nhiều đối tượng được [6] đề xuất để giải quyết bài toán phân tích hiệu năng quy trình từ dữ liệu event log; đây là phương pháp chính mà chúng em tập trung tìm hiểu trong khóa luận. Đầu tiên, chúng em đưa ra vấn đề gặp phải của phương pháp truyền thống. Sau đó, chúng em trình bày về ý tưởng của phương pháp xem xét nhiều đối tượng để giải quyết vấn đề gặp phải của phương pháp truyền thống. Cuối cùng, chúng em trình bày chi tiết các bước thực hiện của phương pháp xem xét nhiều đối tượng, bao gồm: (1) tìm mô hình quy trình có xem xét nhiều đối tượng từ dữ liệu event log, (2) tính các kết quả trung gian liên quan đến chỉ số hiệu năng từ mô hình quy trình và dữ liệu event log, (3) tính các chỉ số hiệu năng từ các kết quả trung gian và mô hình quy trình. Phương pháp xem xét nhiều đối tượng cũng có 3 bước tương tự như phương pháp truyền thống đã trình bày tại chương 2, nhưng khác biệt ở chỗ: thay mô hình quy trình truyền thống bằng mô hình quy trình có xem xét nhiều đối tượng, và bổ sung thêm các chỉ số hiệu năng mà liên quan đến sự tương tác của các đối tượng.

3.1 Vấn đề gặp phải của phương pháp truyền thống

Với dữ liệu event log đơn giản, khi mỗi sự kiện chỉ liên quan đến một đối tượng trong một trường hợp thực hiện quy trình, chúng ta có thể áp dụng ngay các phương pháp truyền thống để phân tích. Tuy nhiên, trong các trường hợp đặc biệt với các quy trình phức tạp hơn, một sự kiện có thể liên quan đến nhiều đối tượng (ví dụ như khi một đơn hàng chứa nhiều mặt hàng và gói hàng khác nhau). Khi đó, để đảm bảo dữ liệu event log khi phân tích hiệu năng quy trình bằng phương pháp truyền thống đúng với giả định ban đầu, chúng em sẽ tiền xử lý dữ liệu [2.1.2]. Quá trình này bao gồm việc tách biệt các sự kiện phức tạp thành nhiều sự kiện đơn lẻ hoặc tạo ra các trường hợp mới để đảm bảo rằng mỗi sự kiện chỉ liên quan đến một loại đối tượng duy nhất.

Tuy nhiên, khi áp dụng cách làm này cũng sẽ phát sinh một số vấn đề như sau:

- Vấn đề đầu tiên phải nhắc đến chính là mô hình quy trình sau khi sử dụng phương pháp truyền thống có thể sẽ không phản ánh một cách chính xác so với quy trình thực tế. Việc tiền xử lý dữ liệu đối với các sự kiện có tính đồng thời, có sự lặp lại, hoặc có các mối quan hệ phức tạp giữa các đối tượng liên quan sẽ làm cho kết quả tính toán các chỉ số hiệu năng không được chính xác. Điều này dẫn đến các chỉ số hiệu năng như thời gian lưu trú (Sojourn Time), thời gian chờ (Waiting Time), thời gian thực thi (Service Time) bị tính toán sai lệch.
- Thiếu sót trong việc xem xét sự tương tác giữa các đối tượng có liên quan. Không chỉ dẫn đến các chỉ số hiệu năng được tính toán sai mà còn làm các chỉ số hiệu năng liên quan nhiều đối tượng cũng không được xem xét tới. Quan trọng hơn, bằng cách xem xét sự

tương tác giữa các đối tượng, chúng ta có thể xác định được các chỉ số hiệu năng có xem xét nhiều đối tượng như thời gian luồng (Flow Time), thời gian đồng bộ hóa (Synchronization Time), thời gian gộp (Pooling Time), thời gian trễ (Lagging Time). Từ đó, có thể có được những góc nhìn chi tiết và rõ ràng hơn về quy trình đang xét.

Để minh họa cho những vấn đề gặp phải này, chúng em sẽ sử dụng lại ví dụ đầu vào đã được trình bày ở chương [1]. Để đúng với giả định ban đầu, sau khi tách riêng từng đối tượng, ta được bảng [3.1] là dữ liệu event log của quy trình xét nghiệm máu đối với đối tượng Test T_1 , có các bước là: Prepare test, Conduct test và Publish test.

Event ID	Activity	Test	Time start	Time end
e1	Prepare test	T_1	0:00:00	0:00:15
e4	Conduct test	T_1	0:03:00	0:04:00
e5	Publish test	T_1	0:04:30	0:05:00

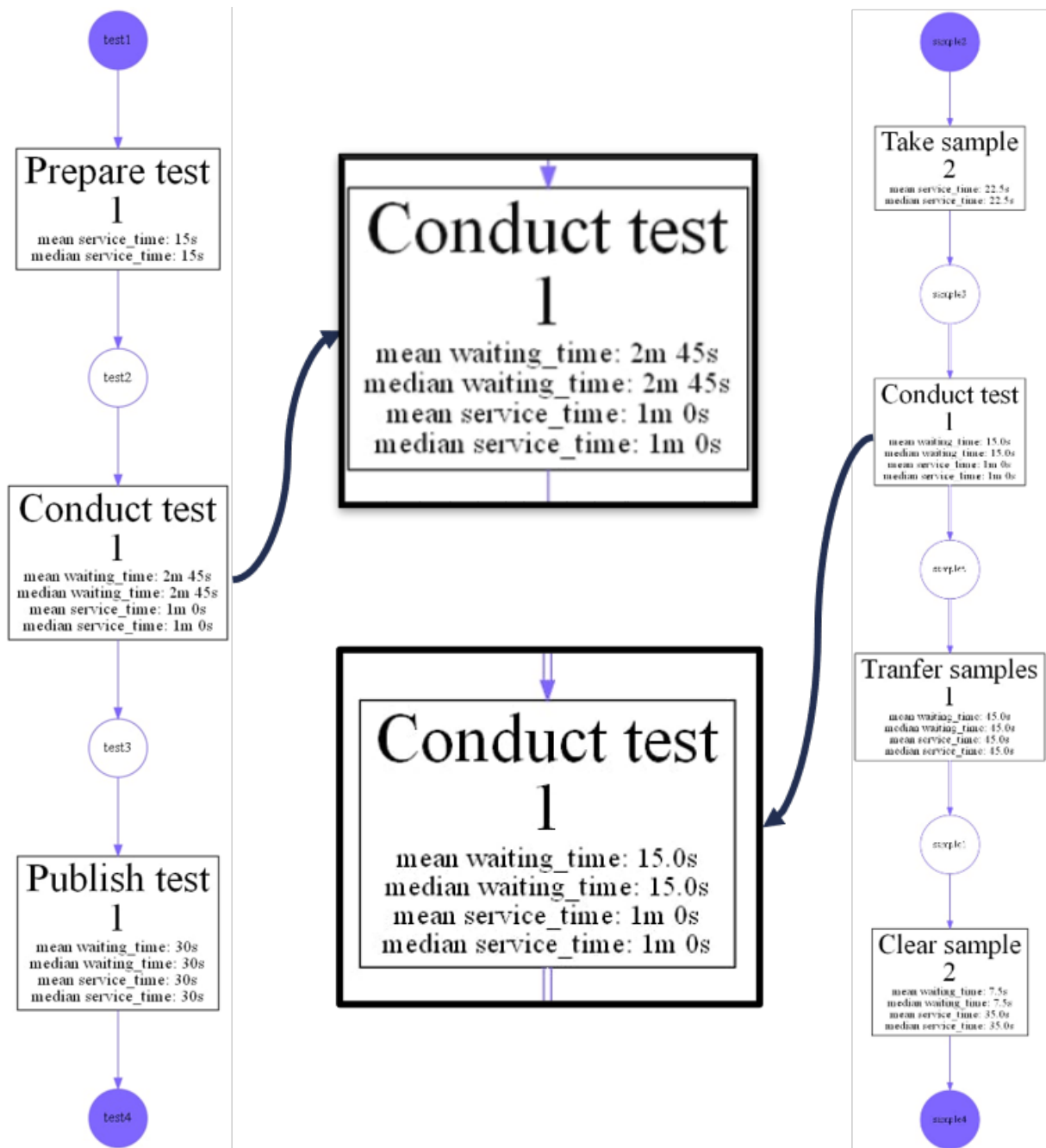
Bảng 3.1: Dữ liệu event log của quy trình xét nghiệm máu đối với đối tượng Test.

Event ID	Activity	Sample	Time start	Time end
e2	Take sample	S_1	0:02:00	0:02:30
e3	Take sample	S_2	0:02:30	0:02:45
e4	Conduct test	S_1	0:03:00	0:04:00
e4	Conduct test	S_2	0:03:00	0:04:00
e6	Transfer sample	S_1	0:04:45	0:05:30
e6	Transfer sample	S_2	0:04:45	0:05:30
e7	Clear sample	S_1	0:05:30	0:05:45
e8	Clear sample	S_2	0:05:45	0:06:40

Bảng 3.2: Dữ liệu event log của quy trình xét nghiệm máu của đối tượng Sample.

Thực hiện tương tự, sẽ được các bảng [3.2] tương ứng với đối tượng Sample S_1 và S_2 , gồm các bước là: Take sample, Conduct test, Transfer

sample, Clear sample. Sau khi áp dụng các bước của phương pháp truyền thống sẽ được kết quả như hình [3.1].



Hình 3.1: Mô hình quy trình xét nghiệm máu của đối tượng *Test* và *Sample*.

