



(12) BẢN MÔ TẢ SÁNG CHẾ THUỘC BẰNG ĐỘC QUYỀN SÁNG CHẾ

(19) Cộng hòa xã hội chủ nghĩa Việt Nam (VN)
CỤC SỞ HỮU TRÍ TUỆ

(11)



1-0052394

(51)^{2013.01} H04L 12/801; H04W 28/02; H04L
29/08; H04L 1/22

(13) B

(21) 1-2022-06546

(22) 12/03/2021

(86) PCT/EP2021/056306 12/03/2021

(87) WO2021/180910 16/09/2021

(30) 62/989,340 13/03/2020 US

(45) 27/10/2025 451

(43) 26/12/2022 417A

(73) TELEFONAKTIEBOLAGET LM ERICSSON (PUBL) (SE)
SE-164 83 Stockholm, Sweden

(72) VARGA, Balázs (HU); FARKAS, János (HU).

(74) Công ty Luật TNHH T&G (TGVN)

(54) PHƯƠNG PHÁP VÀ NÚT TRUYỀN ĐỂ TÁI TẠO GÓI TIN HOẶC KHUNG

(21) 1-2022-06546

(57) Sáng chế đề cập tới phương pháp được thực hiện bởi nút truyền và nút truyền để tái tạo gói tin hoặc khung, phương pháp được thực hiện bởi nút thu nhận và nút thu nhận để loại bỏ gói tin hoặc khung. Theo một phương án thực hiện, phương pháp được thực hiện bởi nút truyền bao gồm bước xác định rằng chức năng sinh ra trình tự ở nút truyền đã được cài đặt lại và, đáp ứng với điều đó, truyền các gói tin thứ nhất trong dòng qua ít nhất hai đường dẫn tách rời, trong đó: (a) mỗi một gói tin trong số các gói tin thứ nhất bao gồm số trình tự tương ứng từ không gian số trình tự tuyến tính và (b) ít nhất một gói tin thứ nhất từ các gói tin thứ nhất bao gồm bộ chỉ báo rõ ràng của sự cài đặt lại. Phương pháp nêu trên còn bao gồm bước xác định rằng đầu cuối của không gian số trình tự tuyến tính đã được đạt tới hoặc rằng việc sử dụng không gian số trình tự tuyến tính đã bị vô hiệu hóa và, đáp ứng với điều đó, truyền các gói tin thứ hai trong dòng qua ít nhất hai đường dẫn tách rời, trong đó (a) mỗi một gói tin trong số các gói tin thứ hai bao gồm số trình tự tương ứng của không gian số trình tự tuần hoàn.

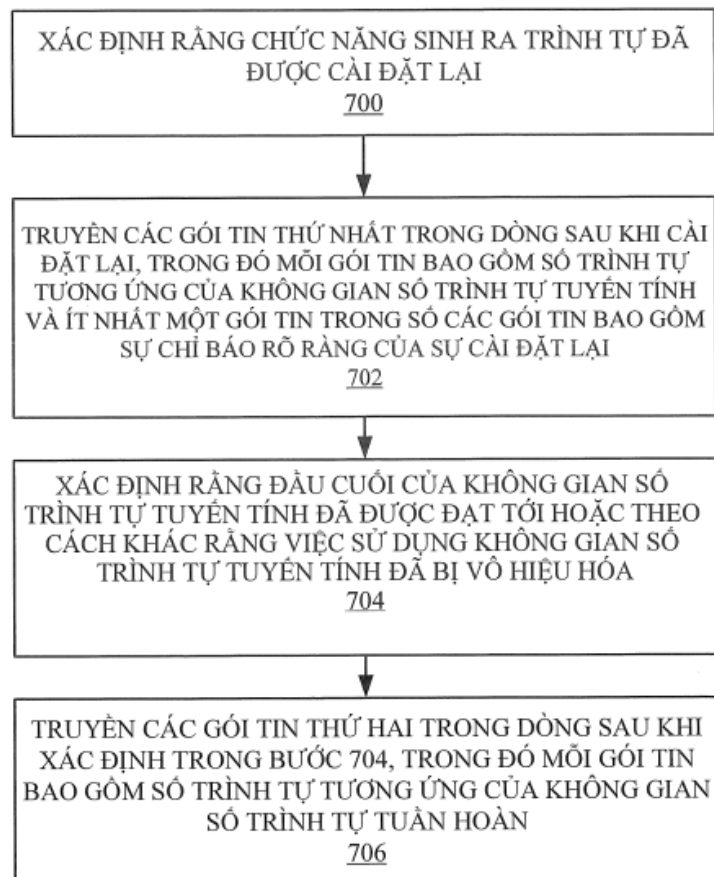


FIG.7

Lĩnh vực kỹ thuật được đề cập

Sáng chế đề cập tới sự nổi mạng nhạy với thời gian (TSN - Time Sensitive Networking) và sự nổi mạng tất định (DetNet - Deterministic Networking) và, cụ thể hơn, tới sự tái tạo và sự loại bỏ khung hoặc gói tin trong mạng TSN hoặc DetNet.

Tình trạng kỹ thuật của sáng chế

Sự nổi mạng nhạy với thời gian (TSN) hiện đang được phát triển ở viện kỹ sư điện và điện tử (IEEE - Institute for Electronics and Electrical Engineering) như một kỹ thuật mới nhằm nâng cao các tiêu chuẩn Ethernet IEEE 802.1 và IEEE 802.3 tới mức hoàn toàn mới của thuyết quyết định. Nó có thể được xem như sự tiến hóa của Ethernet để đảm bảo thời gian chờ từ đầu cuối tới đầu cuối thấp, sự biến động thấp, và sự tổn thất gói tin thấp.

Nhóm nhiệm vụ (TG - Task Group) TSN trong nhóm hoạt động (WG - Working Group) IEEE 802.1 giải quyết các dịch vụ tất định qua các mạng IEEE 802. TSN TG chỉ rõ các công cụ của hộp công cụ TSN, cũng như việc sử dụng các công cụ cho mục đích cụ thể. TSN TG được điều hành để cung cấp các dịch vụ tất định qua các mạng IEEE 802 với:

- việc vận chuyển gói tin được đảm bảo,
- sự tổn thất gói tin thấp,
- thời gian chờ được chặn dưới, và
- sự thay đổi độ trễ gói tin thấp.

Để đạt được sự tổn thất gói tin cực kỳ thấp, TSN TG đã chỉ rõ sự tái tạo và sự loại bỏ khung cho độ tin cậy (FRER - Frame Replication and Elimination for Reliability) (802.1CB), vốn được nhằm để tránh sự tổn thất khung do sự cố thiết bị. Trên thực tế nó là hàm dư 1+1 (hoặc 1+n) theo mỗi khung. Không có việc dò sự cố/sự chuyển đổi được hợp nhất. FRER gửi các khung trên hai (hoặc nhiều hơn) đường dẫn tách rời lớn nhất, sau đó kết hợp các dòng và xóa các khung bổ sung.

Chú ý rằng các chức năng tương tự được xác định cho các mạng nổi mạng tất định (DetNet) như các chức năng tái tạo và loại bỏ gói tin (PREF - Packet Replication and

Elimination Function) để đơn giản hóa việc thực hiện và cho phép sử dụng khái niệm tương tự trong cả mạng lớp 2 (TSN) và lớp 3 (DetNet). Trong phần mô tả được đưa ra ở đây, sự tập trung nằm trên FRER.

Lưu ý theo IEEE 802.1CB: "... tiêu chuẩn này xác định sự tái tạo và sự loại bỏ khung cho độ tin cậy (FRER), vốn chia một dòng thành một hoặc nhiều dòng thành phần được liên kết, nhờ đó làm cho dòng ban đầu trở thành dòng hỗn hợp. Nó tái tạo các gói tin của dòng, tách các bản sao thành nhiều dòng thành phần, và sau đó ghép các dòng thành phần này lại ở một hoặc nhiều điểm khác, loại bỏ các phần tái tạo, và phân phối dòng đã cấu tạo lại từ các điểm này."

Chức năng loại bỏ đánh giá tham số phụ "sequence_number" của gói tin của một hoặc nhiều dòng thành phần được chuyển lên từ các lớp dưới, để loại bỏ các gói tin bị lặp. Biến số "SequenceHistory" duy trì lịch sử của các tham số phụ "sequence_number" của các gói tin đã thu nhận gần đây. Trong quá trình loại bỏ sự lặp, "sequence_number" được kiểm tra dựa vào cửa sổ lịch sử (được xác định bởi "frerSeqRcvyHistoryLength"). Các gói tin nằm bên ngoài cửa sổ lịch sử được loại bỏ dưới dạng không hợp lệ. Ở điều kiện vận hành bình thường, các gói tin đã thu nhận nằm trong cửa sổ lịch sử và chỉ các phần lặp bị rơi.

IEEE 802.1CB xác định cơ chế quá thời gian cho chức năng loại bỏ để đối phó một vài kịch bản nối mạng mà dẫn tới các khung bị rơi không cần thiết (chẳng hạn, nếu chức năng loại bỏ bằng cách nào đó không hoạt động với chức năng sinh ra trình tự tương ứng của nó; nếu chức năng sinh ra trình tự được cài đặt lại; v.v.). Nếu sự quá thời gian xuất hiện, lịch sử được cài đặt lại, và nó được cho phép chấp nhận gói tin tiếp theo bởi thuật toán phục hồi, bất kể giá trị của tham số phụ "sequence_number" của nó (xem "TakeAny" trong 7.4.3.2.6 trong IEEE 802.1CB).

