

MỘT PHƯƠNG PHÁP TÍNH MA TRẬN TỐC ĐỘ CHUYỂN TRẠNG THÁI VỚI BÀI TOÁN PHÂN CHIA MẠNG TRONG MẠNG LỖI 5G SỬ DỤNG MÔ HÌNH HÀNG ĐỢI

Đặng Thanh Chương, Hoàng Công Hoàng, Hoa Lý Cương

Khoa Công nghệ Thông tin, Trường Đại học Khoa học, Đại học Huế

Email: dtchuong@hueuni.edu.vn, hoangconghoang511@gmail.com

Ngày nhận bài: 18/01/2024; ngày hoàn thành phản biện: 01/02/2024; ngày duyệt đăng: 5/3/2024

TÓM TẮT

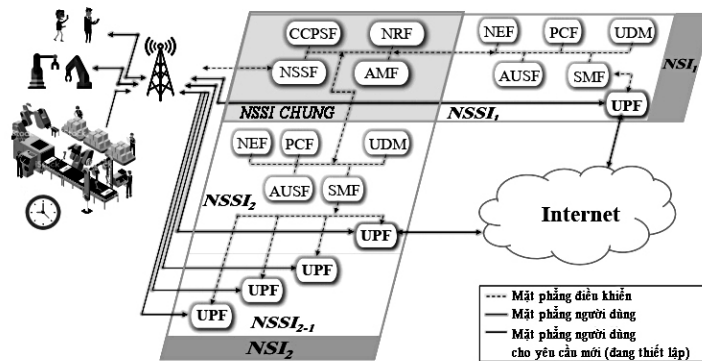
Trong 5G, các chức năng mạng có thể được phân chia tỉ lệ hay thay đổi quy mô (scaled out/in) một cách linh hoạt để điều chỉnh dung lượng cho các lát cắt mạng (network slices). Quy trình tăng/giảm quy mô, cụ thể là tự động thay đổi quy mô, nâng cao hiệu suất bằng cách mở rộng quy mô các phiên bản và giảm chi phí vận hành bằng cách thay đổi quy mô trong các phiên bản. Trong bài viết này, trước tiên chúng tôi sẽ mô hình hóa hệ thống phân chia mạng cho bài toán đặt trước và thiết lập khối động các tài nguyên ứng với các lát cắt mạng trong mạng lõi 5G theo mô hình hàng đợi Markov [1]. Theo đó, bài viết đề xuất một thuật toán để tính xác suất trạng thái cân bằng dựa trên phương pháp chuyển đổi các phương trình trạng thái cân bằng sang dạng ma trận tổng quát Q , đây cũng được xem là kết quả chính của bài báo.

Từ khóa: Chia mạng, Đặt trước, Thiết lập khối, 5G, Ma trận tốc độ chuyển trạng thái.

1. MỞ ĐẦU

Từ thế hệ đầu tiên (1G) đến 4G, các mạng băng thông rộng đều được 3GPP Release 15 (R15) [1-5] quản lý, tuy nhiên thế hệ 5G được thiết kế không chỉ cho con người mà còn cho tất cả mọi thứ, bao gồm cả các loại máy móc khác nhau. Hơn nữa, các chức năng mạng trong các mạng lõi 5G sẽ được triển khai không chỉ bởi các nhà điều hành mạng. Các doanh nghiệp, nhà máy hoặc cơ quan chính phủ cũng có thể xây dựng các mạng di động riêng của họ để tăng hiệu suất và tăng cường bảo mật. Khi người dùng triển khai một mạng di động riêng, các chức năng mạng về mặt phẳng điều khiển (control plane) có thể vẫn được cung cấp bởi các nhà điều hành. Người dùng chỉ sở hữu các chức năng mặt phẳng người dùng (user plane) theo yêu cầu của họ.

Ảo hóa chức năng mạng (NFV - *network function virtualization*), giúp ảo hóa các nút mạng vật lý thành các chức năng mạng ảo hóa (VNFs - *virtualized network functions*), đã trở thành công nghệ chính cho mạng 5G. Thông qua NFV, các nhà khai thác có thể tự động triển khai các tài nguyên ảo để đạt được hiệu quả chi phí tối ưu. Như được định nghĩa trong 3GPP TS 28.530 [5], một lát cắt mạng bao gồm một số phiên bản lát cắt mạng (NSI - *network slice instances*), có thể được phân chia tỉ lệ hay mở rộng/thu nhỏ quy mô (bật/tắt) để tự động điều chỉnh dung lượng của các chức năng mạng lõi (Hình 1). Nghĩa là, các nhà khai thác có thể áp dụng các thuật toán chia tỷ lệ để quản lý các phiên bản lát cắt mạng NSI một cách hợp lý [1].