Dựa vào 2 mô hình quy trình con thu được này, có thể dễ dàng nhận thấy:

- Khi xét một trong những thông số đo lường hiệu năng của quy trình

xét nghiệm máu là thời gian chờ (Waiting Time) tại sự kiện Conduct test thì thời gian chờ trung bình (Mean Waiting Time) khi tách riêng từng đối tượng Test và Sample tuy cùng một sự kiện Conduct Test nhưng lại không giống nhau. Trong đó, Mean Waiting Time của đối tượng Test là 2 phút 45 giây, còn của đối tượng Sample là 15 giây.

- Giữa 2 đối tượng Test và Sample không thể hiện được mối quan hệ và liên quan với nhau. Trong quy trình thực thể, các đối tượng Test và Sample có tương tác qua lại với nhau. Chẳng hạn như khi tiến hành xét nghiệm máu (Conduct test) cần phải có đủ các đối tượng Test và Sample. Vậy nên nếu thiếu một trong số các đối tượng cũng sẽ làm ảnh hưởng đến quy trình và kết quả xét nghiệm.
- Dựa vào kết quả phân tích khi xét riêng những đối tượng này sẽ không thể phân tích hiệu năng quy trình xét nghiệm máu một cách chính xác. Từ đó dẫn đến các chỉ số hiệu năng liên quan sẽ không được chính xác.

Từ những vấn đề trên, thay vì sử dụng phương pháp truyền thống, một phương pháp mới đã được đề xuất chính là phương pháp xem xét nhiều đối tượng [6]. Ý tưởng của phương pháp sẽ được trình bày chi tiết hơn ở phần tiếp theo.

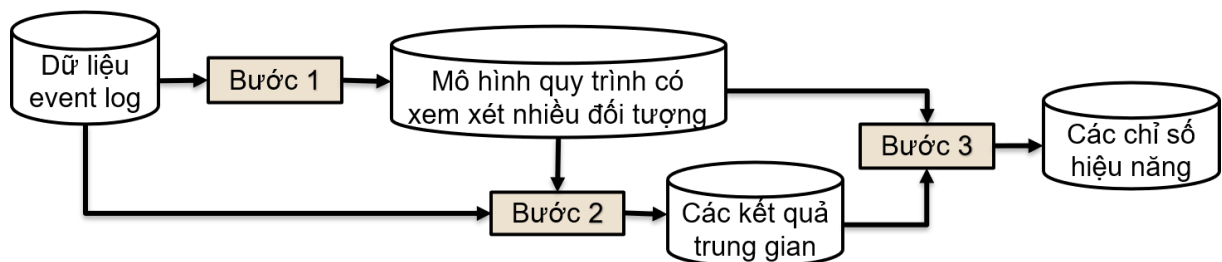
3.2 Ý tưởng của phương pháp xem xét nhiều đối tượng để giải quyết vấn đề gặp phải của phương pháp truyền thống

Nhận thấy nhược điểm của các phương pháp truyền thống, bài báo “OPerA: Object-Centric Performance Analysis” [6] đã đề xuất một phương pháp phân tích hiệu năng quy trình từ dữ liệu event log có xem xét tự liên quan, tương tác giữa nhiều đối tượng. Phương pháp này cho phép một sự

kiện có thể liên kết với nhiều trường hợp khác nhau, mở ra cánh cửa để giải quyết các hạn chế của phương pháp truyền thống. Cách làm của phương pháp này sẽ tương tự như cách làm của phương pháp truyền thống, được trình bày trong nghiên cứu [5], nhưng thay vì sử dụng mô hình quy trình dạng “Petri Net” , phương pháp sẽ sử dụng mô hình quy trình có xem xét nhiều đối tượng - “Object-Centric Petri Net” [1]. Bằng cách này, mô hình quy trình có khả năng phản ánh được thực tế chính xác hơn, từ đó cho ra kết quả tính toán các chỉ số hiệu năng của quy trình sẽ chính xác hơn.

Ngoài ra, sử dụng phương pháp phân tích hiệu năng quy trình từ dữ liệu event log có xem xét nhiều đối tượng [6] cũng đề xuất thêm các chỉ số hiệu năng liên quan đến sự tương tác của các đối tượng. Điều này có ý nghĩa quan trọng trong việc phân tích các quy trình phức tạp, nơi mà sự biến đổi của dữ liệu và luồng công việc có thể tạo ra nhiều hướng đi và kết quả khác nhau. Ngoài giải quyết được yêu cầu bài toán là phân tích hiệu năng quy trình có thông số tính toán hiệu năng, phương pháp này còn giải quyết được những hạn chế của phương pháp truyền thống như:

- Các sự kiện được thể hiện chính xác hơn bằng cách xem xét tính đồng thời, lặp lại và dựa vào mối quan hệ giữa các sự kiện với nhau.
- Kết quả tính toán các chỉ số hiệu năng và tương tác giữa các đối tượng được thể hiện chính xác hơn.



Hình 3.2: Tổng quan phương pháp xem xét nhiều đối tượng.

Quy trình để tạo ra một mô hình quy trình có xem xét nhiều đối tượng - “Object-Centric Petri Net” có thể được hiểu đơn giản như Hình [3.2]:

- Đầu tiên, phương pháp này sẽ bắt đầu khám phá (discover) một mô hình quy trình dạng Object-Centric Petri Net từ đầu vào là dữ liệu event log [3.3.1].
- Sau đó, các sự kiện trong dữ liệu event log được phát lại (replay) dựa trên việc khám phá này để cho ra các kết quả trung gian (gồm có token visits, event occurrences) [3.3.2].
- Dựa vào đó, phương pháp sẽ đo lường, tính toán và ghi lại các chỉ số hiệu suất để đưa ra được những thông số hiệu năng trực quan cho mô hình quy trình [3.3.3].

Kết quả là, mô hình quy trình không chỉ là một biểu đồ tĩnh mà còn là một công cụ động và linh hoạt, có khả năng phản ánh sự thay đổi trong điều kiện hoạt động thực tế. Điều này tạo ra cơ sở cho việc đưa ra các cải tiến hiệu quả và phản ứng nhanh chóng đối với các thay đổi trong môi trường kinh doanh hiện nay. Tiếp theo, chúng em sẽ trình bày chi tiết hơn về các bước được nêu trên để hiểu rõ hơn về phương pháp này.

3.3 Chi tiết về các bước thực hiện của phương pháp xem xét nhiều đối tượng

3.3.1 Bước 1 - Tìm mô hình quy trình có xem xét nhiều đối tượng từ dữ liệu event log

Mô hình quy trình Object-Centric Petri Net là một biến thể của mạng Petri [2.2.1] nhưng sẽ tập trung vào việc biểu diễn các quy trình dựa trên đối tượng. Mô hình sử dụng các điểm (place) và chuyển tiếp (transition) để mô tả các tương tác giữa các loại đối tượng và sự kiện trong quy trình. Điều này giúp ích cho việc hiển thị mối quan hệ phức tạp giữa các đối tượng khác nhau. Ngoài ra, mô hình có thể đưa ra cái nhìn sâu hơn về cách mà các đối tượng tương tác với nhau và ảnh hưởng đến quy trình

thông qua khả năng mô hình hóa các quan hệ phức tạp giữa các đối tượng và các sự kiện trong quy trình.

Trong Object-Centric Petri Net, không chỉ các sự kiện mà cả các đối tượng tham gia trong quy trình cũng được thể hiện rõ. Mỗi đối tượng được biểu diễn bằng các token. Các token này biểu diễn trạng thái của các đối tượng trong quy trình, và các sự kiện diễn ra khi các token chuyển đổi trạng thái.

Vì đa số dữ liệu event log hiện nay không phù hợp với giả định ban đầu của phương pháp truyền thống (mỗi sự kiện chỉ liên quan đến một đối tượng trong một trường hợp thực hiện quy trình). Vậy nên việc đầu tiên cần xử lý khi muốn tìm mô hình quy trình dạng Object-Centric Petri Nets (OCPN) chính là đưa dữ liệu event log về dạng phù hợp. Trong nghiên cứu “Discovering Object-Centric Petri Nets” [1], làm phẳng (flattening) dữ liệu event log ban đầu được thực hiện bằng cách biến đổi các đối tượng và các hoạt động thành các sự kiện và trạng thái phù hợp với mô hình. Điều này giúp tạo ra một mô hình quy trình dạng Object-Centric Petri Nets chi tiết và phù hợp với cấu trúc dữ liệu event log nhằm mục đích phân tích và tối ưu hóa các quy trình trong hệ thống. Để thực hiện việc này cần trải qua nhiều bước xử lý và áp dụng một số phương pháp kỹ thuật.

Quá trình này cho phép khám phá (discover) một mô hình quy trình phức tạp, có khả năng nắm bắt các mối quan hệ chồng chéo từ dữ liệu sự kiện, mang lại sự hiểu rõ và chi tiết hơn về hoạt động của các quy trình kinh doanh. Để tìm mô hình quy trình có xem xét nhiều đối tượng từ dữ liệu event log cần thực hiện các bước sau đây để khám phá (discover) mô hình quy trình có xem xét nhiều đối tượng - “Object-Centric Petri Net” (OCPN):

1. Tiền xử lý dữ liệu: sử dụng dữ liệu event log chứa thông tin về các sự kiện và các đối tượng liên quan. Mỗi sự kiện có thể liên kết với nhiều đối tượng khác nhau.
2. Bước 1: Xác định loại đối tượng (OT): Xác định các loại đối tượng

xuất hiện trong nhật ký sự kiện. Đây là các danh sách các loại đối tượng khác nhau mà các sự kiện trong nhật ký liên quan đến.