Bản chất kỹ thuật của sáng chế

Các hệ thống và các phương pháp được bộc lộ ở đây để cài đặt lại liên mạch đối với sự dư thừa liên mạch, chẳng hạn, của mạng nối mạng nhảy với thời gian (TSN) hoặc mạng nối mạng tắt định (DetNet). Theo một phương án thực hiện, phương pháp được thực hiện bởi nút truyền để tái tạo gói tin hoặc khung bao gồm bước xác định rằng chức năng sinh ra trình tự ở nút truyền đã được cài đặt lại. Phương pháp nêu trên còn bao

gồm, đáp ứng với bước xác định rằng chức năng sinh ra trình tự ở nút truyền đã được cài đặt lại, bước truyền các gói tin thứ nhất trong dòng các gói tin qua ít nhất hai đường dẫn tách rời qua mạng, trong đó: (a) mỗi một gói tin trong số các gói tin thứ nhất bao gồm số trình tự tương ứng từ không gian số trình tự tuyến tính và (b) ít nhất một gói tin thứ nhất từ trong số các gói tin thứ nhất mà được gửi sau khi cài đặt lại còn bao gồm bộ chỉ báo rõ ràng của sự cài đặt lại. Phương pháp nêu trên còn bao gồm bước xác định rằng đầu cuối của không gian số trình tự tuyến tính đã được đạt tới hoặc theo cách khác rằng việc sử dụng của không gian số trình tự tuyến tính đã bị vô hiệu hóa. Phương pháp nêu trên còn bao gồm, đáp ứng với bước xác định rằng đầu cuối của không gian số trình tự tuyến tính đã được đạt tới hoặc theo cách khác rằng việc sử dụng của không gian số trình tự tuyến tính đã bị vô hiệu hóa, bước truyền các gói tin thứ hai trong dòng các gói tin qua ít nhất hai đường dẫn tách rời qua mạng, trong đó (a) mỗi một gói tin trong số các gói tin thứ hai bao gồm số trình tự tương ứng của không gian số trình tự tuần hoàn. Theo cách này, việc cài đặt lại liên mạch đối với việc sinh ra số trình tự được đề xuất.

Theo một phương án thực hiện, mỗi một gói tin trong số các gói tin thứ nhất còn bao gồm sự chỉ báo rõ ràng rằng không gian số trình tự tuyến tính đang được sử dụng.

Theo một phương án thực hiện, mạng nêu trên là mạng TSN.

Theo một phương án thực hiện, phương pháp nêu trên còn bao gồm bước cài đặt lại chức năng sinh ra trình tự, trong đó việc cài đặt lại chức năng sinh ra trình tự bao gồm bước cài đặt lại lịch sử số trình tự, cửa sổ lịch sử, hoặc cả lịch sử số trình tự và cửa sổ lịch sử.

Theo một phương án thực hiện, các bước gồm xác định rằng chức năng sinh ra trình tự ở nút truyền đã được cài đặt lại, truyền các gói tin thứ nhất trong dòng các gói tin, xác định rằng đầu cuối của không gian số trình tự tuyến tính đã được đạt tới hoặc theo cách khác rằng việc sử dụng không gian số trình tự tuyến tính đã bị vô hiệu hóa, và truyền các gói tin thứ hai trong dòng các gói tin được thực hiện bởi chức năng tái tạo và loại bỏ khung cho độ tin cậy (FRER) của nút truyền.

Theo một phương án thực hiện, mạng nêu trên là mạng DetNet. Theo một phương án thực hiện, các bước gồm xác định rằng chức năng sinh ra trình tự ở nút truyền đã được cài đặt lại, truyền các gói tin thứ nhất trong dòng các gói tin, xác định rằng đầu

cuối của không gian số trình tự tuyến tính đã được đạt tới hoặc theo cách khác rằng việc sử dụng không gian số trình tự tuyến tính đã bị vô hiệu hóa, và truyền các gói tin thứ hai trong dòng các gói tin được thực hiện bởi chức năng tái tạo và loại bỏ gói tin (PRER - Packet Replication and Elimination Function) của nút truyền.

Các phương án thực hiện tương ứng của nút truyền cũng được bộc lộ. Theo một phương án thực hiện, nút truyền để tái tạo gói tin hoặc khung được làm thích ứng để xác định rằng chức năng sinh ra trình tự ở nút truyền đã được cài đặt lại. Nút truyền còn được làm thích ứng để, đáp ứng với bước xác định rằng chức năng sinh ra trình tự ở nút truyền đã được cài đặt lại, truyền các gói tin thứ nhất trong dòng các gói tin qua ít nhất hai đường dẫn tách rời qua mạng, trong đó: (a) mỗi một gói tin trong số các gói tin thứ nhất bao gồm số trình tự tương ứng từ không gian số trình tự tuyến tính và (b) ít nhất một gói tin thứ nhất từ trong số các gói tin thứ nhất mà được gửi sau khi cài đặt lại còn bao gồm bộ chỉ báo rõ ràng của sự cài đặt lại. Nút truyền còn được làm thích ứng để xác định rằng đầu cuối của không gian số trình tự tuyến tính đã được đạt tới hoặc theo cách khác rằng việc sử dụng không gian số trình tự tuyến tính đã bị vô hiệu hóa và, đáp ứng với bước xác định rằng đầu cuối của không gian số trình tự tuyến tính đã được đạt tới hoặc theo cách khác rằng việc sử dụng không gian số trình tự tuyến tính đã bị vô hiệu hóa, truyền các gói tin thứ hai trong dòng các gói tin qua ít nhất hai đường dẫn tách rời qua mạng, trong đó (a) mỗi một gói tin trong số các gói tin thứ hai bao gồm số trình tự tương ứng của không gian số trình tự tuần hoàn.

Theo một phương án thực hiện, nút truyền để tái tạo gói tin hoặc khung bao gồm giao diện mạng và hệ mạch xử lý được kết hợp với giao diện mạng này. Hệ mạch xử lý được tạo cấu hình để làm cho nút truyền xác định rằng chức năng sinh ra trình tự ở nút truyền đã được cài đặt lại. Hệ mạch xử lý còn được tạo cấu hình để làm cho nút truyền, đáp ứng với bước xác định rằng chức năng sinh ra trình tự ở nút truyền đã được cài đặt lại, truyền các gói tin thứ nhất trong dòng các gói tin qua ít nhất hai đường dẫn tách rời qua mạng, trong đó: (a) mỗi một gói tin trong số các gói tin thứ nhất bao gồm số trình tự tương ứng từ không gian số trình tự tuyến tính và (b) ít nhất một gói tin thứ nhất từ trong số các gói tin thứ nhất mà được gửi sau khi cài đặt lại còn bao gồm bộ chỉ báo rõ ràng của sự cài đặt lại. Hệ mạch xử lý còn được tạo cấu hình để làm cho nút truyền xác

định rằng đầu cuối của không gian số trình tự tuyến tính đã được đạt tới hoặc theo cách khác rằng việc sử dụng không gian số trình tự tuyến tính đã bị vô hiệu hóa và, đáp ứng với bước xác định rằng đầu cuối của không gian số trình tự tuyến tính đã được đạt tới hoặc theo cách khác rằng việc sử dụng không gian số trình tự tuyến tính đã bị vô hiệu hóa, truyền các gói tin thứ hai trong dòng các gói tin qua ít nhất hai đường dẫn tách rời qua mạng, trong đó (a) mỗi một gói tin trong số các gói tin thứ hai bao gồm số trình tự tương ứng của không gian số trình tự tuần hoàn.

Theo một phương án thực hiện, phương pháp được thực hiện bởi nút truyền để tái tạo gói tin hoặc khung bao gồm bước thu được gói tin cần được gửi và bước xác định rằng gói tin nêu trên không bao gồm thẻ mà chỉ báo rằng gói tin đã được tái tạo. Phương pháp nêu trên còn bao gồm, đáp ứng với bước xác định rằng gói tin không bao gồm thẻ mà chỉ báo rằng gói tin đã được tái tạo, bước xác định rằng không gian số trình tự tuyến tính sẽ được sử dụng và, đáp ứng theo đó, cho phép sự chỉ báo rõ ràng trong gói tin mà chỉ báo rằng không gian số trình tự tuyến tính đang được sử dụng, bổ sung số trình tự từ không gian số trình tự tuyến tính vào gói tin, và bổ sung thẻ vào gói tin mà chỉ báo rằng gói tin đã được tái tạo. Phương pháp nêu trên còn bao gồm bước cung cấp gói tin để tái tạo và truyền.

Theo một phương án thực hiện, phương pháp nêu trên còn bao gồm, đáp ứng với bước xác định rằng gói tin nêu trên không bao gồm thẻ mà chỉ báo rằng gói tin đã được tái tạo, bước xác định rằng chức năng sinh ra trình tự ở nút truyền đã được cài đặt lại, cho phép sự chỉ báo rõ ràng trong gói tin mà chỉ báo rằng chức năng sinh ra trình tự đã được cài đặt lại đáp ứng với bước xác định rằng chức năng sinh ra trình tự ở nút truyền đã được cài đặt lại, và theo cách khác vô hiệu hóa sự chỉ báo rõ ràng trong gói tin mà chỉ báo rằng chức năng sinh ra trình tự đã được cài đặt lại.