Hình 1. Mô hình bài toán phân chia mạng theo chuẩn 3GPP R15

Như được hiển thị trong Hình 1, một lát cắt mạng bao gồm nhiều NSI, chẳng hạn như NSI_1 và NSI_2 , trong đó một NSI có thể bao gồm nhiều phiên bản con lát cắt mạng (NSSI - *network slice subnet instances*) hoặc một NSI chỉ gồm một NSSI duy nhất. Ví dụ: NSI_1 bao gồm $NSSI_1$ và NSSI chia sẻ chung. Một NSSI bao gồm nhiều phiên bản chức năng mạng (NFI - *network function instances*), về cơ bản là các VNFs. Ví dụ: NEF, PCF, UDM, AUSF, SMF và UPF đều là các VNF bên trong $NSSI_1$. Ngoài ra, các NSI có thể chia sẻ các chức năng của mặt phẳng điều khiển trên một NSSI (ví dụ: NSSI trong Hình 1 được chia sẻ giữa NSI_1 và NSI_2). NSSI chia sẻ chung chủ yếu bao gồm các chức năng mạng mặt phẳng điều khiển. Khi thiết lập một kết nối, đầu tiên người dùng sẽ gửi yêu cầu xác thực ban đầu (initial authentication) đến mặt phẳng điều khiển. Sau khi thiết lập phiên thành công, dữ liệu có thể được truyền qua UPF mà không cần đi qua mặt phẳng điều khiển. Tuy nhiên, khi người dùng đột nhiên gửi một lượng lớn dữ liệu, nhà điều hành có thể tự động thiết lập nhiều UPF hơn nhằm giảm thiểu tắc nghẽn hệ thống.

Trong 5G, UPF là chức năng mặt phẳng người dùng, được sử dụng để truyền dữ liệu. Cụ thể, UPF chịu trách nhiệm duy trì bảng định tuyến và thiết lập các đường hầm GTP để chuyển tiếp các gói tin. Trước khi truyền, UPF giải mã các tiêu đề của gói và tìm kiếm các quy tắc định tuyến (routing rules) trong tập dữ liệu của nó. Trong 3GPP R15, chức năng quản lý phiên (SMF - *session management function*) có thể kiểm soát/điều khiển nhiều UPF. Tuy nhiên, đối với các nhà khai thác (operators), làm thế nào để thiết lập một

số lượng lớn UPF với chi phí thấp nhất đồng thời đáp ứng yêu cầu của người dùng luôn là một thách thức [1].

Tương tự như các chiến lược phân chia tỉ lệ tự động (*autoscaling*) trên đám mây, phân chia tỉ lệ tự động trong *NSSI* có thể mang lại sự linh hoạt, giảm chi phí vận hành và đáp ứng các yêu cầu về hiệu suất của người dùng. Tuy nhiên, phân chia tỉ lệ tự động của *NSSI* trong 3GPP khác với phân chia tỉ lệ tự động trên đám mây truyền thống do những điều sau [1]:

- *Thiết lập khối (Block setup)*.
- *Cơ chế đặt trước tài nguyên (Reservation)*.

Trong bài báo này, chúng tôi sẽ mô hình hóa hệ thống phân chia mạng theo mô hình hàng đợi Markov cho bài toán đặt trước và thiết lập khối động các tài nguyên ứng với các lát cắt mạng trong mạng lõi 5G [1]. Theo đó chúng tôi đề xuất một phương pháp sử dụng ma trận tốc độ chuyển trạng thái để tính toán xác suất trạng thái cân bằng [7]. Nội dung tiếp theo của bài báo bao gồm: phần 2 giới thiệu mô hình phân tích bài toán phân chia mạng trong mạng lõi 5G, bao gồm mô hình hóa hệ thống tổng quát theo mô hình hàng đợi Markov và thuật toán tính xác suất trạng thái cân bằng dựa trên ma trận tốc độ chuyển trạng thái; Kết quả đồ họa về thời gian phản hồi trung bình (W) và tổng chi phí trung bình (C), cùng các phân tích sẽ được trình bày trong phần 3; và cuối cùng là phần kết luận.

2. MÔ HÌNH PHÂN TÍCH BÀI TOÁN PHÂN CHIA MẠNG TRONG MẠNG LỐI 5G

2.1. Mô hình hóa hệ thống tổng quát theo mô hình hàng đợi Markov

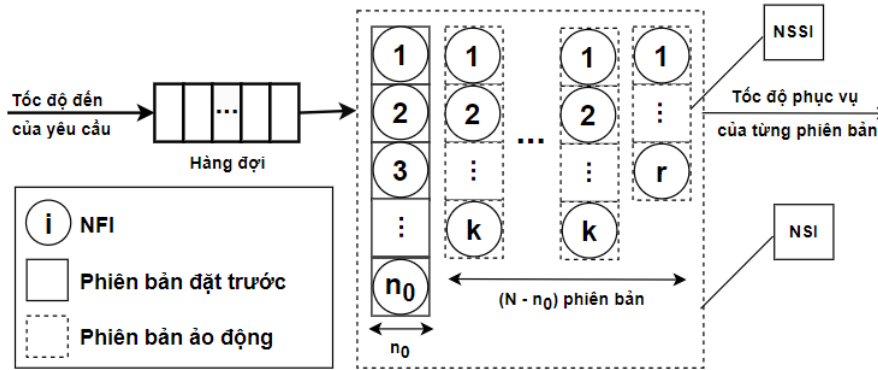
Một số ký hiệu được sử dụng trong mô hình được trình bày trong Bảng 1 [1].