3. Bước 2: Làm phẳng nhật ký sự kiện (Flatten event log): Tạo ra các nhật ký sự kiện làm phẳng cho mỗi loại đối tượng. Điều này có nghĩa là tạo ra các bản sao của các sự kiện từ nhật ký gốc và chỉ giữ lại các sự kiện liên quan đến từng loại đối tượng cụ thể.
4. Bước 3: Khám phá mạng Petri trung tâm đối tượng (OCPN): Áp dụng các kỹ thuật khai thác quy trình để tạo ra một mạng Petri cho mỗi loại đối tượng:
 - Kỹ thuật khai thác: Sử dụng các thuật toán khai thác quy trình như region-based techniques, inductive mining techniques, hoặc các kỹ thuật khác không chia nhãn (label splitting).
 - Điều kiện: Yêu cầu một mạng Petri không có nhãn trùng lặp (duplicate labels), cho phép các chuyển tiếp câm (silent transitions), nghĩa là hàm nhãn l là một ánh xạ đơn ánh xạ (injective), có thể không hoàn chỉnh (partial).
5. Bước 4: Tích hợp các mạng Petri cho các đối tượng: Sau khi khám phá, các mạng Petri cho từng loại đối tượng cần được kết hợp lại thành một mạng Petri tổng thể cho toàn bộ quy trình, gọi là mạng Petri trung tâm đối tượng (Object-Centric Petri Net - OCPN).
 - Tổng hợp: Đảm bảo các tên chuyển tiếp và chỗ chứa là khác nhau trong các mạng khác nhau.
 - Gán nhãn: Các chuyển tiếp có cùng nhãn phải có cùng tên (name), nhờ vào hàm nhãn l là một ánh xạ đơn ánh xạ (injective).
6. Bước 5: Gán đối tượng cho các vị trí (Places): Gán từng loại đối tượng cho các vị trí trong mạng Petri tổng hợp. Điều này là có thể vì các vị trí cho các loại đối tượng khác nhau là không giao nhau.

7. Bước 6: Xác định các cạnh biến (Variable arcs): Xác định các cạnh biến trong mạng Petri tổng hợp, đây là các cạnh mà có thể sản xuất hoặc tiêu thụ nhiều token. Các cạnh này thường được xác định bằng các thông số như ngưỡng (threshold) và hàm điểm (score function) để đánh giá số lượng token liên quan đến sự kiện.
8. Bước 7: Hoàn tất mạng Petri trung tâm đối tượng (OCPN): Cuối cùng, trả về mạng Petri trung tâm đối tượng ($AN = (ON, Minit, Mfinal)$) với:
- ON: Mạng Petri trung tâm đối tượng kết hợp từ các mạng Petri cho từng loại đối tượng.
 - Minit, Mfinal: Đánh dấu ban đầu và kết thúc được sao chép từ các đánh dấu của các mạng Petri cá nhân cho từng loại đối tượng tương ứng.

3.3.2 Bước 2 - Tính các kết quả trung gian liên quan đến chỉ số hiệu năng từ mô hình quy trình và dữ liệu event log

Trong thực tế, đa số các sự kiện trong quy trình đều diễn ra không theo trình tự tuyến tính, thay vào đó là những trường hợp diễn ra đồng thời, có sự lặp lại hay có các mối quan hệ phức tạp. Điều này đòi hỏi một cách tiếp cận linh hoạt và đa chiều hơn trong việc phân tích và mô hình hóa quy trình. Vậy nên sau khi có kết quả của bước tìm mô hình quy trình có xem xét nhiều đối tượng từ dữ liệu event log, bước tiếp theo chính là kết hợp mô hình này với dữ liệu event log thông qua việc phát lại (replay). Quá trình này không chỉ giúp xác minh tính chính xác của mô hình quy trình so với thực tế diễn ra mà còn cho phép phát hiện ra những sai lệch và cung cấp cái nhìn sâu sắc về cách thức hoạt động của quy trình. Để làm được việc này, cần xem xét tính đồng thời, vòng lặp và xác định chính

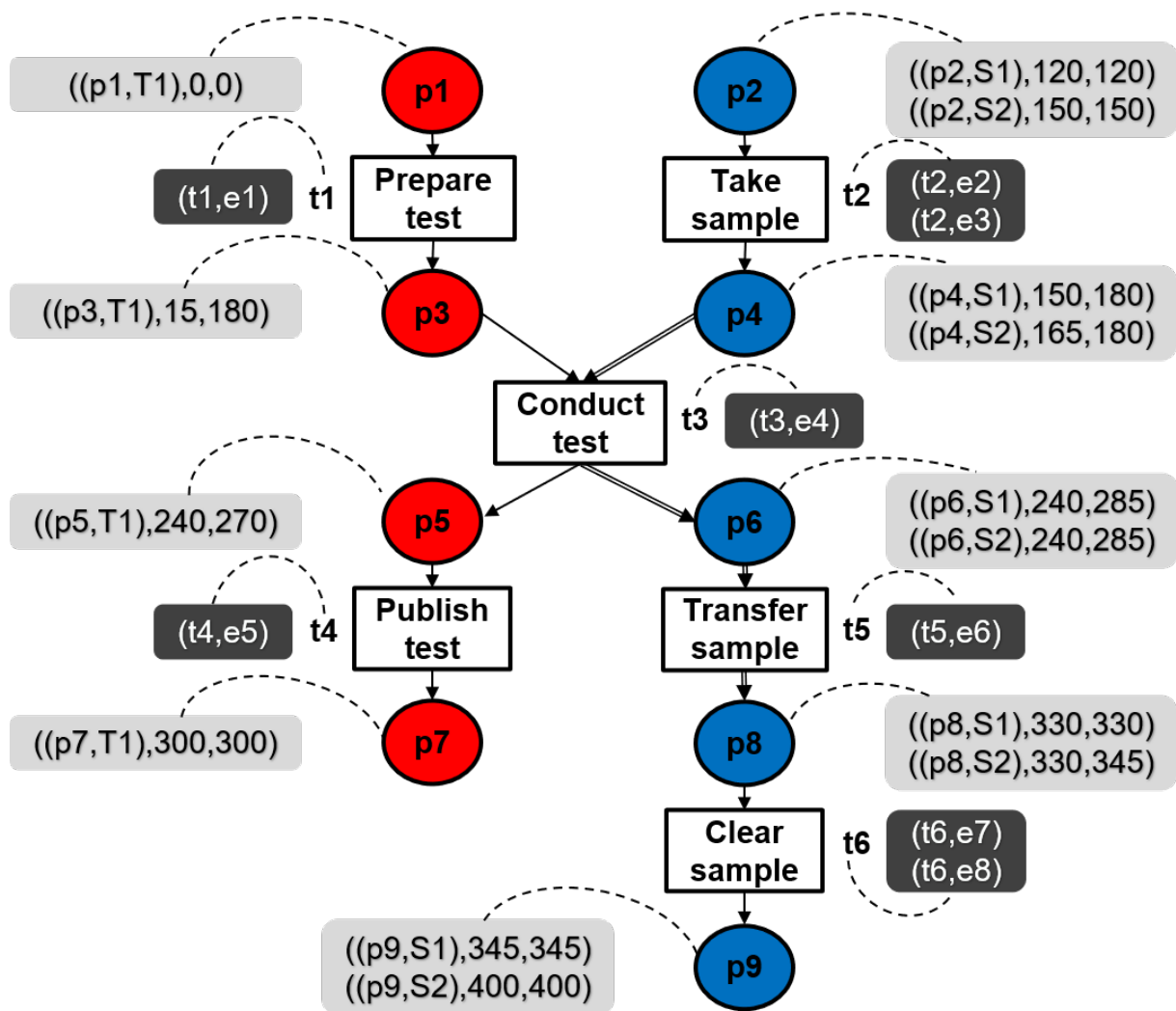
xác mối quan hệ giữa các sự kiện. Mục đích của bước này chính là tính toán ra được các kết quả trung gian liên quan đến chỉ số hiệu năng (Hình [3.2]), bao gồm các lần xuất hiện sự kiện (event occurrences) và các lượt truy cập bằng mã thông báo (token visits). Trong đó:

- Các lần xuất hiện sự kiện (event occurrences) cho biết sự xuất hiện của một sự kiện (event) liên quan đến mỗi lần chuyển đổi (transition) được hiển thị trên Object-Centric Petri Net.
- Các lượt truy cập bằng mã thông báo (token visits) cho biết việc mã thông báo (token) truy cập tới một vị trí (place) tương ứng, được thực hiện trong mô hình quy trình.

Trong bước này, hàm phát lại (replay function) đơn giản là dùng để ánh xạ hành vi của việc thực thi quy trình (ở đây là dữ liệu event log) với hành vi được cho phép của mô hình quy trình (tập hợp các lần xuất hiện sự kiện (event occurrences) và các lượt truy cập bằng mã thông báo (token visits)).

Hình [3.3] minh họa cho kết quả của quá trình phát lại (replay) các sự kiện trong dữ liệu event log của một quy trình xét nghiệm máu [1.1]. Các ô màu xám đậm biểu thị các lần xuất hiện sự kiện (event occurrences) O_1 và các ô màu xám nhạt biểu thị các lượt truy cập mã thông báo (token visits) V_1 . Ví dụ, việc phát lại sự kiện $e1$ và $e4$ trong 1.1 tạo ra các lần xuất hiện sự kiện $(t1, e1)$ và $(t3, e4)$ tương ứng và lượt truy cập mã thông báo $((p3, T1), 15, 180)$, trong đó 15 là thời điểm khi $e1$ hoàn thành và 180 là thời điểm khi $e4$ bắt đầu.