Theo một phương án thực hiện, phương pháp nêu trên còn bao gồm, đáp ứng với bước xác định rằng không gian số trình tự tuyến tính sẽ được sử dụng, bước tăng số trình tự từ không gian số trình tự tuyến tính.

Theo một phương án thực hiện, phương pháp nêu trên còn bao gồm bước lặp lại phương pháp cho các gói tin bổ sung cho đến khi không gian số trình tự tuyến tính được dùng hết hoặc việc sử dụng không gian số trình tự tuyến tính theo cách khác bị vô hiệu

hóa. Theo một phương án thực hiện, phương pháp nêu trên còn bao gồm, sau khi không gian số tuyến tính được dùng hết hoặc sau khi việc sử dụng không gian số trình tự tuyến tính theo cách khác bị vô hiệu hóa, bước thu được gói tin khác nữa cần được gửi, xác định rằng gói tin khác nữa nêu trên không bao gồm thẻ mà chỉ báo rằng gói tin khác nữa đã được tái tạo. Phương pháp nêu trên còn bao gồm, đáp ứng với bước xác định rằng gói tin khác nữa không bao gồm thẻ mà chỉ báo rằng gói tin khác nữa này đã được tái tạo: bước xác định rằng không gian số trình tự tuần hoàn, thay vì không gian số trình tự tuyến tính, sẽ được sử dụng; đáp ứng với bước xác định rằng không gian số trình tự tuần hoàn sẽ được sử dụng, vô hiệu hóa sự chỉ báo rõ ràng trong gói tin khác nữa mà chỉ báo rằng không gian số trình tự tuyến tính đang sử dụng và bổ sung số trình tự từ không gian số trình tự tuần hoàn vào gói tin khác nữa; và bổ sung thẻ vào gói tin khác nữa mà chỉ báo rằng gói tin khác nữa đã được tái tạo. Phương pháp nêu trên còn bao gồm cung cấp gói tin khác nữa để tái tạo và truyền.

Theo một phương án thực hiện, phương pháp nêu trên còn bao gồm, đáp ứng với bước xác định rằng gói tin khác nữa không bao gồm thẻ mà chỉ báo rằng gói tin khác nữa đã được tái tạo, bước xác định xem liệu chức năng sinh ra trình tự ở nút truyền đã được cài đặt lại hay chưa, cho phép sự chỉ báo rõ ràng trong gói tin khác nữa mà chỉ báo rằng chức năng sinh ra trình tự đã được cài đặt lại đáp ứng với bước xác định rằng chức năng sinh ra trình tự ở nút truyền đã được cài đặt lại, và theo cách khác vô hiệu hóa sự chỉ báo rõ ràng trong gói tin khác nữa mà chỉ báo rằng chức năng sinh ra trình tự đã được cài đặt lại.

Các phương án thực hiện tương ứng của nút truyền cũng được bộc lộ. Theo một phương án thực hiện, nút truyền để tái tạo gói tin hoặc khung được làm thích ứng để thu được gói tin cần được gửi và xác định rằng gói tin nêu trên không bao gồm thẻ mà chỉ báo rằng gói tin đã được tái tạo. Nút truyền còn được làm thích ứng để, đáp ứng với bước xác định rằng gói tin không bao gồm thẻ mà chỉ báo rằng gói tin đã được tái tạo, xác định rằng không gian số trình tự tuyến tính sẽ được sử dụng. Nút truyền còn được làm thích ứng để, đáp ứng với bước xác định rằng không gian số trình tự tuyến tính sẽ được sử dụng, cho phép sự chỉ báo rõ ràng trong gói tin mà chỉ báo rằng không gian số trình tự tuyến tính đang được sử dụng và bổ sung số trình tự từ không gian số trình tự

tuyến tính vào gói tin. Nút truyền còn được làm thích ứng để bổ sung thẻ vào gói tin mà chỉ báo rằng gói tin đã được tái tạo và cung cấp gói tin để tái tạo và truyền.

Theo một phương án thực hiện, nút truyền để tái tạo gói tin hoặc khung bao gồm giao diện mạng và hệ mạch xử lý được kết hợp với giao diện mạng này. Hệ mạch xử lý được tạo cấu hình để làm cho nút truyền thu được gói tin cần được gửi và xác định rằng gói tin nêu trên không bao gồm thẻ mà chỉ báo rằng gói tin đã được tái tạo. Hệ mạch xử lý còn được tạo cấu hình để làm cho nút truyền, đáp ứng với bước xác định rằng gói tin không bao gồm thẻ mà chỉ báo rằng gói tin đã được tái tạo, xác định rằng không gian số trình tự tuyến tính sẽ được sử dụng. Hệ mạch xử lý còn được tạo cấu hình để làm cho nút truyền, đáp ứng với bước xác định rằng không gian số trình tự tuyến tính sẽ được sử dụng, cho phép sự chỉ báo rõ ràng trong gói tin mà chỉ báo rằng không gian số trình tự tuyến tính đang được sử dụng và bổ sung số trình tự từ không gian số trình tự tuyến tính vào gói tin. Hệ mạch xử lý còn được tạo cấu hình để làm cho nút truyền bổ sung thẻ vào gói tin mà chỉ báo rằng gói tin đã được tái tạo và cung cấp gói tin này để tái tạo và truyền.

Theo một phương án thực hiện, phương pháp được thực hiện bởi nút thu nhận để loại bỏ gói tin hoặc khung bao gồm bước thu nhận gói tin từ một dòng trong số các dòng đã tái tạo mà đi qua các đường dẫn tách rời từ nút truyền tới nút thu nhận qua mạng, gói tin nêu trên bao gồm số trình tự. Phương pháp nêu trên còn bao gồm bước xác định rằng gói tin bao gồm sự chỉ báo rõ ràng rằng không gian số trình tự tuyến tính, thay vì không gian số trình tự tuần hoàn, đang được sử dụng, xác định xem liệu gói tin có bao gồm sự chỉ báo rõ ràng rằng chức năng sinh ra trình tự ở nút truyền đã được cài đặt lại hay không, và xác định xem liệu số trình tự của gói tin có nằm bên ngoài phạm vi bỏ qua cài đặt lại (iRIR - Reset ignore range) được kết hợp với việc sử dụng không gian số trình tự tuyến tính hay không. Phương pháp nêu trên còn bao gồm bước chấp nhận gói tin đáp ứng với bước xác định rằng gói tin bao gồm sự chỉ báo rõ ràng rằng không gian số trình tự tuyến tính đang được sử dụng và rằng gói tin này bao gồm sự chỉ báo rõ ràng rằng chức năng sinh ra trình tự ở nút truyền đã được cài đặt lại và số trình tự của gói tin nằm bên ngoài iRIR.

Theo một phương án thực hiện, phương pháp nêu trên còn bao gồm, đáp ứng với bước xác định rằng gói tin bao gồm sự chỉ báo rõ ràng rằng không gian số trình tự tuyến tính đang được sử dụng và rằng gói tin này bao gồm sự chỉ báo rõ ràng rằng chức năng sinh ra trình tự ở nút truyền đã được cài đặt lại và số trình tự của gói tin nằm bên ngoài iRIR, bước cập nhật cửa sổ lịch sử và lịch sử được kết hợp với không gian số trình tự tuyến tính dựa trên số trình tự của gói tin.

Theo một phương án thực hiện, phương pháp nêu trên còn bao gồm, khi xác định rằng gói tin bao gồm sự chỉ báo rõ ràng rằng không gian số trình tự tuyến tính đang được sử dụng và rằng gói tin hoặc không bao gồm sự chỉ báo rõ ràng rằng chức năng sinh ra trình tự ở nút truyền đã được cài đặt lại hoặc số trình tự của gói tin không nằm bên ngoài iRIR, bước xác định xem liệu số trình tự của gói tin có nằm trong cửa sổ lịch sử (iHSW - History window) được kết hợp với việc sử dụng không gian số trình tự tuyến tính hay không, xác định xem liệu số trình tự của gói tin đã nằm trong lịch sử được kết hợp với việc sử dụng không gian số trình tự tuyến tính mà bao gồm các số trình tự từ không gian số trình tự tuyến tính vốn đã thu nhận hay chưa, và chấp nhận gói tin đáp ứng với bước xác định rằng số trình tự của gói tin nằm trong iHSW và xác định rằng số trình tự của gói tin chưa nằm trong lịch sử. Theo một phương án thực hiện, phương pháp nêu trên còn bao gồm bước loại bỏ gói tin đáp ứng với bước xác định rằng số trình tự của gói tin nằm trong iHSW và xác định rằng số trình tự của gói tin đã nằm trong lịch sử. Theo một phương án thực hiện, phương pháp nêu trên còn bao gồm bước cập nhật iHSW và lịch sử được kết hợp với không gian số trình tự tuyến tính dựa trên số trình tự của gói tin đáp ứng với bước xác định rằng số trình tự của gói tin nằm trong iHSW và xác định rằng số trình tự của gói tin chưa nằm trong lịch sử.