Bảng 1. Các ký hiệu được sử dụng trong mô hình phân tích

Ký hiệu	Ý nghĩa	Ký hiệu	Ý nghĩa
λ	Tốc độ đến trung bình của các yêu cầu dịch vụ	W	Thời gian phản hồi trung bình
K	Dung lượng hệ thống (số lượng yêu cầu tối đa)	C	Tổng chi phí trung bình
N	Tổng số phiên bản mà hệ thống cung cấp	C_b	Chi phí bận trung bình
n_0	Số lượng phiên bản đặt trước	C_i	Chi phí nhàn rỗi trung bình
k	Số lượng phiên bản trong một khối (kích thước khối)	C_s	Chi phí thiết lập trung bình
r	Kích thước của khối bao gồm các phiên bản còn lại	N_b	Số phiên bản bận
α	Tỷ lệ thiết lập	N_i	Số phiên bản nhàn rỗi
		N_s	Số phiên bản thiết lập

μ	Tỷ lệ dịch vụ
m	Ngưỡng trong hàng đợi (để thiết lập khối)
W	Thời gian phản hồi trung bình

Dưới đây là mô hình hóa theo mô hình hàng đợi Markov, như được chỉ ra trong Hình 2 [1, 7].



Hình 2. Mô hình hóa hệ thống đặt trước và thiết lập khối động theo mô hình hàng đợi

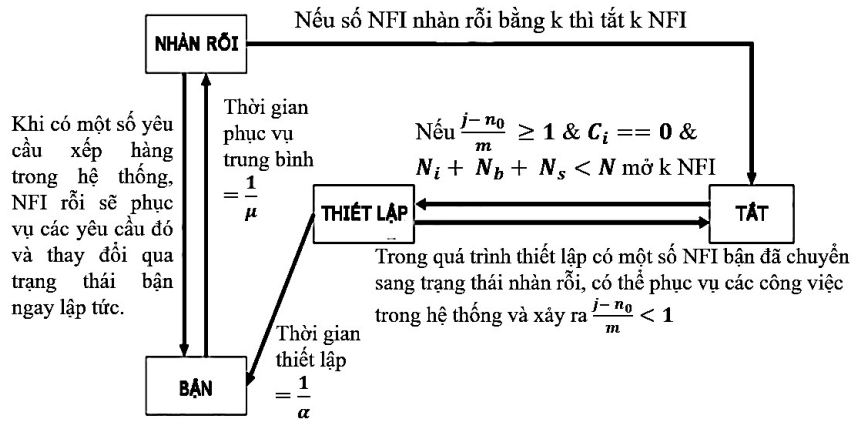
Thông qua mô hình này, có thể tối ưu hóa hiệu quả chi phí của hệ thống bằng cách phân tích *đặt trước* và *giá trị ngưỡng*. Các lát cắt mạng được triển khai với cấu trúc phân lớp trong đó nhiều *NFI* được thiết lập để tạo thành một *NSSI* và nhiều *NSSI* được triển khai để tạo thành một *NSI*. Để đạt được mục tiêu này, chúng ta coi k phiên bản là một *khối* (block) để mô tả mối quan hệ giữa *NFI* và *NSSI* và thiết lập các *khối* riêng biệt để mô tả mối quan hệ giữa *NSSI* và *NSI*. Ngoài ra, vì việc triển khai thường xuyên sẽ làm giảm đáng kể hiệu suất hệ thống, vì vậy cần xem xét cân nhắc giá trị n_0 phiên bản đặt trước (dành riêng) và đặt giá trị ngưỡng (m) phù hợp để kiểm soát thời gian thiết lập. Ở đây, chúng tôi sẽ giới thiệu ba thông số quan trọng là số lượng phiên bản cần đặt trước (n_0), giá trị ngưỡng (m) và kích thước khối (k), như sau:

- *Số lượng phiên bản đặt trước* (n_0): Trong mô hình phân tích, luôn có n_0 phiên bản (*NFI*s) được đặt trước (dự trữ) để có thể luôn sẵn sàng phục vụ người dùng.
- *Kích thước khối* (k): Mỗi khi cần thiết lập các phiên bản, chúng ta sẽ thiết lập một khối gồm k phiên bản. Bằng cách thiết lập nhiều *NSSI*, chúng ta có thể thiết lập *NSI* một cách linh động với số lượng *NSSI* khác nhau.
- *Giá trị ngưỡng* (m): giá trị ngưỡng (m) được sử dụng để xác định khi nào (*thời điểm*) và bao nhiêu (*số lượng*) phiên bản cần được thiết lập theo các chiến lược tiết kiệm chi phí của nhà khai thác. Giá trị của ngưỡng m kiểm soát thời gian thiết lập và số khối.

Trong mô hình phân tích này, việc bật và tắt ($N - n_0$) phiên bản một cách linh động trên khối cơ sở. Cụ thể, mô hình sẽ thiết lập (setup) một khối gồm k phiên bản khi số lượng yêu cầu phục vụ đang chờ trong hàng đợi đạt đến ngưỡng m . Ngoài ra, mô

hình cũng sẽ tắt một khối gồm k phiên bản khi tất cả k phiên bản (trong khối) đều không hoạt động. Trong mỗi khối, có k phiên bản, ngoại trừ khối cuối cùng, có thể chỉ có r phiên bản ($r < k$).