Trong bước này, cần cụ thể hóa hàm phát lại (replay function) bằng phương pháp phát lại dựa trên token được mô tả trong nghiên cứu “A Novel Token-Based Replay Technique to Speed Up Conformance Checking and Process Enhancement” [3]. Bước đầu tiên là làm phẳng (flatting) một OCEL thành một dữ liệu event log truyền thống và chiếu một OCPN thành một Petri Net cho mỗi loại đối tượng. Tiếp theo, cần áp dụng phát lại dựa trên token cho mỗi dữ liệu event log và Petri Net, như được giới thiệu trong



Hình 3.3: Ví dụ về phát lại (replay) dữ liệu event log lên mô hình quy trình OCPN.

[6]. Hàm phát lại cần được cụ thể hóa để bỏ qua các sự kiện không phù hợp để xử lý các dữ liệu event log có độ phù hợp không hoàn hảo. Để đơn giản hóa vấn đề, chúng em giả định rằng các dữ liệu event log đã được làm phẳng phù hợp hoàn toàn với các Petri Net được chiếu (tức là không có token bị thiếu hoặc còn lại).

3.3.3 Bước 3 - Tính các chỉ số hiệu năng từ các kết quả trung gian và mô hình quy trình

Trong bước này, chúng em sẽ tính toán các chỉ số hiệu năng có xem xét nhiều đối tượng cho mỗi lần xuất hiện sự kiện (event occurrence). Một số cách tính chỉ số hiệu năng được sử dụng có thể kể đến là sử dụng các biện pháp đã có và sử dụng các biện pháp mới có xem xét sự tương tác giữa các đối tượng. Hình [3.4] trực quan hóa các thước đo hiệu năng hiện có và thường được sử dụng. Trong đó có:

- Độ đo hiệu năng theo thời gian, bao gồm 7 độ đo giúp có được nhưng góc nhìn khác nhau về hiệu năng của quy trình và được tính toán bằng các phép tính riêng biệt. Đây cũng là các độ đo hiệu năng hiện có, liên quan đến thử nghiệm và được sử dụng trong khóa luận này.
 1. Thời gian chờ (Waiting Time)
 2. Thời gian thực thi (Service Time)
 3. Thời gian lưu trú (Sojourn Time)
 4. Thời gian gộp (Pooling Time)
 5. Thời gian trễ (Lagging Time)
 6. Thời gian đồng bộ hóa (Synchronization Time)
 7. Thời gian luồng (Flow Time)
- Độ đo hiệu năng phi thời gian, bao gồm 2 độ đo giúp cung cấp những thông tin quan trọng, giúp hiểu sâu hơn về cấu trúc, xu hướng và biểu hiện của dữ liệu. Các độ đo này là:
 1. Tần suất đối tượng (Object Frequency): cho biết số lần xuất hiện của một đối tượng cụ thể trong một tập dữ liệu, mục đích chính dùng để xác định sự phổ biến của một đối tượng.

2. Tần suất loại đối tượng (Object Type Frequency): cho biết số lần xuất hiện của các loại đối tượng khác nhau trong một tập dữ liệu, mục đích chính dùng để so sánh mức độ phân bố của các loại đối tượng.

Các độ đo truyền thống như: thời gian chờ (Waiting Time), thời gian thực thi (Service Time), thời gian lưu trú (Sojourn Time) đã được trình bày tại chương [2]. Các độ đo mới như: thời gian luồng (Flow Time), thời gian đồng bộ hóa (Synchronization Time), thời gian gộp (Pooling Time), thời gian trễ (Lagging Time) là những độ đo tập trung vào các đối tượng cụ thể và mối quan hệ giữa chúng. Để tính các chỉ số hiệu năng cần dựa trên các kết quả trung gian và mô hình quy trình dạng “Object-centric Petri Net” [6]. Các kết quả trung gian này bao gồm các lần xuất hiện của sự kiện (event occurrences) và các lượt truy cập bằng mã thông báo (token visits), được ghi nhận lại thông qua quá trình phát lại (replay) ở bước 2 [3.3.2]. Mô hình quy trình có dạng “Object-Centric Petri Net” là kết quả của quá trình khám phá quy trình ở bước 1 [3.3.1].

Để có các chỉ số hiệu năng chính xác, cần tính toán các tính chất cơ bản như: trung bình, trung vị, lớn nhất, nhỏ nhất,... của những độ đo hiệu năng theo thời gian. Để làm được điều này, trước tiên cần liên kết lần xuất hiện sự kiện (event occurrence) với các lượt truy cập mã thông báo (token visits): 1) liên quan đến sự chuyển đổi (transition) của event occurrence và 2) liên quan đến các đối tượng (objects) liên quan đến sự kiện của event occurrence này. Bên cạnh đó, một bước không thể thiếu trước khi tính hiệu năng là liên kết các lần xuất hiện sự kiện (event occurrences) với các lượt truy cập bằng mã thông báo (token visits) - “Relating An Event Occurrence to Token Visits”. Bước này sẽ lấy lượt truy cập bằng mã thông báo (token visit) có thời gian bắt đầu muộn nhất trong số những token liên quan của đối tượng đó làm token visit đại diện để tính toán. Trong ví dụ một quy trình xét nghiệm máu [1.1], dựa vào hình minh họa [3.5] có được tv_1 là lượt truy cập bằng mã thông báo (token visit) muộn nhất của

đối tượng T_1 .



Hình 3.4: Tổng hợp các độ đo hiệu năng.

Việc phân tích chính xác các chỉ số hiệu năng là một việc làm quan trọng để đánh giá và cải thiện hiệu quả hoạt động của quy trình. Mỗi chỉ số này sẽ cung cấp một góc nhìn khác nhau về hiệu năng của quy trình và có các phương pháp tính toán riêng biệt, cụ thể như sau:

1. Thời gian lưu trú (Sojourn Time): là tổng thời gian mà một đối tượng (ở đây sẽ là một mã thông báo) ở trong một địa điểm cụ thể trong hệ thống, từ lúc nó xuất hiện đến khi nó rời khỏi địa điểm đó. Thời gian lưu trú của một mã thông báo trong một địa điểm được tính bằng cách lấy dấu thời gian khi mã thông báo được tiêu thụ trừ đi dấu thời gian khi mã thông báo được tạo ra.
2. Thời gian chờ (Waiting Time): là khoảng thời gian mà một đối tượng phải chờ đợi trước khi được xử lý hoặc thực thi trong hệ thống. Đây là thời gian mà mã thông báo (token) chưa được xử lý dù đã sẵn sàng. Thời gian chờ của một mã thông báo được tính bằng cách lấy dấu thời gian khi mã thông báo (token) thực sự bị tiêu thụ trừ đi dấu thời gian khi chuyển tiếp tiêu thụ mã thông báo được kích hoạt hoàn toàn.
3. Thời gian thực thi (Service Time): là khoảng thời gian mà một đối tượng thực sự được xử lý hoặc thực thi trong hệ thống. Đây là

khoảng thời gian mà mã thông báo đang được hệ thống xử lý. Thời gian thực thi sẽ được tính là khoảng thời gian từ lúc mã thông báo thực sự bị tiêu thụ cho đến khi sự kiện liên quan đến mã thông báo kết thúc.

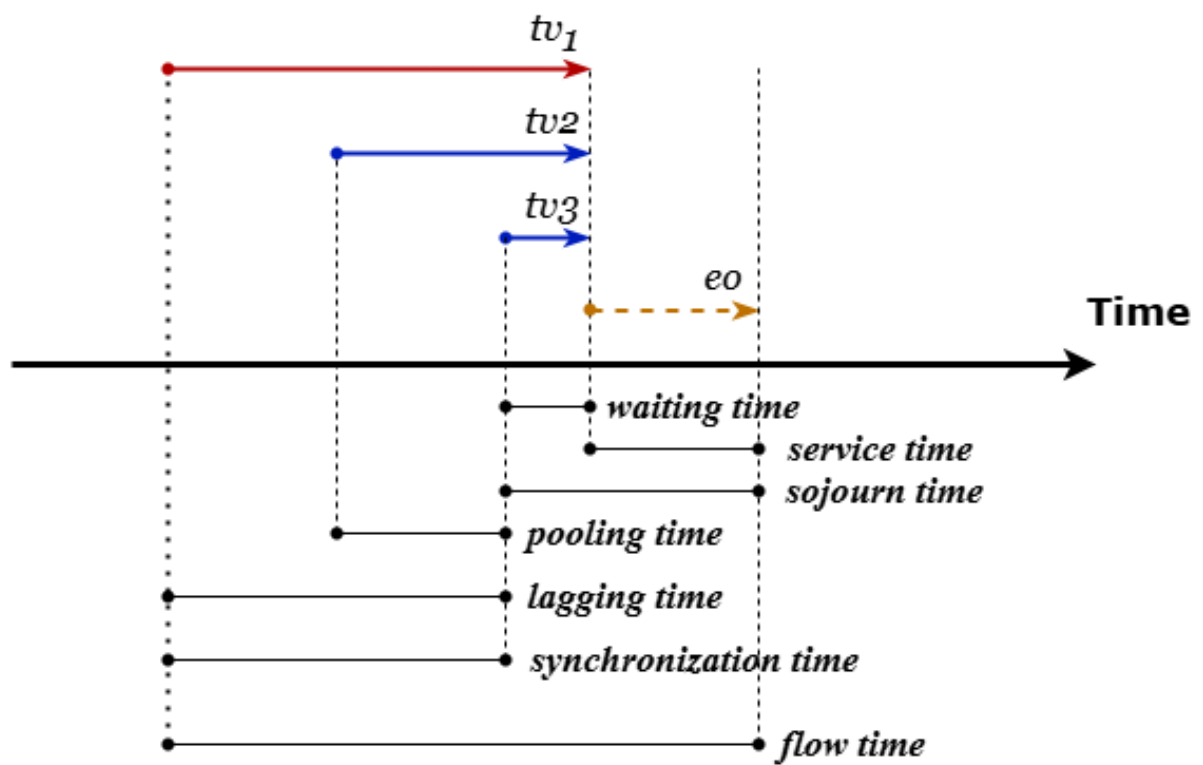
4. Thời gian luồng (Flow Time) là thời gian mà một đối tượng (thường là một đơn vị công việc, sản phẩm hoặc thông tin) mất để đi từ đầu vào đến đầu ra của một quy trình hoặc hệ thống. Thời gian luồng thường bao gồm cả thời gian chờ đợi và thời gian xử lý.
5. Thời gian đồng bộ hóa (Synchronization Time) là thời gian mà một đối tượng (thường là một đơn vị công việc, sản phẩm hoặc thông tin) mất để đi từ đầu vào đến đầu ra của một quy trình hoặc hệ thống. Thời gian luồng thường bao gồm cả thời gian chờ đợi và thời gian xử lý.
6. Thời gian gộp (Pooling Time) là thời gian mà một đối tượng (thường là một đơn vị công việc, sản phẩm hoặc thông tin) mất để đi từ đầu vào đến đầu ra của một quy trình hoặc hệ thống. Thời gian luồng thường bao gồm cả thời gian chờ đợi và thời gian xử lý.
7. Thời gian trễ (Lagging Time) là thời gian mà một đối tượng (thường là một đơn vị công việc, sản phẩm hoặc thông tin) mất để đi từ đầu vào đến đầu ra của một quy trình hoặc hệ thống. Thời gian luồng thường bao gồm cả thời gian chờ đợi và thời gian xử lý.