Các phương án thực hiện tương ứng của nút thu nhận cũng được bộc lộ. Theo một phương án thực hiện, nút thu nhận để loại bỏ gói tin hoặc khung được làm thích ứng để thu nhận gói tin từ một dòng trong số các dòng đã tái tạo mà đi qua các đường dẫn tách rời từ nút truyền tới nút thu nhận qua mạng, gói tin nêu trên bao gồm số trình tự. Nút thu nhận còn được làm thích ứng để xác định rằng gói tin bao gồm sự chỉ báo rõ ràng rằng không gian số trình tự tuyến tính, thay vì không gian số trình tự tuần hoàn, đang được sử dụng, xác định xem liệu gói tin có bao gồm sự chỉ báo rõ ràng rằng chức năng

sinh ra trình tự ở nút truyền đã được cài đặt lại hay không, và xác định xem liệu số trình tự của gói tin có nằm bên ngoài phạm vi bỏ qua cài đặt lại (iRIR) được kết hợp với việc sử dụng không gian số trình tự tuyến tính hay không. Nút thu nhận còn được làm thích ứng để chấp nhận gói tin đáp ứng với bước xác định rằng gói tin bao gồm sự chỉ báo rõ ràng rằng không gian số trình tự tuyến tính đang được sử dụng và rằng gói tin này bao gồm sự chỉ báo rõ ràng rằng chức năng sinh ra trình tự ở nút truyền đã được cài đặt lại và số trình tự của gói tin nằm bên ngoài iRIR.

Theo một phương án thực hiện, nút thu nhận để loại bỏ gói tin hoặc khung bao gồm giao diện mạng và hệ mạch xử lý được kết hợp với giao diện mạng này. Hệ mạch xử lý được tạo cấu hình để làm cho nút thu nhận thu nhận gói tin từ một dòng trong số các dòng đã tái tạo mà đi qua các đường dẫn tách rời từ nút truyền tới nút thu nhận qua mạng, gói tin nêu trên bao gồm số trình tự. Hệ mạch xử lý còn được tạo cấu hình để làm cho nút thu nhận xác định rằng gói tin bao gồm sự chỉ báo rõ ràng rằng không gian số trình tự tuyến tính, thay vì không gian số trình tự tuần hoàn, đang được sử dụng, xác định xem liệu gói tin có bao gồm sự chỉ báo rõ ràng rằng chức năng sinh ra trình tự ở nút truyền đã được cài đặt lại hay không, và xác định xem liệu số trình tự của gói tin có nằm bên ngoài phạm vi bỏ qua cài đặt lại (iRIR) được kết hợp với việc sử dụng không gian số trình tự tuyến tính hay không. Hệ mạch xử lý còn được tạo cấu hình để làm cho nút thu nhận chấp nhận gói tin đáp ứng với bước xác định rằng gói tin bao gồm sự chỉ báo rõ ràng rằng không gian số trình tự tuyến tính đang được sử dụng và rằng gói tin này bao gồm sự chỉ báo rõ ràng rằng chức năng sinh ra trình tự ở nút truyền đã được cài đặt lại và số trình tự của gói tin nằm bên ngoài iRIR.

Mô tả vắn tắt các hình vẽ

Các hình vẽ kèm theo được hợp nhất trong và tạo thành một phần bản mô tả này minh họa một vài khía cạnh của sáng chế, và cùng với phần mô tả dùng để giải thích các nguyên tắc của sáng chế.

Fig.1 minh họa kịch bản dựa trên đám mây yêu cầu sự cải tiến đối với các chức năng tái tạo và loại bỏ khung cho độ tin cậy (FRER);

Fig.2 minh họa các phạm vi của không gian số trình tự tuần hoàn IEEE 802.1CB-2017 được dùng ở đây để phân tích xác suất rơi gói tin;

Fig.3 minh họa hệ thống bao gồm nút truyền (TX) và nút thu nhận (RX) mà cung cấp FRER theo các phương án thực hiện của sáng chế;

Fig.4 minh họa một ví dụ về mối tương quan của không gian số trình tự (tuyến tính) mới và không gian số trình tự (tuần hoàn) ban đầu được dùng cho FRER theo các phương án thực hiện của sáng chế;

Fig.5 minh họa sơ đồ trạng thái cho chức năng tái tạo ở nút TX theo một phương án thực hiện của sáng chế;

Fig.6 là lưu đồ minh họa sự vận hành của chức năng tái tạo ở nút TX theo một phương án thực hiện của sáng chế;

Fig.7 là lưu đồ minh họa sự vận hành của chức năng tái tạo ở nút TX theo một phương án thực hiện khác của sáng chế;

Fig.8 minh họa sơ đồ trạng thái cho chức năng loại bỏ ở nút RX theo một phương án thực hiện của sáng chế;

Fig.9A và Fig.9B cung cấp lưu đồ minh họa sự vận hành của chức năng loại bỏ ở nút RX theo một phương án thực hiện sáng chế;

Fig.10 minh họa định dạng R-TAG theo IEEE 802.1CB và trong đó “*SeqResetFlag*”, “*InitSeqFlag*”, và “số trình tự mới thuộc về không gian số trình tự tuyến tính” có thể được mã hóa theo một phương án thực hiện của sáng chế; và

Fig.11, Fig.12, và Fig.13 là các sơ đồ khối giản lược của nút mạng.

Mô tả chi tiết sáng chế

Một vài phương án thực hiện được dự tính trong bản mô tả này sẽ được mô tả đầy đủ hơn dưới đây dựa vào các hình vẽ kèm theo. Tuy nhiên, các phương án thực hiện khác được bao hàm trong phạm vi của đối tượng đã được bộc lộ ở đây, đối tượng đã bộc lộ sẽ không được hiểu là bị giới hạn chỉ ở các phương án thực hiện đưa ra ở đây; đúng hơn là, các phương án thực hiện nêu trên được đưa ra bằng ví dụ để truyền tải phạm vi của đối tượng tới người có hiểu biết trung bình trong lĩnh vực kỹ thuật này.

Các phương án thực hiện đưa ra dưới đây thể hiện thông tin để cho phép người có hiểu biết trung bình trong lĩnh vực kỹ thuật này thực hiện các phương án thực hiện và minh họa chế độ tốt nhất để thực hiện các phương án thực hiện này. Khi đọc phần mô tả dưới đây dựa vào các hình vẽ kèm theo, người có hiểu biết trung bình trong lĩnh vực

kỹ thuật này sẽ hiểu các giải pháp của sáng chế và sẽ nhận ra các ứng dụng của các giải pháp không được mô tả cụ thể ở đây. Cần hiểu rằng các giải pháp và các ứng dụng này sẽ nằm trong phạm vi của sáng chế.

Nói chung, tất cả các thuật ngữ được sử dụng ở đây sẽ được hiểu theo nghĩa thông thường của chúng trong lĩnh vực kỹ thuật có liên quan, trừ khi nghĩa khác được đưa ra một cách rõ ràng và/hoặc được ngụ ý từ ngữ cảnh mà nó được sử dụng trong đó. Tất cả các tham chiếu tới một phần tử, thiết bị, bộ phận, phương tiện, bước, v.v. sẽ được hiểu công khai khi dựa vào ít nhất một trường hợp của phần tử, thiết bị, bộ phận, phương tiện, bước, v.v., trừ khi được nêu một cách rõ ràng theo cách khác. Các bước của các phương pháp bất kỳ đã bộc lộ ở đây không cần phải được thực hiện theo thứ tự chính xác đã bộc lộ, trừ khi bước được mô tả một cách rõ ràng là sau hoặc trước một bước khác và/hoặc trong đó không được nêu rõ ràng rằng một bước phải được thực hiện sau hoặc trước một bước khác. Dấu hiệu bất kỳ của phương án thực hiện bất kỳ trong số các phương án thực hiện đã bộc lộ ở đây có thể được áp dụng với phương án thực hiện khác bất kỳ, khi thích hợp. Tương tự như vậy, ưu điểm bất kỳ của phương án thực hiện bất kỳ trong số các phương án thực hiện có thể áp dụng với các phương án thực hiện khác bất kỳ, và ngược lại. Các mục đích, các dấu hiệu, và các lợi ích khác của các phương án thực hiện đã bộc lộ sẽ trở nên rõ ràng từ phần mô tả sau đây.

Nút TSN: Như được sử dụng ở đây, nút nối mạng nhảy với thời gian (TSN) là nút mạng bất kỳ trong mạng TSN. Các ví dụ của nút TSN bao gồm điểm cuối TSN và cầu TSN.

Hiện nay, vẫn đang tồn tại các thách thức nhất định đối với TSN và sự nối mạng tất định (DetNet). Các cơ chế khả dụng để cài đặt lại các chức năng dư thừa liên mạch bao gồm quá nhiều sự rơi gói tin không cần thiết và không sẵn sàng cho đám mây. Mục đích cuối cùng của sự dư thừa liên mạch đó là tránh sự tổn thất gói tin nhiều nhất có thể. Sự rơi gói tin không cần thiết do các vận hành của cơ chế dư thừa liên mạch sẽ được tối thiểu. Hơn nữa, việc di chuyển các bộ phận dư thừa liên mạch (chẳng hạn, viện kỹ sư điện và điện tử (IEEE) 802.1CB) tới môi trường đám mây tạo ra các thách thức liên quan tới tính khả dụng, vì vậy các chức năng dư thừa liên mạch được cài đặt lại thường xuyên hơn nhiều so với trong các môi trường phần cứng trong công nghiệp hiện nay. Do đó,

việc dò và việc thích ứng liên mạch với các kịch bản được gây ra bởi việc cài đặt lại các chức năng dư thừa liên mạch (chẳng hạn, IEEE 802.1CB) là thiết yếu.