Ngoài ra, mô hình cũng xem xét thời gian thiết lập (setup time), tuân theo phân phối hàm mũ với tốc độ α . Sau khi hoàn tất quá trình thiết lập, các phiên bản có thể ngay lập tức cung cấp dịch vụ khi có yêu cầu đến. Thời gian phục vụ của một yêu cầu tuân theo phân phối hàm mũ với tốc độ μ . Một phiên bản sẽ trở lại trạng thái rỗi ngay sau khi nó hoàn thành phục vụ của mình. Hình 3 biểu thị sơ đồ chuyển đổi trạng thái của một phiên bản. Trạng thái khởi tạo ban đầu là tắt, theo đó một phiên bản không thể cung cấp bất kỳ phục vụ nào.



Hình 3. Quy tắc chuyển đổi trạng thái trong thiết lập khối

Đặt (i, j) biểu thị trạng thái hệ thống, trong đó i biểu thị số phiên bản đã được khởi tạo (nghĩa là phiên bản đang ở trạng thái bận hoặc rỗi), trong khi j là số yêu cầu phục vụ còn lại trong hệ thống. Do đó, $\min(i, j)$ là số phiên bản đang bận, $\max(0, i - j)$ là số phiên bản rỗi và $\max(0, j - i)$ là số các yêu cầu đang chờ phục vụ trong hàng đợi. Đặt $\mu_{i,j}$ biểu thị tốc độ phục vụ ở trạng thái (i, j) và $\alpha_{i,j}$ biểu thị tốc độ thiết lập ở trạng thái (i, j) . Theo đó, ta có [1]:

$$\mu_{i,j} = \min(i\mu, j\mu)$$

$$\alpha_{i,j} = \begin{cases} 0, & i \geq j \\ \min\left(\left\lfloor \frac{j-i}{m} \right\rfloor \alpha, \left\lceil \frac{N-i}{k} \right\rceil \alpha\right) & i < j \end{cases} \quad (1)$$

Ở đây

$$i \in I = \left\{ n_0, n_0 + k, \dots, n_0 + \left\lceil \frac{N-n_0}{k} - 2 \right\rceil k, n_0 + \left\lceil \frac{N-n_0}{k} - 1 \right\rceil k, N \right\} \quad (2)$$

$$j \in J = \{0, 1, \dots, K\}$$

Theo đó, ta có không gian trạng thái S là:

$$S = \{(i, j) | i \in I, j \in J, j \geq f(i)\} \quad (3)$$

Trong đó

$$f(i) = \begin{cases} 0, & i = n_0 \\ i - k + 1, & n_0 < i < N \\ i - r + 1, & i = N \end{cases} \quad (4)$$

Do đó, $((i(t), j(t)); t \geq 0)$ tạo thành chuỗi Markov thời gian liên tục trong không gian trạng thái S , trong đó $|S|$ là số trạng thái trong chuỗi Markov của mô hình.

2.2. Các thông số độ đo hiệu năng

Để đánh giá hiệu suất hệ thống và chi phí, các chỉ số sau đây được xác định như sau [1]:

- Thời gian phản hồi trung bình (W): Thời gian phản hồi được xem xét là thời gian đi tới và trở lại (round-trip) cho một thiết bị người dùng (UE) kết nối với Internet thông qua một trạm cơ sở 5G và mạng lõi 5G.

$$W = \frac{\sum_{(i,j) \in S} j \pi_{i,j}}{\lambda(1 - \sum_{i \in I} \pi_{i,K})} \quad (5)$$

- Tổng chi phí trung bình (C): Chi phí của một phiên ở trạng thái bận rộn, thiết lập và rảnh rỗi được định nghĩa là C , tính được:

$$C = c_1 C_i + (c_2 + c_3 \alpha) C_s + C_b \quad (6)$$

Trong đó,

$$C_i = \sum_{(i,j) \in S} \max(i - j, 0) \pi_{i,j} \quad (7)$$

$$C_s = \sum_{i \in I \setminus \{N\}} \sum_{j=i}^K \min\left(\left\lfloor \frac{j-i}{m} \right\rfloor k, \left(\left\lfloor \frac{N-i}{k} \right\rfloor - 1\right)k + r\right) \pi_{i,j} \quad (8)$$

$$C_b = \sum_{(i,j) \in S} \min(i, j) \pi_{i,j} \quad (9)$$

C_i , C_s và C_b là các chi phí gây ra bởi trạng thái nhàn rỗi, thiết lập và bận rộn của hệ thống, tương ứng. c_1 là hệ số trọng số (weight factor) đối với các phiên bận rỗi. c_2 và c_3 lần lượt là các hệ số trọng số cho chi phí tức thời và chi phí xử lý của toàn bộ quy trình thiết lập [1].