Sau đây là cách tính để đưa ra được các độ đo hiệu năng của quy trình xét nghiệm máu có đầu vào là dữ liệu event log.

1. Thời gian chờ (Waiting Time) được tính bằng khoảng thời gian từ lúc bắt đầu lượt truy cập bằng mã thông báo muộn nhất liên quan đến sự kiện cho đến lúc sự kiện được bắt đầu.
2. Thời gian thực thi (Service Time) được tính bằng khoảng thời gian

từ lúc bắt đầu sự kiện cho đến lúc sự kiện được hoàn thành (hay kết thúc).

3. Thời gian lưu trú (Sojourn Time) được tính bằng khoảng thời gian từ lượt truy cập bằng mã thông báo muộn nhất liên quan đến sự kiện cho đến lúc sự kiện được hoàn thành. Ngoài ra thời gian lưu trú (Sojourn Time) còn được tính bằng tổng thời gian chờ (Waiting Time) và thời gian thực thi (Service Time).
4. Thời gian gộp (Pooling Time) được tính bằng khoảng thời gian từ lúc bắt đầu lượt truy cập bằng mã thông báo sớm nhất cho đến lúc bắt đầu lượt truy cập bằng mã thông báo muộn nhất liên quan đến sự kiện của một loại đối tượng.
5. Thời gian trễ (Lagging Time) được tính bằng khoảng thời gian từ lúc bắt đầu lượt truy cập bằng mã thông báo sớm nhất của một loại đối tượng cho đến lúc bắt đầu lượt truy cập bằng mã thông báo muộn nhất của một loại đối tượng khác liên quan đến sự kiện.
6. Thời gian đồng bộ hóa (Synchronization Time) được tính bằng khoảng thời gian từ lúc bắt đầu lượt truy cập bằng mã thông báo sớm nhất cho đến lúc bắt đầu lượt truy cập bằng mã thông báo muộn nhất liên quan đến sự kiện. Thời gian đồng bộ hóa (Synchronization Time) bao trùm thời gian gộp (Pooling Time) và thời gian trễ (Lagging Time).
7. Thời gian luồng (Flow Time) được tính bằng khoảng thời gian từ lúc bắt đầu lượt truy cập bằng mã thông báo sớm nhất liên quan đến sự kiện cho đến lúc sự kiện được hoàn thành. Thời gian luồng (Flow Time) còn được tính bằng tổng thời gian đồng bộ hóa (Synchronization Time) và thời gian lưu trú (Sojourn Time).



Hình 3.5: Ví dụ minh họa các độ đo hiệu năng theo thời gian của sự kiện *Conduct Test* trong một quy trình xét nghiệm máu.

Chương 4

Thí nghiệm

Trong chương này, chúng em sẽ trình bày các thí nghiệm để đánh giá mức độ hiệu quả của phương pháp xem xét nhiều đối tượng được đề cập ở trên trong việc phân tích hiệu năng quy trình ở các bộ dữ liệu thực tế. Đầu tiên, chúng em sẽ nói về cách thiết lập các thí nghiệm, bao gồm: mô tả tập dữ liệu và trình bày về việc rút trích dữ liệu từ tập dữ liệu ban đầu phục vụ cho thí nghiệm. Ở thí nghiệm đầu tiên, chúng em sẽ so sánh kết quả cài đặt của khóa luận với bài báo gốc.

4.1 Các thiết lập thí nghiệm

4.1.1 Tập dữ liệu

Để thực hiện các thí nghiệm trong khóa luận, chúng em đã sử dụng bộ dữ liệu được cung cấp bởi Hội thảo Quốc tế về Business Process Intelligence (BPI) lần thứ 13, diễn ra vào năm 2017. Bộ dữ liệu này đã được công bố trong khuôn khổ của hội thảo và được sử dụng rất rộng rãi trong cộng đồng nghiên cứu về các bài toán phân tích khai thác quy trình kinh doanh. Bộ dữ liệu trên là một bộ dữ liệu sự kiện log, nội dung của bộ dữ liệu là thông tin về quy trình đăng ký khoản vay của một viện tài chính Hà Lan. Dữ liệu được ghi nhận trong khoảng thời gian từ tháng 01 năm 2016 đến

tháng 02 năm 2017.

Bộ dữ liệu gốc được cung cấp bởi Hội thảo Quốc tế về Business Process Intelligence (BPI) lần thứ 13 về quy trình đăng ký khoản vay sẽ có tổng cộng 393,931 dòng dữ liệu tương ứng với 393,931 sự kiện được ghi lại với 23 cột thuộc tính giá trị khác nhau cùng lưu trữ thông tin về quy trình đăng ký khoản vay. Mỗi sự kiện trong bộ dữ liệu sẽ được ghi lại khi một sự kiện trong quy trình được hoàn thành với các thông tin chính như sau:

- Quy trình sẽ bao gồm hai loại đối tượng chính: đơn đăng ký (application) và đề xuất (offer). Một đơn đăng ký có thể bao gồm một hoặc nhiều đề xuất kèm theo.
- Quy trình sẽ bắt đầu khi khách hàng tạo ra một đơn đăng ký, có thể thực hiện bằng cách trực tiếp đến ngân hàng hoặc sử dụng hệ thống trực tuyến của ngân hàng. Trong trường hợp khách hàng đến trực tiếp ngân hàng, bước nộp đơn đăng ký qua hệ thống trực tuyến sẽ được bỏ qua. Sau khi đơn đăng ký được hoàn thành và chấp nhận bởi ngân hàng, ngân hàng sẽ tiến hành cung cấp các đề xuất về khoản vay cho khách hàng. Điều này được thực hiện bằng cách gửi các đề xuất vay đến khách hàng qua hệ thống hoặc qua cuộc gọi điện thoại. Mỗi đề xuất được gửi đi có thể được khách hàng chấp nhận hoặc từ chối. Nếu khách hàng chấp nhận đề xuất vay, quá trình vay sẽ tiếp tục với các bước tiếp theo trong quy trình. Nếu đề xuất bị từ chối, nó sẽ bị hủy bỏ và không còn hiệu lực.
- Thông tin đầu tiên được ghi nhận là thông tin định danh cho sự kiện được hoàn thành trong quy trình xử lý đơn vay, cụ thể ở đây sẽ là 'event id' của sự kiện.
- Tiếp theo là thông tin về hoạt động cụ thể trong quá trình xử lý đơn vay, mỗi một hoạt động đại diện cho một bước trong quy trình xử lý.
- Thông tin về thời gian bắt đầu và kết thúc của sự kiện. Thời gian

được lưu với định dạng (YYYY/MM/DD HH:MM:SS), giúp theo dõi chính xác thời điểm các sự kiện diễn ra.

- Ngoài ra sẽ còn các thông tin khác như mục tiêu vay, loại đơn vay, số tiền được vay, kỳ hạn cho vay,...

4.1.2 Rút trích dữ liệu

Trong phần này, chúng em sẽ trình bày về nguyên nhân, mục đích của việc rút trích dữ liệu cũng như cách mà nhóm tác giả của bài báo “OPerA: Object-Centric Performance Analysis” [6] đã làm để có thể rút trích dữ liệu từ bộ dữ liệu gốc để phục vụ cho việc tiến hành các thí nghiệm.

Nhóm tác giả đã tập trung vào việc phân tích các đơn vay bị hủy. Mục đích của việc làm này là nhằm cải thiện quy trình xử lý các đơn đăng ký khoản vay và nâng cao hiệu suất hoạt động của ngân hàng. Việc phân tích các đơn vay bị hủy mang lại nhiều lợi ích quan trọng. Trước hết, nó giúp xác định rõ các nguyên nhân cụ thể dẫn đến quyết định hủy đơn vay, chẳng hạn như điều kiện tài chính của khách hàng không đủ đáp ứng, thông tin cung cấp không đầy đủ hoặc sai lệch, và sự thay đổi trong yêu cầu của khách hàng. Thứ hai, phân tích này cho phép phát hiện các vấn đề trong quy trình xử lý vay, từ đó nhận diện các bước có thể gây ra sự chậm trễ hoặc khó khăn, giúp ngân hàng có thể đưa ra các biện pháp cải thiện quy trình hiệu quả hơn. Thứ ba, việc tập trung vào các đơn vay bị hủy cũng giúp tối ưu hóa quy trình xử lý vay, giảm thiểu tỉ lệ hủy và cải thiện trải nghiệm khách hàng.