Ngoài ra, cơ chế quá thời gian và cửa sổ lịch sử đã mô tả trên đây yêu cầu một thiết kế tốt đối với các tham số có liên quan. Tuy nhiên, đây không phải là những nhiệm vụ tầm thường vì các yêu cầu đối lập phải được đáp ứng. Trong quá trình thiết kế cửa sổ lịch sử, người ta có thể dự tính lựa chọn kích cỡ cửa sổ nhỏ, ví dụ, để bảo vệ các tài nguyên nút loại bỏ hoặc để bảo vệ chống lại các gói tin giả (bảo mật). Ngược lại, các giá trị kích cỡ cửa sổ lớn sẽ dễ chịu hơn đối với các sự cố và các sai số mạng. Việc tìm thấy kích cỡ cửa sổ tối ưu là một nhiệm vụ khó khăn. Theo cách tương tự, việc thiết kế giá trị tham số quá thời gian quá thấp có thể gây ra sự cài đặt lại thường xuyên (và không cần thiết) của chức năng loại bỏ. Mặt khác, giá trị tham số quá thời gian quá lớn làm chậm sự phục hồi sau các kịch bản gặp sự cố và gây ra sự chuyển tiếp nối mạng không mong muốn. Hơn nữa, việc sử dụng sự tái tạo và sự loại bỏ khung cho độ tin cậy (FRER) cho các dòng truyền loạt (tốc độ bit không cố định (CBR)) làm cho việc thiết kế nêu trên trở nên thách thức hơn hoặc thậm chí không thể tìm thấy sự cân bằng phù hợp.

Trong công bố đơn yêu cầu cấp bằng độc quyền sáng chế số WO2021005397A1 trước đây của chúng tôi (dưới đây được xem như “Đơn 397”), giải pháp thông báo rõ ràng được mô tả vốn được dựa trên cờ mới được bao gồm trong R-TAG, cụ thể là “*SeqResetFlag*”. Cờ nêu trên được cài đặt bởi chức năng tái tạo khi chức năng sinh ra trình tự được cài đặt lại, vì vậy các sự kiện nêu trên có thể được nhận ra dễ dàng bởi các chức năng loại bỏ. Với giải pháp nêu trên, chúng ta sẽ tập trung vào kịch bản trong đó các khung có thể bị rơi không cần thiết (với xác suất cao) khi chức năng sinh ra trình tự được cài đặt lại. Giải pháp nêu trên cung cấp một giải pháp tốt hơn nhiều so với IEEE 802.1CB-2017 khi chức năng sinh ra trình tự được cài đặt lại, nhưng nó không thể cung cấp sự phục hồi không tổn thất hoàn toàn trong tất cả các trường hợp. Nó có thể dẫn đến một vài gói tin bị rơi sau sự kiện cài đặt lại.

Các khía cạnh nhất định của sáng chế và các phương án thực hiện của chúng có thể đưa ra các giải pháp với các thách thức đã nói trên đây hoặc các thách thức khác. Các hệ thống và các phương pháp được mô tả ở đây cung cấp sự phục hồi liên mạch cho

các cơ chế dư thừa liên mạch dựa trên số trình tự sau sự kiện cài đặt lại sinh ra số trình tự.

Các phương án thực hiện của giải pháp đã đề xuất ở đây nhằm mục tiêu giải quyết kịch bản trong đó các khung bị rơi không cần thiết bởi các chức năng IEEE 802.1CB-2017 do sự cài đặt lại chức năng sinh ra trình tự.

Các khía cạnh của giải pháp đã đề xuất bao gồm:

1. Thông báo rõ ràng của sự kiện cài đặt lại, và
2. Bổ sung không gian số trình tự ban đầu tuyến tính mới vào không gian số trình tự tuần hoàn hiện đang tồn tại của IEEE 802.1CB-2017.

Chú ý rằng không gian số trình tự tuần hoàn hiện đang tồn tại của IEEE 802.1CB-2017 được nói tới trong sáng chế này là “không gian số trình tự ban đầu”.

Thông báo rõ ràng của sự kiện cài đặt lại được dựa trên cờ được bao gồm trong R-TAG, mà ở đây được xem như “*SeqResetFlag*”. Việc sử dụng không gian số trình tự ban đầu tuyến tính mới (mà ở đây được xem như “*InitSeqNumSpace*”) được ghi chú qua cờ mới được bao gồm trong R-TAG, mà ở đây được xem như “*InitSeqFlag*”. Các giá trị trình tự của không gian số mới (“*InitSeqNumSpace*”) cũng được bao gồm trong R-TAG.

Trong khi các phương án thực hiện đã mô tả ở đây tập trung vào FRER của TSN, các giải pháp đã đề xuất ở đây cũng áp dụng được với PREF của DetNet hoặc các cơ chế dư thừa liên mạch khác dựa trên sự đánh số trình tự hoặc tính năng tương đương (chẳng hạn, được cung cấp bởi các dấu thời gian).

Các phương án thực hiện của giải pháp đã mô tả ở đây cho phép việc đám mây hóa sự dư thừa liên mạch bằng cách cung cấp việc cài đặt lại liên mạch cho việc sinh ra số trình tự (chức năng sinh ra trình tự) qua các cải tiến của, chẳng hạn, chức năng tái tạo và loại bỏ của IEEE 802.1CB-2017. Theo một vài phương án thực hiện, các khía cạnh sau đây được đưa vào (1) cờ mới cho sự chỉ báo rõ ràng của việc cài đặt lại chức năng sinh ra trình tự và (2) việc sử dụng không gian số trình tự ban đầu tuyến tính mới với không gian số trình tự tuần hoàn hiện đang tồn tại của, chẳng hạn, IEEE 802.1CB-2017. Hơn nữa, trong khi phần mô tả được cung cấp ở đây tập trung vào FRER như được xác định trong IEEE 802.1CB, giải pháp đã mô tả ở đây có thể áp dụng được với FRER của

TSN, PREF của DetNet, hoặc các cơ chế dư thừa liên mạch khác dựa trên sự đánh số trình tự hoặc tính năng tương đương (chẳng hạn, được cung cấp bởi các dấu thời gian).

Nhiều phương án thực hiện được đề xuất ở đây để giải quyết một hoặc nhiều vấn đề đã bộc lộ ở đây. Các phương án thực hiện nhất định có thể mang lại một hoặc nhiều ưu điểm kỹ thuật sau đây. Các phương án thực hiện của giải pháp đã đề xuất được mô tả ở đây cho phép việc đám mây hóa sự dư thừa liên mạch bằng cách cung cấp việc cài đặt lại liên mạch cho việc sinh ra số trình tự (chức năng sinh ra trình tự) qua các cải tiến của chức năng tái tạo và loại bỏ của IEEE 802.1CB-2017. Các cải tiến nêu trên đảm bảo sự thích ứng liên mạch và nhanh hơn nhiều với các kịch bản sự cố mạng và bảo vệ chống lại sự rơi gói tin không cần thiết khi việc sinh ra trình tự được cài đặt lại.

Động cơ chính cho các cải tiến nêu trên đó cho phép việc đám mây hóa, nghĩa là, việc thực hiện “gốc đám mây - cloud native” của các chức năng FRER/chức năng tái tạo và loại bỏ gói tin (PREF). Xu hướng ngày nay là di chuyển các ứng dụng về phía các môi trường được ảo hóa. Xu hướng này cũng đã đạt tới các ứng dụng được sử dụng trong các môi trường công nghiệp, chẳng hạn, điện toán biên, điện toán sương mù. Việc sử dụng FRER/PREF trong kịch bản dựa trên đám mây, trong đó các người nói/các người nghe (các nguồn/các điểm đến) được di chuyển tới đám mây, tạo ra nhiều thách thức bổ sung cho các chức năng FRER/PREF. Các FRER/các PREF là các chức năng TSN/DetNet mà thuộc về các điểm cuối, vì vậy FRER/PREF phải hoạt động bên trong đám mây. Ví dụ, FRER/PREF thường là thực thể trong nhóm bộ điều khiển (ctrl-cluster) phục vụ ứng dụng công nghiệp trong đám mây, như được minh họa trên Fig.1. Về mặt này, Fig.1 minh họa kịch bản dựa trên đám mây yêu cầu sự cải tiến đối với các chức năng FRER.

Các hành động đám mây điển hình như chạy nhiều máy ảo (VM - Virtual Machine)/bộ chứa/thực thể, tạo ra VM/bộ chứa/thực thể, di chuyển VM/bộ chứa/thực thể, cài đặt lại chức năng, loại bỏ VM/bộ chứa/thực thể, v.v. yêu cầu các chức năng FRER để thích ứng với việc thay đổi môi trường liên mạch. Các thay đổi trong môi trường nêu trên là thường xuyên hơn nhiều so với trong các kịch bản mạng công nghiệp hiện nay hoặc các triển khai mạng không có đám mây.

Các hệ thống có tính khả dụng cao yêu cầu sự loại bỏ điểm sự cố bất kỳ. Do đó, các chức năng FRER/PREF (nghĩa là, sinh ra trình tự) phải được cải tiến để có thể hỗ trợ các giải pháp dự thừa dành riêng cho đám mây.