2.3. Phương trình trạng thái cân bằng

Để tính được các thông số độ đo ở trên, ta cần tính các xác suất trạng thái cân bằng của chuỗi Markov. Theo mô hình lược đồ chuyển trạng thái nói chung (minh họa ở Hình 4 trong phần tiếp theo), chúng ta định nghĩa là xác suất trạng thái cân bằng (dừng) của hệ

thống tại trạng thái và được biểu diễn qua công thức (10) – (24) ở dưới đây. Theo đó, ta có các phương trình cân bằng theo ba phần ứng với các khoảng giá trị của chỉ số i , đó là: ($i = n_0$), ($i = n_0 + k, n_0 + 2k, \dots, N - r$) và ($i = N$) [1].

(I). $\pi_{i,j}$ với $i = n_0$:

$$\lambda\pi_{n_0,j} = \mu_{n_0,j+1}\pi_{n_0,j+1}, \quad j = 0 \quad (10)$$

$$(\lambda + \mu_{n_0,j})\pi_{n_0,j} = \lambda\pi_{n_0,j-1} + \mu_{n_0,j+1}\pi_{n_0,j+1}, \quad 0 < j < n_0 \quad (11)$$

$$(\lambda + \mu_{n_0,j})\pi_{n_0,j} = \lambda\pi_{n_0,j-1} + \mu_{n_0,j+1}\pi_{n_0,j+1} + \mu_{n_0,j+1}\pi_{n_0+k,j+1}, \quad j = n_0 \quad (12)$$

$$(\lambda + \mu_{n_0,j} + \alpha_{n_0,j})\pi_{n_0,j} = \lambda\pi_{n_0,j-1} + \mu_{n_0,j+1}\pi_{n_0,j+1}, \quad n_0 < j < K \quad (13)$$

$$(\mu_{n_0,j} + \alpha_{n_0,j})\pi_{n_0,j} = \lambda\pi_{n_0,j-1}, \quad j = K \quad (14)$$

Khi $n_0 = 0$, chúng ta thu được (15) thay vì (10), (11) và (12):

$$\lambda\pi_{0,j} = \mu_{0+k,j+1}\pi_{0+k,j+1}, \quad j = 0 \quad (15)$$

(II). $\pi_{i,j}$ với $i = n_0 + k, n_0 + 2k, \dots, N - r$:

$$(\lambda + \mu_{i,j})\pi_{i,j} = \mu_{i,j+1}\pi_{i,j+1} + \alpha_{i-k,j}\pi_{i-k,j}, \quad j = i - k + 1 \quad (16)$$

$$(\lambda + \mu_{i,j})\pi_{i,j} = \lambda\pi_{i,j-1} + \mu_{i,j+1}\pi_{i,j+1} + \alpha_{i-k,j}\pi_{i-k,j}, \quad i - k + 1 < j < i \quad (17)$$

$$(\lambda + \mu_{i,j})\pi_{i,j} = \lambda\pi_{i,j-1} + \mu_{i,j+1}\pi_{i,j+1} + \alpha_{i-k,j}\pi_{i-k,j} + \mu_{\min(i+k,N),j+1}\pi_{\min(i+k,N),j+1}, \quad j = i \quad (18)$$

$$(\lambda + \mu_{i,j} + \alpha_{i,j})\pi_{i,j} = \lambda\pi_{i,j-1} + \mu_{i,j+1}\pi_{i,j+1} + \alpha_{i-k,j}\pi_{i-k,j}, \quad i < j < K \quad (19)$$

$$(\mu_{i,j} + \alpha_{i,j})\pi_{i,j} = \lambda\pi_{i,j-1} + \alpha_{i-k,j}\pi_{i-k,j}, \quad j = K \quad (20)$$

Khi $k = 1$, chúng ta thu được (21) thay vì (16), (17) và (18):

$$(\lambda + i\mu)\pi_{i,i} = (i + 1)\mu\pi_{\min(i+k,N),i+1} + i\mu\pi_{i,i+1} + \alpha_{i-k,i}\pi_{i-k,i}, \quad j = i \quad (21)$$

(III). $\pi_{i,j}$ với $i = N$:

$$(\lambda + \mu_{N,j})\pi_{N,j} = \mu_{N,j+1}\pi_{N,j+1} + \alpha_{N-r,j}\pi_{N-r,j}, \quad j = N - r + 1 \quad (22)$$

$$(\lambda + \mu_{N,j})\pi_{N,j} = \lambda\mu_{N,j-1} + \mu_{N,j+1}\pi_{N,j+1} + \alpha_{N-r,j}\pi_{N-r,j}, \quad N - r + 1 < j < K \quad (23)$$

$$\mu_{N,j}\pi_{N,j} = \lambda\pi_{N,j-1} + \alpha_{N-r,j}\pi_{N-r,j}, \quad j = K \quad (24)$$

Tổng của các giá trị xác suất cần bằng này phải bằng 1 do tính chất sau đây:

$$\sum_{(i,j) \in S} \pi_{i,j} = 1 \quad (25)$$

2.4. Xây dựng thuật toán tính xác suất trạng thái cân bằng

Trong bài báo này, chúng tôi xây dựng một thuật toán tính các xác suất trạng thái cân bằng $\pi_{i,j}$ bằng cách chuyển các công thức (10) – (24) sang dạng ma trận (ma trận tốc độ chuyển trạng thái) [7], từ đó có thể tính được các thông số độ đo theo các

công thức (5) – (9). Theo đó, chúng tôi đưa ra thuật toán như sau:

Thuật toán. Xây dựng ma trận tốc độ chuyển trạng thái (ký hiệu là Q) như sau:

Bảng 2. Ma trận tổng quát Q ($n_s \times n_s$)

$$Q = \begin{array}{c} \begin{array}{ccccc} & A_0 & B_0 & & \\ & C_1 & A_1 & B_1 & \\ & & C_2 & A_2 & \dots \\ & & & \dots & \dots & B_i \\ & & & & C_i & A_i \end{array} \end{array}$$

Input: Không gian trạng thái S .

Bước 1: Tạo ra các ma trận chuyển trạng thái A_i, B_i, C_i như sau:

$A_i(l, j)$ xác định tốc độ chuyển trạng thái (i, j) sang trạng thái (l, j) với:

$$n_0 \leq i, l \leq N, \text{ với } i, l \in I = \left\{ n_0, n_0 + k, \dots, n_0 + \left\lceil \frac{N - n_0}{k} - 2 \right\rceil k, n_0 + \left\lceil \frac{N - n_0}{k} - 1 \right\rceil k, N \right\};$$

$$j \in J = \{0, 1, 2, 3, 4, \dots, K\}; j \geq f(i) = \begin{cases} 0, & i = n_0 \\ i - k + 1, & n_0 < i < N \\ i - r + 1, & i = N \end{cases}$$

Ma trận $A_i(l, j)$ có kích thước là $(K + 1 - f(i)) \times (K + 1 - f(i))$

Các phần tử khác 0 của ma trận A_i được tính như sau:

$$A_i(z, z - 1) = \min(i\mu, (z + f(i))\mu), \text{ với } 1 \leq z \leq K - f(i) \quad (26)$$

$$\text{và} \quad A_i(z, z + 1) = \lambda, \text{ với } 0 \leq z \leq K - 1 - f(i) \quad (27)$$

$B_i(l, j)$ xác định tốc độ chuyển trạng thái (i, j) sang trạng thái (l, s) với:

$$n_0 \leq i < N, \text{ với } i \in I = \left\{ n_0, n_0 + k, \dots, n_0 + \left\lceil \frac{N - n_0}{k} - 2 \right\rceil k, n_0 + \left\lceil \frac{N - n_0}{k} - 1 \right\rceil k \right\}$$

$$j \in J = \{0, 1, 2, 3, 4, \dots, K\}; j \geq f(i) = \begin{cases} 0, & i = n_0 \\ i - k + 1, & n_0 < i < N \end{cases}$$

$$n_0 < l \leq N, \text{ với } l \in I = \left\{ n_0 + k, \dots, n_0 + \left\lceil \frac{N - n_0}{k} - 2 \right\rceil k, n_0 + \left\lceil \frac{N - n_0}{k} - 1 \right\rceil k, N \right\}$$

$$s \in J = \{0, 1, 2, 3, 4, \dots, K\}; s \geq f(l) = \begin{cases} l - k + 1, & n_0 < l < N \\ l - r + 1, & l = N \end{cases}$$

Ma trận $B_i(l, j)$ có kích thước là $(K + 1 - f(i)) \times (K + 1 - f(l))$

Các phần tử khác 0 của ma trận B_i được tính như sau:

$$B_i(z + n_0 + m, z + 1) = \min\left(\left\lfloor \frac{z + f(l) - f(i) - 1}{m} \right\rfloor \alpha, \left\lfloor \frac{N - i}{k} \right\rfloor \alpha\right) \quad (28)$$

với $0 \leq z \leq K - f(l) - 1$, (α tốc độ thiết lập)

$C_i(l, j)$ xác định tốc độ chuyển trạng thái (i, j) sang trạng thái (l, m) với:

$$\begin{aligned} n_0 < i \leq N, \text{ với } i \in I &= \left\{ n_0 + k, \dots, n_0 + \left\lfloor \frac{N - n_0}{k} - 2 \right\rfloor k, n_0 + \left\lfloor \frac{N - n_0}{k} - 1 \right\rfloor k, N \right\} \\ j \in J &= \{0, 1, 2, 3, 4, \dots, K\}; j \geq f(i) = \begin{cases} i - k + 1, & n_0 < i < N \\ i - r + 1, & i = N \end{cases} \\ n_0 \leq l < N, \text{ với } l \in I &= \left\{ n_0, n_0 + k, \dots, n_0 + \left\lfloor \frac{N - n_0}{k} - 2 \right\rfloor k, n_0 + \left\lfloor \frac{N - n_0}{k} - 1 \right\rfloor k \right\} \\ m \in J &= \{0, 1, 2, 3, 4, \dots, K\}; m \geq f(l) = \begin{cases} 0, & l = n_0 \\ l - k + 1, & n_0 < l < N \end{cases} \end{aligned}$$