Nhóm tác giả đã sử dụng một phương pháp tiếp cận chi tiết và hệ thống để phân tích các đơn vay bị hủy. Đầu tiên, họ lọc ra các hành vi thường xuyên nhất liên quan đến các đơn vay bị hủy, bằng cách chọn ra mười loại hành vi phổ biến nhất. Điều này giúp tập trung phân tích vào các mẫu hành vi có tần suất cao, từ đó dễ dàng nhận diện các mẫu và xu hướng chung. Tiếp theo, các hoạt động thừa thãi như các cập nhật trạng thái không cần thiết (ví dụ: trạng thái "Completed" sau khi hoàn thành đơn

vay) đã được loại bỏ. Việc này không chỉ giúp làm sạch dữ liệu mà còn tập trung vào các bước quan trọng trong quy trình, đảm bảo rằng phân tích tập trung vào những khía cạnh có ý nghĩa nhất của quy trình.

Sau khi thực hiện quá trình rút trích dữ liệu từ bộ dữ liệu gốc, chúng ta sẽ thu được một bộ dữ liệu mới gồm tổng cộng 20,478 sự kiện. Trong đó, có 1,682 đơn đăng ký (applications) và 3,573 đề xuất (offers).

4.2 Thí nghiệm 1 - So sánh kết quả cài đặt của khóa luận với bài báo gốc

Ở thí nghiệm đầu tiên, chúng em sẽ sử dụng phương pháp xem xét nhiều đối tượng ở chương 3 để tiến hành thử nghiệm trên tập dữ liệu “BPI 2017” được giới thiệu ở phần 4.1.1. Sau đó chúng em sẽ sử dụng kết quả có được để so sánh với kết quả trong bài báo gốc [6].

Với hành động *Send*, ta có thể thấy ở hình [4.2] rằng phương pháp truyền thống cho rằng có đến 1.799 đơn đăng ký thực hiện lặp lại hành động này. Tuy nhiên, thực tế là hành động này không hề lặp lại; mỗi đơn đăng ký chỉ thực hiện hành động *Send* một lần duy nhất. Điều này được thể hiện chính xác trong kết quả của tác giả và của cài đặt khóa luận, như được minh họa ở hình [4.3] và [4.1], khi sử dụng phương pháp xem xét nhiều đối tượng. Ngoài ra, thời gian lưu trữ trung bình cho hành động *Send* theo phương pháp truyền thống ở hình [4.2] được tính toán là khoảng 2 ngày và 23 giờ. Tuy nhiên, khi xem xét nhiều đối tượng, thời gian lưu trữ thực tế chỉ khoảng 15 phút, như kết quả ở hình [4.1] và [4.3] cho thấy. Điều này làm rõ sự khác biệt giữa phương pháp truyền thống và phương pháp xem xét nhiều đối tượng trong việc đánh giá hiệu năng của hành động *Send*.

Tiếp tục với hành động *CancelApplication*, ta thấy rằng phương pháp truyền thống, như được minh họa ở hình [4.2], cho rằng hành động này được lặp lại 1.891 lần. Tuy nhiên, trên thực tế, hành động này chỉ xảy ra

1.682 lần cho mỗi đơn đăng ký, như kết quả được thể hiện ở hình [4.1] và [4.3]. Thêm vào đó, thời gian lưu trú trung bình cho hoạt động này được đo lường là khoảng 12 ngày và 22 giờ. Nhưng theo tác giả thì thực tế kết quả cho thời gian lưu trú trung bình cho hoạt động *CancelApplication* sẽ là khoảng 31 ngày 22 giờ thể hiện ở hình [4.3], tuy nhiên với cài đặt khóa luận như ở hình [4.1] thì thời gian lưu trú trung bình chỉ rơi vào khoảng 27 ngày 11 giờ cho hành động này. Về sự khác biệt này, chúng em đã tìm hiểu và nhận ra rằng trong phần cài đặt của tác giả, tác giả đã có sự nhầm lẫn khi sử dụng lượt truy cập bằng mã thông báo (token visit) sớm nhất để tính toán cho độ đo này trong khi theo lý thuyết thì chúng ta phải sử dụng lượt truy cập bằng mã thông báo (token visit) muộn nhất. Mặc dù có sự nhầm lẫn nhưng với hành động *Send* thì tác giả vẫn cho ra kết quả đúng với thời gian lưu trú trung bình vì ở hành động *Send* chỉ có sự tham gia của một đối tượng duy nhất là Đề xuất (Offer) nên nó không ảnh hưởng đến việc phân biệt lượt truy cập bằng mã thông báo (token visit) muộn nhất hay sớm nhất tại hành động này. Tuy nhiên ở hành động *CancelApplication* thì sự tham gia của cả hai loại đối tượng Đơn đăng kí (Application) và Đề xuất (Offer) nên dẫn đến kết quả tính toán sẽ bị chênh lệch, thay đổi. Thế nên chúng em đã chỉnh sửa lại chỗ nhầm lẫn đó để kết quả được chính xác và phù hợp với lý thuyết. Sau khi chỉnh sửa, chúng em có cho ra kết quả chính xác cho thời gian lưu trú trung bình của hoạt động *CancelApplication* sẽ là khoảng 27 ngày 11 giờ. Tiếp theo, chúng ta sẽ phân tích các chỉ số hiệu năng mới theo cách xem xét nhiều đối tượng. Các chỉ số này bao gồm thời gian đồng bộ hóa, thời gian trễ và thời gian gộp. Những chỉ số này không thể tính toán được bằng phương pháp truyền thống do thiếu dữ kiện thông tin.

Theo kết quả của tác giả ở hình [4.3], thời gian đồng bộ hóa trung bình cho hành động *CancelApplication* là khoảng 4 ngày 11 giờ. Trong khi đó, kết quả cài đặt khóa luận ở hình [4.1] của chúng em cho thấy thời gian đồng bộ hóa trung bình là khoảng 4 ngày 8 giờ. Để hiểu rõ sự khác biệt này, chúng em đã tìm hiểu chi tiết các phương pháp tính toán được sử

dụng trong cả hai trường hợp.

- Với cài đặt của tác giả đã sử dụng dấu thời gian (timestamps) của các sự kiện đầu vào liên quan đến hành động *CancelApplication* để tính toán thời gian đồng bộ hóa. Cụ thể, tác giả xác định thời gian đồng bộ hóa bằng cách lấy khoảng chênh lệch giữa dấu thời gian lớn nhất và dấu thời gian nhỏ nhất trong các sự kiện đầu vào. Phương pháp này dựa trên giả định rằng tất cả các sự kiện đầu vào đều được xử lý cùng lúc.
- Ngược lại, trong cài đặt của chúng em, chúng em sử dụng các lượt truy cập bằng mã thông báo (token visits) để tính toán thời gian đồng bộ hóa. Điều này có nghĩa là chúng em xem xét thời gian từ khi mã thông báo được tạo ra cho đến khi nó được tiêu thụ. Phương pháp này cho phép chúng em tính toán thời gian đồng bộ hóa một cách chính xác hơn, vì nó phản ánh đúng quá trình di chuyển của mã thông báo qua các trạng thái khác nhau trong hệ thống.

Sự khác biệt trong phương pháp tính toán đã dẫn đến sự chênh lệch giữa hai kết quả. Tuy nhiên, mặc dù có sự khác biệt nhỏ, cả hai kết quả đều cho thấy một mức độ tương đồng nhất định và có thể chấp nhận được. Điều này cho thấy rằng dù sử dụng phương pháp nào, thời gian đồng bộ hóa của hành động *CancelApplication* vẫn nằm trong một khoảng thời gian hợp lý.

Tiếp tục với hành động *CancelApplication*, theo kết quả nghiên cứu của tác giả được trình bày ở hình [4.3], thời gian trễ trung bình của các đơn đăng ký (Applications) là 3 ngày và 15 giờ, trong khi thời gian trễ trung bình của các đề nghị (Offers) là khoảng 19 giờ. Tuy nhiên, theo kết quả cài đặt của khóa luận ở hình [4.1], thời gian trễ trung bình của các đơn đăng ký (Applications) là 4 ngày và 8 giờ, và thời gian trễ trung bình của các đề nghị (Offers) chỉ là khoảng 22 phút.

Để hiểu rõ hơn về thời gian trễ, chúng ta có thể tham khảo định nghĩa và công thức tính được đề cập trong bài báo [6]. Tác giả đã đưa ra định

nghĩa và công thức tính như sau: $lag_{ot} \in U_m$ tính toán thời gian trễ liên quan đến loại đối tượng ot , là khoảng thời gian chênh lệch giữa lượt truy cập mã thông báo muộn nhất của ot và lượt truy cập mã thông báo sớm nhất của các loại đối tượng khác liên quan đến sự kiện. Theo công thức này, với mỗi $eo = (t, e) \in O$, $lag_{ot}(eo, V) = \max(T') - \min(T)$ trong đó $T = \{\pi_{bt}(tv) \mid tv \in rel_{ON}(eo, V)\}$ và $T' = \{\pi_{bt}(tv) \mid tv \in rel_{ON}(eo, V) \wedge type(\pi_{oi}(tv)) \neq ot\}$ nếu $\max(T') > \min(T)$, và bằng 0 nếu ngược lại. Theo công thức được đề cập của tác giả thì:

- T' là tập thời gian bắt đầu của các lượt truy cập mã thông báo (token visit) (liên quan đến eo đang xét) mà có loại đối tượng khác ot .
- T là tập begin time của các các lượt truy cập mã thông báo (token visit) (liên quan đến eo đang xét) mà có loại đối tượng bất kỳ.