Sử dụng FRER của IEEE 802.1CB làm một ví dụ, sự tác động của sự cài đặt lại chức năng sinh ra số trình tự phụ thuộc vào giá trị thực tế của `sequence_number` được sử dụng cho gói tin cuối cùng được gửi trước khi cài đặt lại (được ghi chú ở đây là SN_L) và giá trị của `sequence_number` được sử dụng cho gói tin thứ nhất được gửi sau khi cài đặt lại (được ghi chú ở đây là SN_R). Theo IEEE 802.1CB-2017, SN_R luôn là 0 và SN_L là giá trị trong phạm vi $\{0; \dots; \text{GenSeqSpace} - 1\}$.

Các phạm vi sau đây có thể được xác định để phân tích sự tác động của sự cài đặt lại cho (1) giải pháp đã mô tả trong IEEE 802.1CB-2017, (2) giải pháp đã mô tả trong "Đơn 397", và (3) giải pháp đã mô tả ở đây.

- A: $SN_R > SN_L + d$
- B: $SN_L + d \geq SN_R > SN_L$
- C: $SN_L \geq SN_R > SN_L - d$
- D: $SN_L - d \geq SN_R > SN_L - 2d$
- E: $SN_L - 2d \geq SN_R$

trong đó $d = \text{"frerSeqRcvyHistoryLength"}$ được xác định trong IEEE 802.1CB-2017. Lưu ý rằng do đặc điểm tuần hoàn của không gian số trình tự ban đầu, "A" và "E" là các phạm vi liền kề, trong đó biên giữa chúng được xác định ở đây cho sự phân tích dưới dạng modulo $\text{GenSeqSpace}(SN_L + \text{GenSeqSpace}/2)$. Fig.2 minh họa các phạm vi của không gian số trình tự tuần hoàn.

Giả định cho sự phân tích đó là (i) không có các sự kiện khác trong mạng, chỉ chức năng sinh ra trình tự được cài đặt lại và (ii) "frerSeqRcvyHistoryLength" (d) nhận giá trị 100 cho các phép tính xác suất.

Đối với việc đánh giá giải pháp đã mô tả trong IEEE 802.1CB-2017, phạm vi sau đây là quan trọng:

- cửa sổ lịch sử (HSW) = $\{ SN_L + d; \dots; SN_L - d + 1 \}$, trong đó $d = \text{"frerSeqRcvyHistoryLength"}$.

Chú ý rằng HSW trên thực tế là sự hợp nhất của phạm vi “B” và phạm vi “C” được sử dụng cho việc đánh giá.

Theo IEEE 802.1CB-2017, các gói tin với `sequence_number` ngoài cửa sổ lịch sử (HSW) bị rơi và các gói tin trong cửa sổ lịch sử được đánh giá dựa vào “SequenceHistory” để quyết định xem liệu chúng có phải là các phân lặp hay không. Do đó, IEEE 802.1CB-2017 hoạt động như sau cho mỗi một phạm vi trong số các phạm vi từ A tới E:

A: các gói tin bị rơi cho đến khi bộ định thời hết thời hạn hoặc `sequence_number` của các gói tin đạt tới $SN_L + 1$,

B: không có sự rơi gói tin, SN_R được chấp nhận,

C: các gói tin bị rơi cho đến khi `sequence_number` của các gói tin đạt tới $SN_L + 1$,

D: các gói tin bị rơi cho đến khi `sequence_number` của các gói tin đạt tới $SN_L + 1$,

E: các gói tin bị rơi cho đến khi bộ định thời hết thời hạn hoặc `sequence_number` của các gói tin đạt tới $SN_L + 1$.

Nghĩa là, có xác suất cao sẽ xuất hiện sự rơi gói tin (99,8%) trong hầu hết các trường hợp cho đến khi sự quá thời gian kích khởi việc chấp nhận gói tin tiếp theo. Chú ý rằng sự rơi gói tin nghĩa là sự tổn thất gói tin đối với ứng dụng mà sự vận hành của chúng có thể bị hư hại bởi các gói tin bị mất.

Đối với việc đánh giá giải pháp đã mô tả trong "Đơn 397", phạm vi bổ sung sau đây là quan trọng:

- phạm vi bỏ qua cài đặt lại (RIR) = $\{ SN_L + d; \dots; SN_L - 2d + 1 \}$, trong đó $d = \text{"frerSeqRcvyHistoryLength"}$

Chú ý rằng RIR trên thực tế là sự hợp nhất của phạm vi “B”, phạm vi “C”, và phạm vi “D” được sử dụng cho việc đánh giá.

RIR là phạm vi trong đó chức năng loại bỏ sẽ bỏ qua “*SeqResetFlag*” đã thu nhận và kiểm tra gói tin đã thu nhận dựa vào HSW. Chú ý rằng phạm vi “D” cũng là một phần của RIR, vì các phân lặp trên các đường dẫn dư thừa chậm hơn có thể có `sequence_number` bên dưới SN_L và có thể thấp tới $SN_L - d$. Do đó, giải pháp đã mô tả trong "Đơn 397" vận hành như sau cho mỗi một phạm vi trong số các phạm vi:

A: không có sự rơi gói tin, SN_R được chấp nhận,

B: không có sự rơi gói tin, SN_R được chấp nhận,

C: có sự rơi gói tin cho đến khi `sequence_number` của các gói tin đạt tới $SN_L + 1$,

D: có sự rơi gói tin cho đến khi `sequence_number` của các gói tin đạt tới $SN_L + 1$,

E: không có sự rơi gói tin, SN_R được chấp nhận.

Có xác suất thấp sẽ xuất hiện sự rơi gói tin (0,3%) trong hầu hết các trường hợp. Khi sự rơi gói tin xảy ra, nó được giới hạn ở giá trị lớn nhất bằng “2d” số lượng các gói tin sau khi cài đặt lại. Một vài ứng dụng TSN có thể cho phép điều này khi sự cài đặt lại xuất hiện, nhưng một vài ứng dụng thì không thể.

Giải pháp đã mô tả ở đây có các đặc điểm sau đây cho mỗi một phạm vi trong số các phạm vi:

A: không có sự rơi gói tin, SN_R được chấp nhận,

B: không có sự rơi gói tin, SN_R được chấp nhận,

C: không có sự rơi gói tin, SN_R được chấp nhận,

D: không có sự rơi gói tin, SN_R được chấp nhận,

E: không có sự rơi gói tin, SN_R được chấp nhận.

Không có sự rơi gói tin. Các gói tin được gửi sau khi cài đặt lại ngay lập tức được xem là có hiệu lực. Các ứng dụng TSN hoàn toàn không bị ảnh hưởng, và sự tác động của việc thực hiện lên các nút FRER là trung bình.

Các hệ thống và các phương pháp được bộc lộ ở đây để cải thiện chức năng loại bỏ trong mạng TSN sử dụng FRER theo IEEE 802.1CB (hoặc tương tự như vậy trong mạng DetNet sử dụng các PREF). Chú ý rằng sự thảo luận ở đây sử dụng thuật ngữ và các tên biến số trong IEEE 802.1CB nếu thích hợp, được biểu thị là “*VariableName*”. Các biến số, các chức năng, và các tham số mới tuân theo các quy ước đặt tên trong IEEE 802.1CB và được kí hiệu là “*NewEntityName*”.

Từ bây giờ, phần mô tả tập trung vào các phương án thực hiện của giải pháp đã đề xuất cho IEEE 802.11 FRER và như vậy sử dụng các thuật ngữ và các ký hiệu như được mô tả trong 802.1CB-2017 FRER. Tuy nhiên, giải pháp đã đề xuất ở đây có thể áp dụng được tương đương với các cơ chế dư thừa liên mạch khác (chẳng hạn, PREF cho DetNet) dựa trên sự đánh số trình tự hoặc tính năng tương đương (chẳng hạn, cung cấp bởi các dấu thời gian).

Fig.3 minh họa hệ thống 300 bao gồm nút truyền (TX) 302 và nút thu nhận (RX) 304 trong đó nút TX 302 truyền dòng đã tái tạo của các gói tin tới nút RX 304 qua mạng TSN 306. Chú ý rằng nút TX 302 và nút RX 304 có thể là, ví dụ, các điểm cuối TSN, các cầu TSN, hoặc kiểu khác bất kỳ của nút TSN, theo ví dụ này. Việc truyền dòng đã tái tạo của các gói tin bao hàm việc tái tạo dòng các gói tin thành nhiều dòng thành phần để nhờ đó cung cấp dòng hỗn hợp. Các dòng thành phần sau đó được truyền tới nút RX 304 qua mạng TSN 306 qua các đường dẫn tách rời lớn nhất. Lưu ý rằng trong khi các nút 302 và 304 được kí hiệu ở đây là “nút TX” và “nút RX”, theo cách tương ứng, thì cần hiểu rằng các nút này đều có thể truyền và thu nhận các dòng qua mạng TSN 306.