Ma trận $C_i(l, j)$ có kích thước là $(K + 1 - f(i)) \times (K + 1 - f(l))$

Các phần tử khác 0 của ma trận C_i được tính như sau:

$$C_i(z, z + n_0 + m - 2) = \min(i\mu, f(i)\mu) \quad (\mu \text{ tốc độ phục vụ}) \quad \text{với } z = 0 \quad (29)$$

Bước 2: Tính các giá trị trên đường chéo ma trận Q : $Q(l, l) = -(\text{tổng các phần tử trên dòng } l)$, như sau:

$$Q(l, l) = - \sum_{s=0, s \neq l}^{n_S-1} Q_{l,s}, \text{ với } n_S = \left(\left\lfloor \frac{N - n_0}{k} \right\rfloor + 1\right) \times (K + 1) - \sum_{i \in I} f(i), i \in I.$$

Tương ứng với

$$\sum_{l,s=0, s \neq l}^{n_S-1} Q_{l,s} = 0$$

Output: Ta được ma trận Q có kích thước $(n_S \times n_S)$.

Để tính vector p (có kích thước n_S) chứa các xác suất trạng thái cân bằng, ta cần giải phương trình đại số tuyến tính $p \cdot Q = 0$.

Input: Ma trận Q

Với điều kiện $\sum_{i=0}^{n_S-1} p_i = 1$, ta có $p \cdot E = e$, ở đây E là ma trận có kích thước $(n_S \times n_S)$ với tất cả các phần tử của E đều bằng 1, và e là vector có kích thước n_S với tất cả các phần tử của e đều bằng 1.

Do đó, ta có $p \cdot Q + p \cdot E = 0 + e$ hay $p \cdot (Q + E) = e$.

Cuối cùng, $p = e \cdot (Q + E)^{-1}$. Ma trận $(Q + E)^{-1}$ là nghịch đảo của ma trận $(Q + E)$.

Output: Vector p .

3. KẾT QUẢ PHÂN TÍCH

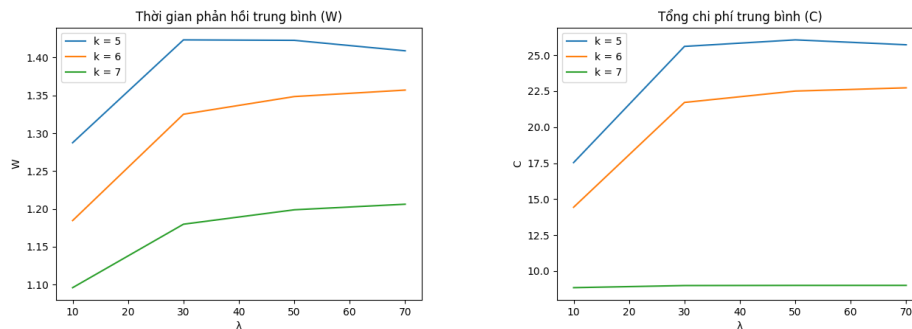
Như được hiển thị trong Hình 4, chúng tôi hiển thị các tính năng của cơ chế thiết lập khối với sự thay đổi về lưu lượng đến λ trên các chỉ số hiệu suất W , C theo các cài đặt tham số sau: $\mu = 1$; $\alpha = 0,5$; $n_0 = 2$; $m = 2$; $N = 10$; $K = 11$; $c_1 = 0,6$; $c_2 = 1,0$; $c_3 = 3,0$. Để biết rõ về sự thay đổi của hiệu suất và chi phí chúng ta tiến hành thay đổi kích thước khối k và lưu lượng đến.

Thay đổi kích thước khối $k = 5, 6, 7$:

- Thời gian phản hồi trung bình của hệ thống W dần giảm xuống, điều này là hiển nhiên vì khi k tăng sẽ có nhiều phiên bản rồi có thể giải quyết được ngay các yêu cầu đến từ người dùng, giảm đi thời gian thiết lập khối, từ đó làm giảm W .
- Tổng chi phí của hệ thống C dần giảm xuống vì khi k tăng giúp tránh việc thiết lập (bật) khối k nhiều lần, từ đó giảm đi chi phí thiết lập C_s , do đó C giảm.

Thay đổi lưu lượng đến $\lambda = 10, 30, 50, 70$:

- Thời gian phản hồi trung bình W tăng lên vì khi lưu lượng đến λ tăng lên mà các phiên bản nhàn rỗi không có sẵn để thực hiện yêu cầu thì phải thiết lập khối k , dẫn đến thời gian phản hồi trung bình W tăng lên.
- Tổng chi phí trung bình C tăng lên, do khi lưu lượng đến λ tăng cao thì cần phải thiết lập khối k để có thể giải quyết các yêu cầu, dẫn đến chi phí thiết lập C_s tăng lên khiến C tăng.