Tuy nhiên, theo lời giải thích của tác giả, có sự nhầm lẫn trong việc xác định T' và T . Điều này xuất hiện khi tính toán $\max(T')$ tượng trưng cho thời gian bắt đầu của lượt truy cập mã thông báo (token visit) muộn nhất của đối tượng ot lại không có lượt truy cập mã thông báo (token visit) của đối tượng ot , trong khi T lại có lượt truy cập mã thông báo (token visit) của tất cả các đối tượng bao gồm cả đối tượng ot . Do đó, công thức chính xác cho phần này sẽ là:

- T' là tập thời gian bắt đầu của các lượt truy cập mã thông báo (token visit) (liên quan đến eo đang xét) mà có loại đối tượng là ot .
- T là tập thời gian bắt đầu của các lượt truy cập mã thông báo (token visit) (liên quan đến eo đang xét) mà có loại đối tượng bất kỳ khác ot .

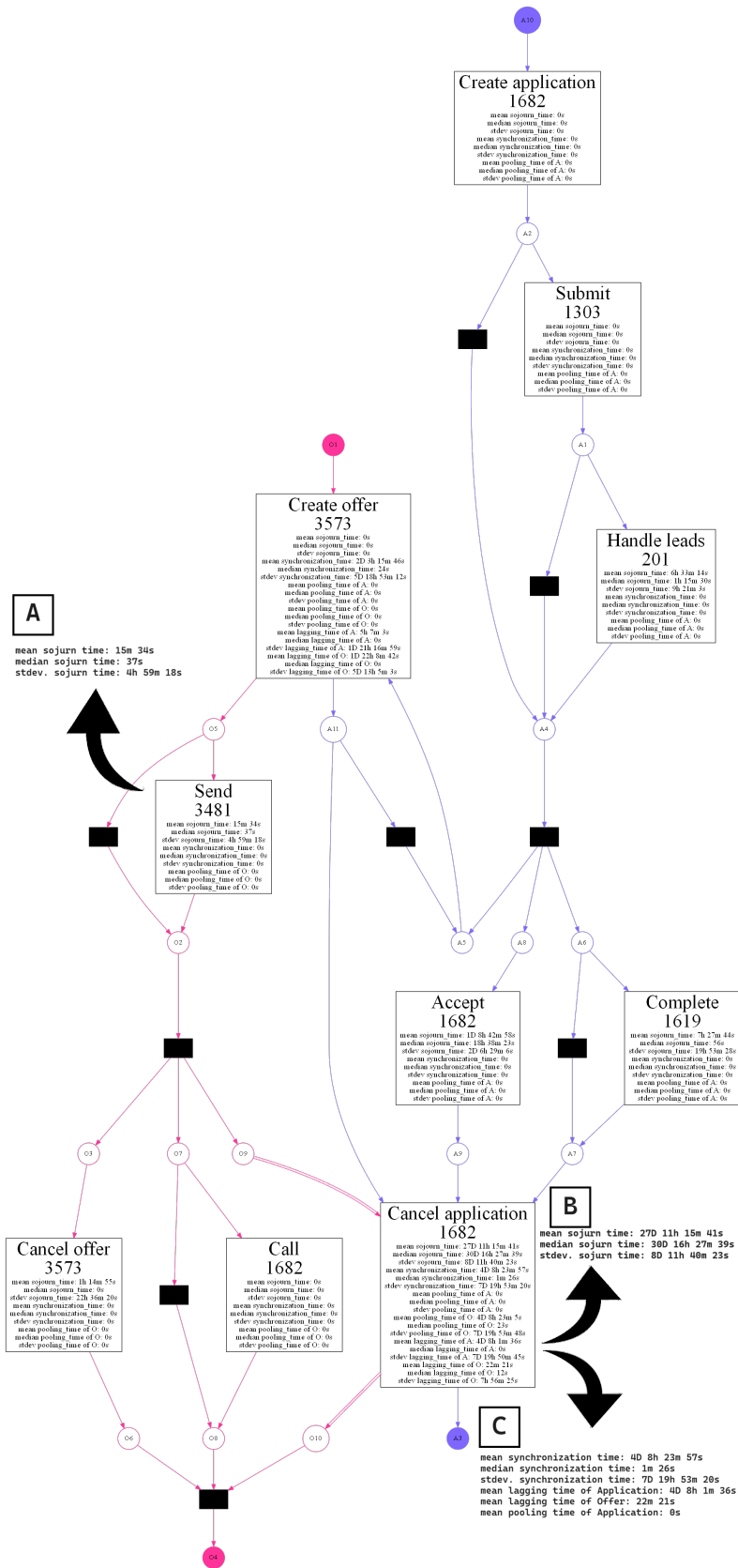
Thêm vào đó, khi tính thời gian trễ cho loại đối tượng ot , nếu kết quả bé hơn 0 thì chúng ta sẽ mặc định kết quả sẽ là 0, tuy nhiên trong cài đặt của mình thì tác giả lại cài đặt cho trường hợp này sẽ không tồn tại thời gian trễ cho loại đối tượng ot . Điều này cũng dẫn đến sự sai lệch khi tính

giá trị trung bình cho thời gian trễ. Trong quá trình nghiên cứu và cài đặt cho khóa luận, chúng em đã sửa đổi và cải thiện những sai sót và nhầm lẫn này để đảm bảo tính chính xác và thống nhất cho các kết quả. Vì vậy, theo kết quả cài đặt của khóa luận ở hình [4.1], thời gian trễ trung bình của các đơn đăng kí (Application) là 4 ngày và 8 giờ được xác định chính xác và phù hợp với lý thuyết, trong khi thời gian trễ trung bình của các đề xuất (Offer) là khoảng 22 phút. Ta có thể thấy các đề xuất (Offer) có độ trễ nghiêm trọng hơn so với các đơn đăng kí (Application).

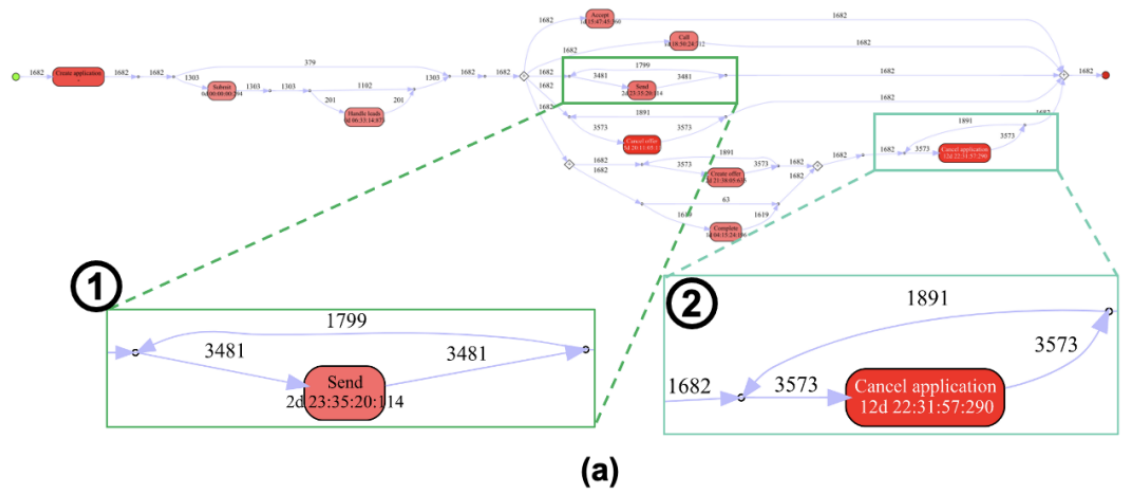
Ngoài ra cũng ở hành động *CancelApplication*, theo tác giả, thời gian gộp trung bình cho các đơn đăng kí (Application) là khoảng 4 ngày 11 giờ. Tuy nhiên, như chúng em đã mô tả và đề cập về tập dữ liệu ở phần 4.1.1, mỗi quy trình chỉ thực hiện với một đơn đăng kí (Application) duy nhất và có thể kèm theo một hoặc nhiều đề xuất (Offer) khác nhau. Do đó, với việc mỗi quy trình chỉ có một đối tượng đơn đăng kí, thời gian gộp cho đối tượng này sẽ luôn luôn bằng 0 trong mọi trường hợp. Kết quả mà tác giả đưa ra về thời gian gộp trung bình cho đối tượng đơn đăng kí (Application) là khoảng 4 ngày 11 giờ là không chính xác, vì nó không phù hợp với lý thuyết. Theo kết quả cài đặt của khóa luận ở hình [4.1], thời gian gộp trung bình cho các đơn đăng kí (Application) sẽ là 0 và thời gian gộp trung bình cho các đề xuất (Offer) sẽ là khoảng 4 ngày 8 giờ. Hơn nữa, thời gian gộp trung bình cho các đề xuất gần như tương đồng với thời gian đồng bộ hóa trung bình, điều này cho thấy rằng các đơn đăng kí đã sẵn sàng để hủy gần như cùng lúc với các đề xuất đầu tiên và đề xuất thứ hai đã sẵn sàng trong khoảng 4 ngày 11 giờ.