Như được minh họa, nút TX 302 bao gồm chức năng FRER 308 mà vận hành để cung cấp FRER theo, theo ví dụ này, IEEE 802.1CB. FRER 308 bao gồm chức năng tái tạo 310 và chức năng loại bỏ 312 (đã minh họa dưới dạng tùy chọn theo nghĩa là nó không được sử dụng để truyền dòng tới nút RX 304). Theo cách tương tự, nút RX 304 bao gồm chức năng FRER 314 mà vận hành để cung cấp FRER theo, theo ví dụ này, IEEE 802.1CB. FRER 314 bao gồm chức năng tái tạo 316 (đã minh họa dưới dạng tùy chọn theo nghĩa là nó không được sử dụng để thu nhận dòng từ nút TX 302) và chức năng loại bỏ 322.

Như được minh họa, chức năng tái tạo 310 bao gồm chức năng sinh ra trình tự 320. Chức năng sinh ra trình tự 320 vận hành để sinh ra các số trình tự cho các gói tin trong dòng gói tin. Chức năng loại bỏ 312 bao gồm chức năng phục hồi trình tự 322. Chức năng phục hồi trình tự 322 vận hành trên các gói tin được chuyển lên ngăn xếp giao thức về phía các chức năng lớp cao hơn và sử dụng tham số phụ số trình tự để quyết định các gói tin nào sẽ được chuyển và các gói tin nào sẽ được loại bỏ.

Theo một vài phương án thực hiện, nút TX 302 là một phần của việc thực hiện đám mây (chẳng hạn, một phần của Ctrl-cluster như được mô tả trên đây đối với Fig.1).

Chức năng sắp xếp trình tự của các FRER 308 và 314 cung cấp tham số phụ “sequence_number” cho các chức năng FRER. Cụ thể là, chức năng sắp xếp trình tự có hai kiểu chức năng thành phần: (1) chức năng sinh ra trình tự (chẳng hạn, chức năng sinh ra trình tự 320) mà vận hành trên các gói tin được chuyển xuống ngăn xếp giao thức về phía lớp vật lý và sinh ra giá trị cho tham số phụ sequence_number và (2) chức

năng phục hồi trình tự (chẳng hạn, chức năng phục hồi trình tự 322) mà vận hành trên các gói tin được chuyển lên ngăn xếp giao thức về phía các chức năng lớp cao hơn và sử dụng tham số phụ *sequence_number* của các gói tin đã thu nhận để quyết định các gói tin nào sẽ được chuyển và các gói tin nào sẽ được loại bỏ.

Các phương án thực hiện của giải pháp đã mô tả ở đây sử dụng cờ được đưa vào bởi "Đơn 379" với R-TAG, cụ thể là "*SeqResetFlag*". Cờ này được sử dụng như sau:

- Khi truyền dòng các gói tin, chức năng tái tạo 310 ở nút TX 302 không thay đổi giá trị "*SeqResetFlag*" của gói tin nếu thẻ dư thừa (R-TAG) đã được bao gồm trong phần đầu của gói tin (nghĩa là, trong phần đầu của khung Ethernet tương ứng).
- Chức năng tái tạo 310 ở nút TX 302 cài đặt "*SeqResetFlag*" (chẳng hạn, cài đặt giá trị "*SeqResetFlag*" = 1) cho khoảng thời gian nhất định hoặc cho nhiều các gói tin đã gửi khi chức năng sinh ra trình tự 320 được cài đặt lại.
- Chức năng tái tạo 310 ở nút TX 302 xóa "*SeqResetFlag*" (chẳng hạn, cài đặt giá trị "*SeqResetFlag*" = 0) trong tất cả các trường hợp khác.

Các phương án thực hiện của giải pháp đã mô tả ở đây đưa vào không gian số trình tự bổ sung mới, mà ở đây được xem như "*InitSeqNumSpace*", vốn được sử dụng sau khi khởi tạo hoặc cài đặt lại chức năng sinh ra trình tự 320. *InitSeqNumSpace* mới đưa vào được minh họa bởi không gian số trình tự tuyến tính in đậm trên Fig.4. *sequence_number* của gói tin tiếp theo được lưu trữ trong biến số mới, cụ thể là "*InitGenSeqNum*". Khi không gian số mới nêu trên được dùng hết, chức năng sinh ra trình tự 320 bắt đầu sử dụng không gian số trình tự ban đầu, vốn được minh họa bởi không gian số trình tự không in đậm trên Fig.4. Nói theo cách khác, Fig.4 minh họa một ví dụ về mối tương quan của không gian số trình tự (tuyến tính) mới và không gian số trình tự (tuần hoàn) ban đầu.

Các phương án thực hiện của giải pháp đã mô tả ở đây đưa vào cờ mới trong R-TAG, mà được xem như "*InitSeqFlag*". Cờ nêu trên được sử dụng bởi chức năng tái tạo 310 ở nút TX 302 như sau:

- Đối với mỗi gói tin trong dòng các gói tin đang được truyền bởi nút TX 302, chức năng tái tạo 310 không thay đổi giá trị “*InitSeqFlag*” của gói tin nếu R-TAG đã được bao gồm trong phần đầu của gói tin.
- Chức năng tái tạo 310 cài đặt “*InitSeqFlag*” (chẳng hạn, cài đặt giá trị “*InitSeqFlag*” =1) khi chức năng sinh ra trình tự 320 được khởi tạo hoặc cài đặt lại. “*InitSeqNumSpace*” được sử dụng bởi chức năng tái tạo 310 để sinh ra số trình tự cho gói tin.
- Chức năng tái tạo 310 xóa “*InitSeqFlag*” (chẳng hạn, cài đặt giá trị “*InitSeqFlag*” =0) trong tất cả các trường hợp khác.

“*InitSeqNumSpace*” được sử dụng như sau:

- “*InitSeqNumSpace*” là không gian số trình tự tuyến tính bắt đầu với “*InitSeqStart*” (chẳng hạn, tham số tạo cấu hình được cài đặt bằng, trong ví dụ trên Fig.4) 32768 và kết thúc với “*GenSeqSpace-1*” (nghĩa là, 65535 theo IEEE 802.1CB-2017).
- “*InitSeqNumSpace*” có tập hợp các biến số của chính nó mà được sử dụng để đối phó với *sequence_number* của các gói tin. Chú ý rằng các tên biến số *InitSeqNumSpace* được dẫn xuất từ các tên biến số của số trình tự ban đầu bằng cách nối chúng với tiền tố “*Init*”, chẳng hạn, “*InitGenSeqNum*”, “*InitRecovSeqNum*”, “*InitSequenceHistory*”, “*InitTakeAny*”, “*InitRemainingTicks*”, v.v.
- Chức năng tái tạo 310 ở nút TX 302 sử dụng “*InitSeqNumSpace*” như sau:
 - “*UseInitSeqSpace*” là biến số trạng thái Boole thể hiện rằng “*InitSeqNumSpace*” mới (giá trị=1) phải được sử dụng bởi chức năng sinh ra trình tự 320 để sinh ra *sequence_number* cho gói tin tiếp theo.
 - Mỗi khi chức năng sinh ra trình tự 320 được khởi tạo hoặc cài đặt lại, “*UseInitSeqSpace*” được cài đặt (chẳng hạn, giá trị=1) để thể hiện rằng “*InitSeqNumSpace*” phải được sử dụng để sinh ra *sequence_number* cho các gói tin đã gửi.

- Tất cả các gói tin được gửi với `sequence_number` từ “*InitSeqNumSpace*” có trong R-Tag của chúng tập hợp “*InitSeqFlag*” (chẳng hạn, giá trị=1).
- “*InitGenSeqNum*” nhận giá trị của “*InitSeqStart*” mỗi khi chức năng sinh ra trình tự 320 được khởi tạo hoặc cài đặt lại, và được tăng thêm 1 khi gói tin được gửi.
- Khi “*InitGenSeqNum*” được tăng và trở nên lớn hơn giá trị lớn nhất của nó, “*UseInitSeqSpace*” được xóa (chẳng hạn, giá trị =0) để thể hiện rằng “*InitSeqNumSpace*” không thể được sử dụng. Chức năng sinh ra trình tự 320 sau đó sử dụng không gian số trình tự ban đầu cho gói tin tiếp theo.
- Nếu “*UseInitSeqSpace*” được xóa (giá trị =0), chức năng sinh ra trình tự 320 sử dụng “*GenSeqNum*” để sinh ra `sequence_number` của các gói tin tiếp theo (nghĩa là, không gian số trình tự ban đầu được sử dụng).