(a) Thời gian phản hồi trung bình (W) (b) Tổng chi phí trung bình (C)

Hình 4. Tác động của kích thước khối ($k = 5, 6, 7$) đối với (W) và (C)

4. KẾT LUẬN

Bài báo đã đề xuất một thuật toán để xây dựng ma trận chuyển trạng thái Q dựa trên mô hình Markov 2 chiều phân tích bài toán phân chia mạng trong mạng lõi 5G sử dụng mô hình hàng đợi, từ đó tính các xác suất trạng thái cân bằng. Kết quả phân tích cũng như vấn đề xây dựng cài đặt thuật toán cho thấy tính hiệu quả và dễ sử dụng của

thuật toán đề xuất. Hướng phát triển của bài báo sẽ là áp dụng cơ chế kiểm soát theo ngưỡng để tối ưu việc thiết lập và gỡ bỏ khối trong mô hình.

TÀI LIỆU THAM KHẢO

- [1]. Cheng-Ying Hsieh, Tuan Phung-Duc, Yi Ren, Jyh-Cheng Chen (2022), "Design and Analysis of Dynamic Block-setup Reservation Algorithm for 5G Network Slicing", IEEE Transactions on Mobile Computing, pp. 1536-1233. DOI 10.1109/TMC.2022.3169034
- [2]. 5GPP, "View on 5G Architecture," White paper 22.891, 5GPP Architecture Working Group, 07 2018. Version 14.2.0.
- [3]. 3GPP, "5G; Study on scenarios and requirements for next generation access technologies," Technical Specification (TS) 38.913, 3rd Generation Partnership Project (3GPP), 07 2020. Version 16.0.0 Release 16.
- [4]. 3GPP, "Technical Specification Group Services and System Aspects; System architecture for the 5G System (5GS); Stage 2," Technical Specification (TS) 23.501, 3rd Generation Partnership Project (3GPP), 12 2020. Version 16.7.0.
- [5]. V. Ziegler, H. Viswanathan, H. Flinck, M. Hoffmann, V. Räsänen, and K. Hätönen, "6G architecture to connect the worlds," IEEE Access, vol. 8, pp. 173508–173520, 2020.
- [6]. C. Rotter and T. V. Do, "A queueing model for threshold-based scaling of UPF instances in 5G core," IEEE Access, vol. 9, pp. 81443–81453, 2021.
- [7]. Đặng Thanh Chương, "Một phương pháp tính ma trận tốc độ chuyển trạng thái với mô hình chuyển đổi bước sóng từng phần trong mạng chuyển mạch chùm quang OBS", Nghiên cứu cơ bản và ứng dụng Công nghệ thông tin, Đồng Nai, pp. 121 – 131, 2011.

A METHOD FOR CALCULATING STATE TRANSITION RATE MATRIX IN NETWORK SLICING PROBLEM WITHIN 5G CORE NETWORK USING QUEUEING MODEL

Dang Thanh Chuong, Hoang Cong Hoang, Hoa Ly Cuong

University of Sciences, Hue University

Email: dtchuong@hueuni.edu.vn, hoangconghoang511@gmail.com

ABSTRACT

Network functions in 5G can be dynamically scaled out/in to adjust the capacity of network slices. The scale-out/-in procedure, namely autoscaling, enhances performance by scaling out instances and reduces operational costs by scaling in

instances. The Markov queueing model is initially used to simulate the system in this research. Then, the queueing model is applied to the reservation problem, and resources corresponding to network slices in the 5G core network are dynamically allocated [1]. Furthermore, an algorithm is proposed to calculate the equilibrium state probability based on transforming the balance state equations into the form of a general Q matrix, which is also a key result of this paper.

Keywords: Network Slicing, Reservation, Resource Allocation, 5G, Performance Analysis, Transition Rate Matrix.



Đặng Thanh Chương sinh ngày 23/3/1975 tại TP Vinh. Ông tốt nghiệp cử nhân ngành Vật lý năm 1997 và thạc sĩ chuyên ngành Tin học tại Trường ĐHKH Huế vào năm 2004. Ông nhận học vị tiến sĩ chuyên ngành Toán học năm 2014 tại Viện CNTT - Viện hàn lâm khoa học và công nghệ Việt Nam. Từ năm 1997 đến nay, ông công tác tại khoa CNTT, Trường Đại học Khoa học, Đại học Huế.

Lĩnh vực nghiên cứu: Lý thuyết hàng đợi, Mạng 5G, Mạng truyền dẫn quang.



Hoàng Công Hoàng sinh ngày 05/11/1994 tại TP Huế. Ông tốt nghiệp cử nhân ngành Toán ứng dụng năm 2016 và thạc sĩ chuyên ngành Khoa học máy tính tại Trường ĐHKH Huế vào năm 2023.

Lĩnh vực nghiên cứu: Lý thuyết hàng đợi, Mạng 5G.



Hoa Lý Cương sinh năm 1990 tại An Giang. Ông tốt nghiệp cử nhân ngành Công nghệ Thông tin năm 2012 tại Trường Đại học Khoa học Tự nhiên, Đại học Quốc gia thành phố Hồ Chí Minh và thạc sĩ chuyên ngành Khoa học Máy tính tại Trường Đại học Khoa học, Đại học Huế vào năm 2017. Hiện tại đang là nghiên cứu sinh ngành Khoa học Máy tính tại Trường Đại học Khoa học, Đại học Huế.

Lĩnh vực nghiên cứu: Lý thuyết hàng đợi, Đánh giá hiệu năng mạng.