4.3 Thí nghiệm 2 - Cải thiện kết quả của phương pháp xem xét nhiều đối tượng ở thí nghiệm 1

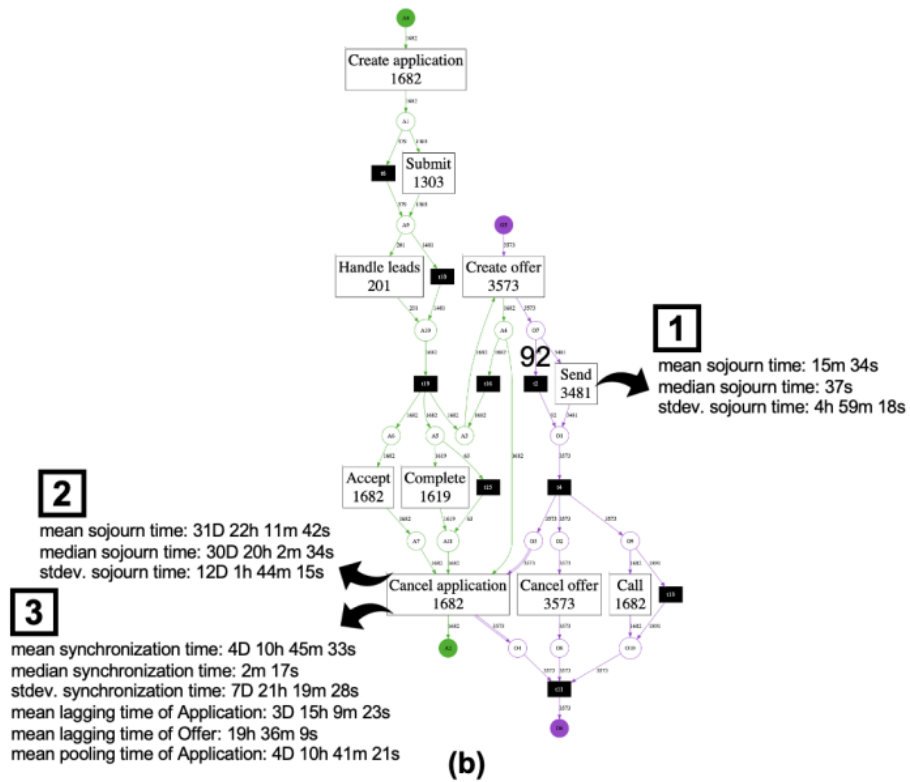
Sau khi thực hiện xong thí nghiệm 1, chúng em nhận thấy rằng mô hình quy trình có một số điểm chưa hợp lý. Đầu tiên, đối với các hoạt động *Accept*, *Complete* và *CreateOffer*, mô hình quy trình hiện tại cho rằng cả ba hoạt động này có thể thực hiện song song với nhau mà không cần tuân theo thứ tự nào. Tuy nhiên, trong thực tế và theo dữ liệu event log, các hoạt động này phải tuân theo một thứ tự cụ thể. Cụ thể, hoạt động *CreateOffer* cần phải diễn ra trước tiên, tạo ra một đề xuất (Offer). Sau đó, hoạt động *Accept* mới có thể diễn ra để chấp nhận đề xuất này. Cuối cùng, sau khi đề xuất đã được chấp nhận, hoạt động *Complete* mới có thể được thực hiện để hoàn thành quy trình. Việc mô hình hiện tại không thể hiện đúng thứ tự này dẫn đến sự không chính xác trong việc mô phỏng quy trình thực tế. Nhằm khắc phục vấn đề này và cải thiện kết quả từ thí nghiệm 1, chúng em sẽ tiến hành thí nghiệm 2. Thí nghiệm này sẽ tập trung vào việc điều chỉnh mô hình quy trình để phản ánh đúng thứ tự thực hiện của các hoạt động, từ đó đưa ra những phân tích mới và chính xác hơn về quy trình.



Hình 4.1: Kết quả phân tích hiệu năng quy trình dựa trên cài đặt khóa luận khi chạy thử nghiệm trên tập dữ liệu “BPI 2017” .



Hình 4.2: Kết quả phân tích hiệu năng quy trình của tác giả tập dữ liệu “BPI 2017” khi áp dụng phương pháp truyền thống .



Hình 4.3: Kết quả phân tích hiệu năng quy trình của tác giả tập dữ liệu “BPI 2017” khi áp dụng phương pháp xem xét nhiều đối tượng .

Chương 5

Tổng kết và hướng phát triển

5.1 Tổng kết

Trong khóa luận này, chúng em đã tìm hiểu và áp dụng một phương pháp mới trong lĩnh vực khai thác quy trình, cụ thể là phương pháp xem xét nhiều đối tượng. Phương pháp này được trình bày chi tiết trong bài báo [6] và nhằm giải quyết bài toán có ý nghĩa quan trọng trong lĩnh vực này đó là bài toán phân tích hiệu năng quy trình từ dữ liệu đặc biệt là các quy trình phức tạp. Phương pháp này có nhiều ưu điểm như:

- **Tính chính xác và linh hoạt:** Phương pháp xem xét nhiều đối tượng cho phép tính toán chính xác các chỉ số hiệu năng trong bối cảnh các quy trình phức tạp có sự liên quan đến nhiều đối tượng khác nhau. Cụ thể, phương pháp này sử dụng OCEL để lưu trữ các chuỗi sự kiện chồng chéo nhau mà không cần làm phẳng chúng thành một chuỗi sự kiện đơn lẻ. Điều này giúp giữ nguyên tính chính xác và toàn vẹn của dữ liệu gốc, từ đó tránh được các sai sót khi tính toán các chỉ số hiệu năng truyền thống như thời gian chờ, thời gian thực thi và thời gian lưu trữ. Khi các sự kiện liên quan đến nhiều loại đối tượng khác nhau, phương pháp này còn đảm bảo rằng các kết quả phân tích phản ánh chính xác hoạt động thực tế của quy trình, điều mà các phương pháp truyền thống gặp khó khăn trong

việc thực hiện.

- **Hỗ trợ các độ đo hiệu năng mới:** Phương pháp này không chỉ giúp tính toán chính xác các chỉ số hiệu suất truyền thống mà còn hỗ trợ các chỉ số hiệu suất mới dựa trên sự tương tác giữa các đối tượng. Các chỉ số mới này bao gồm thời gian đồng bộ hóa (synchronization time), thời gian gộp (pooling time), thời gian trễ (lagging time), thời gian luồng (flow time),... Những độ đo này cung cấp một cái nhìn sâu sắc hơn về cách các đối tượng tương tác và ảnh hưởng lẫn nhau trong quá trình, giúp phát hiện và giải quyết các vấn đề tiềm ẩn một cách chính xác, hiệu quả hơn đặc biệt là trong việc tối ưu hóa các quy trình.

Ngoài những ưu điểm trên, phương pháp xem xét nhiều đối tượng cũng tồn tại những hạn chế nhất định:

- **Phụ thuộc vào chất lượng của mô hình quy trình:** Phương pháp này phụ thuộc rất nhiều vào các mô hình quy trình được khám phá từ dữ liệu event log đầu vào. Nếu mô hình quy trình không phản ánh chính xác thực tế thì các phân tích hiệu năng dựa trên mô hình đó cũng sẽ không chính xác.
- **Chất lượng của dữ liệu đầu vào:** Khi các sự kiện trong event log bị thiếu, mất hoặc có thông tin không chính xác, dữ liệu sẽ không phù hợp hoặc không tương thích hoàn toàn với quy trình mà mô hình đã mô tả. Điều này dẫn đến tính toán sai lệch các chỉ số hiệu năng, ảnh hưởng đến kết quả phân tích và đánh giá không chính xác về hiệu năng thực tế của quy trình. Những sai lệch này có thể ảnh hưởng lớn đến quyết định và các cải tiến tiếp theo đối với quy trình kinh doanh.

5.2 Hướng phát triển

Trong tương lai, phương pháp xem xét nhiều đối tượng có thể được mở rộng để hỗ trợ phân tích hiệu năng ngay cả khi dữ liệu event log không tuân thủ các quy tắc nhất định, đảm bảo rằng kết quả phân tích vẫn chính xác và đáng tin cậy. Phương pháp này cũng sẽ phát triển để phân tích hiệu năng tập trung vào từng đối tượng dựa trên dữ liệu event log mà không cần phụ thuộc quá nhiều vào các mô hình quy trình, giúp giảm bớt những hạn chế liên quan đến chất lượng của mô hình quy trình. Cuối cùng là sẽ xây dựng và tính toán các chỉ số hiệu năng mới, thú vị hơn, xem xét sự tương tác giữa các đối tượng để cung cấp các phân tích sâu sắc và toàn diện hơn về hiệu năng của quy trình.

Tài liệu tham khảo

Tiếng Anh

- [1] Aalst, Wil M. P. van der and Berti, Alessandro. “Discovering Object-Centric Petri Nets”. In: *Fundamenta Informaticae* (2020).
- [2] Adams, Jan Niklas et al. “Defining Cases and Variants for Object-Centric Event Data”. In: *International Conference on Process Mining* (2022).
- [3] Berti, Alessandro and Aalst, Wil van der. “A Novel Token-Based Replay Technique to Speed Up Conformance Checking and Process Enhancement”. In: *Part of the book series: Lecture Notes in Computer Science ((TOPNOC, volume 12530))* (2021).
- [4] Berti, Alessandro et al. “OCEL (Object-Centric Event Log) 2.0 Specification”. In: *Official website of OCEL Standard* (2023).
- [5] Hornix, P. T. G. et al. “Performance Analysis of Business Processes through Process Mining”. In: *Master’s thesis of P.T.G Hornix at Eindhoven University of Technology* (2007).
- [6] Park, Gyunam, Adams, Jan Niklas, and Aalst, Wil M. P. van der. “OPerA: Object-Centric Performance Analysis”. In: *Springer International Publishing* (2022).