Các phương án thực hiện của giải pháp ở đây cũng xác định thủ tục mới cho chức năng loại bỏ 318 ở nút RX 304. Nó sử dụng tập hợp mới của các biến số để xử lý các gói tin chứa `sequence_number` từ “*InitSeqNumSpace*” đã mô tả trên đây. Kích cỡ của cửa sổ lịch sử là tương tự cho cả hai không gian số trình tự. Chức năng loại bỏ 318 hoạt động như sau khi thu nhận gói tin từ một dòng trong số các dòng thành phần của dòng được truyền bởi nút TX 302:

- Nếu còn “*InitSeqFlag*” trong gói tin đã thu nhận được cài đặt (chẳng hạn, giá trị =1), chức năng phục hồi trình tự 322 sử dụng các biến số có liên quan tới không gian số trình tự (tuyến tính) mới (*InitSeqNumSpace*) theo cách sau đây:
 - Chức năng phục hồi trình tự 322 cài đặt “*InitRecovSeqNum*” ở “`sequence_number`” của gói tin và xóa “*InitSequenceHistory*”, nếu “*SeqResetFlag*” được cài đặt và “`sequence_number`” của gói tin nằm ngoài phạm vi iRIR { “*InitRecovSeqNum*” + “*frerSeqRcvyHistoryLength*”; ..., “*InitRecovSeqNum*” - 2 x

- “*frerSeqRcvyHistoryLength*” + 1 }. Chú ý rằng phạm vi nêu trên phải được tính toán theo các biên của không gian số trình tự tuyến tính mới.
- Chức năng phục hồi trình tự 322 bỏ qua “*SeqResetFlag*” nếu “*sequence_number*” của gói tin nằm trong phạm vi iRIR. Nếu “*sequence_number*” nằm ngoài phạm vi iHSW { “*InitRecovSeqNum*” + “*frerSeqRcvyHistoryLength*”; ...; “*InitRecovSeqNum*” - “*frerSeqRcvyHistoryLength*” + 1 } gói tin bị rơi. Nếu nó nằm trong iHSW, thì nó được kiểm tra xem liệu gói tin đã được thu nhận hay chưa.
 - Chức năng phục hồi trình tự 322 không sự thay đổi giá trị “*SeqResetFlag*” và “*InitSeqFlag*” của gói tin.
 - Chức năng phục hồi trình tự 322 cài đặt biến số “*TakeAny*” của không gian số trình tự ban đầu nếu “*sequence_number*” của gói tin nằm trong phạm vi sau đây { “*GenSeqSpace*” - “*frerSeqRcvyHistoryLength*”; ...; “*GenSeqSpace*” - 2 x “*frerSeqRcvyHistoryLength*” }. Chú ý rằng phạm vi nêu trên thể hiện rằng không gian số trình tự khởi tạo mới sẽ sớm được dùng hết, nghĩa là, tiếp cận không gian số trình tự ban đầu.
 - Nếu cờ “*InitSeqFlag*” trong gói tin đã thu nhận được xóa (giá trị =0), thì chức năng phục hồi trình tự sử dụng các biến số có liên quan tới không gian số trình tự ban đầu.
 - Chức năng phục hồi trình tự 322 cài đặt “*RecovSeqNum*” ở giá trị của “*sequence_number*” của gói tin và xóa “*SequenceHistory*” nếu “*SeqResetFlag*” được cài đặt và “*sequence_number*” của gói tin nằm ngoài phạm vi RIR { “*RecovSeqNum*” + “*frerSeqRcvyHistoryLength*”; ...; “*RecovSeqNum*” - 2 x “*frerSeqRcvyHistoryLength*” + 1 }. Chú ý rằng phạm vi nêu trên phải được tính toán theo các quy tắc môđulo của không gian số trình tự ban đầu.
 - Chức năng phục hồi trình tự 322 bỏ qua “*SeqResetFlag*” nếu “*sequence_number*” của gói tin nằm trong phạm vi RIR. Nếu

“sequence_number” nằm ngoài phạm vi HSW, thì gói tin bị rơi. Nếu nó nằm trong HSW, thì kiểm tra xem liệu gói tin đã được thu nhận hay chưa.

- Chức năng phục hồi trình tự 322 không thay đổi giá trị “SeqResetFlag” của gói tin.

Lưu ý rằng cơ chế quá thời gian cho sự phục hồi trình tự – để chấp nhận gói tin tiếp theo, bất kể giá trị của tham số phụ sequence_number của nó (xem biến số “TakeAny” biến số trong 802.1CB-2017 cho không gian số trình tự ban đầu) – không được thay đổi và được áp dụng cho cả hai không gian số trình tự.

Một cách thức thực hiện có thể sử dụng hai biến số để cho phép/vô hiệu hóa việc sử dụng cờ cài đặt lại (“UsingResetFlag” = 1/0 (cho phép/vô hiệu hóa)) và không gian số trình tự khởi tạo mới (“UsingInitSpace” = 1/0 (cho phép/vô hiệu hóa)).

Fig.5 minh họa sơ đồ trạng thái cho chức năng tái tạo 310 ở nút TX 302 theo một phương án thực hiện của sáng chế. Như được minh họa, khi trong trạng thái thứ nhất trong đó chức năng tái tạo 310 sử dụng không gian số trình tự ban đầu và *UsingInitSpace* = Sai, chức năng tái tạo 310 sử dụng không gian số trình tự ban đầu. Khi cài đặt lại với không gian số trình tự ban đầu bị vô hiệu hóa, chức năng tái tạo 312 cài đặt *SeqGenNum* = 0 và *UseInitSeqNum* = Sai và vẫn ở trạng thái thứ nhất. Tuy nhiên, khi cài đặt lại với không gian số trình tự ban đầu được cho phép, chức năng tái tạo 310 cài đặt *InitSeqGenNum* = *InitSeqStart* và *UseInitSeqNum* = True và chuyển tiếp sang trạng thái thứ hai trong đó chức năng tái tạo 310 sử dụng không gian số trình tự ban đầu. Trong khi ở trạng thái thứ hai, khi cài đặt lại với không gian số trình tự ban đầu được cho phép, chức năng tái tạo 310 cài đặt *InitSeqGenNum* = *InitSeqStart* và *UseInitSeqNum* = True và vẫn ở trạng thái thứ hai trong đó chức năng tái tạo 310 sử dụng không gian số trình tự ban đầu. Trong khi ở trạng thái thứ hai, khi dùng hết không gian số trình tự ban đầu, chức năng tái tạo cài đặt *SeqGenNum* = 0 và *UseInitSeqNum* = Sai và chuyển tiếp sang trạng thái thứ nhất trong đó chức năng tái tạo sử dụng không gian số trình tự ban đầu. Trong khi ở trạng thái thứ hai, khi không gian số trình tự ban đầu bị vô hiệu hóa, chức năng tái tạo cài đặt *SeqGenNum* = 0 và *UseInitSeqNum* = Sai và chuyển tiếp sang trạng thái thứ nhất trong đó chức năng tái tạo sử dụng không gian số trình tự ban đầu.

Fig.6 là lưu đồ minh họa sự vận hành của chức năng tái tạo 310 ở nút TX 302 theo một phương án thực hiện của sáng chế. Các bước tùy chọn được thể hiện bằng các đường nét đứt. Như được minh họa, chức năng tái tạo 310 thu nhận gói tin (chẳng hạn, từ lớp (các lớp) cao hơn trong ngăn xếp giao thức) cần được gửi (bước 600). Chức năng tái tạo 310 xác định xem liệu R-Tag có xuất hiện hoặc được cài đặt trong gói tin hay không (bước 602). Nếu vậy (602, CÓ), gói tin với R-Tag đã cài đặt là sẵn sàng được gửi, và như vậy được cung cấp tới lớp dưới trong ngăn xếp giao thức để truyền (bước 604). Ví dụ, nhiều bản sao của gói tin sau đó có thể được sinh ra và được gửi qua các đường dẫn (chẳng hạn, tách rời) khác nhau qua mạng TSN 306. Nếu R-Tag không xuất hiện hoặc được cài đặt trong gói tin (bước 602, KHÔNG), chức năng tái tạo 310 xác định xem liệu “UsingResetFlag” có được cho phép hay không (bước 606). Nếu vậy, chức năng tái tạo 310 xác định xem liệu chức năng sinh ra trình tự 320 gần đây đã được cài đặt lại hay chưa (bước 608). Điều này là đúng nếu sự cài đặt lại đã xuất hiện trong lượng thời gian được xác định trước hoặc được tạo cấu hình trước trước khi thu nhận gói tin (nghĩa là, thời gian hiện tại) hoặc trong số lượng các gói tin được xác định trước hoặc được tạo cấu hình trước trước khi thu nhận gói tin. Nếu chức năng sinh ra trình tự 320 gần đây đã được cài đặt lại (bước 608, CÓ), thì chức năng tái tạo 310 cho phép “SeqResetFlag” (chẳng hạn, cài đặt nó về giá trị bằng 1”) (bước 610). Theo cách khác, chức năng tái tạo 310 vô hiệu “SeqResetFlag” (chẳng hạn, cài đặt nó về giá trị bằng 0”) (bước 612).

Dù tiến tới từ bước 610 hay bước 612, thì chức năng tái tạo 310 vẫn xác định xem liệu việc sử dụng “InitSeqNumSpace” có được cho phép hay không (bước 614). Nếu không (614, KHÔNG), quá trình tiến tới bước 632, mà được mô tả bên dưới. Theo cách khác (bước 614, CÓ), chức năng tái tạo 310 xác định xem liệu “UseInitSeqNum” có được cho phép hay không (chẳng hạn, được cài đặt là “Đúng”) (bước 616). Nếu vậy (bước 616, CÓ), chức năng tái tạo 310 cho phép “InitSeqFlag” (chẳng hạn, cài đặt nó là “1”) (bước 618), bổ sung số trình tự (seq_num) bằng với “InitGenSeqNum” vào gói tin (bước 620), tăng “InitGenSeqNum” (bước 622), và xác định xem liệu “InitSeqNumSpace” đã được dùng hết hay chưa (bước 624). Nếu không (bước 624, KHÔNG), quá trình tiến tới bước 628, mà được mô tả bên dưới. Theo cách khác (bước 624, CÓ), chức năng tái tạo 310 cài đặt “GenSeqNum” bằng “0” và cài đặt

“UseInitSeqNum” là Sai (bước 626). Sau đó, dù tiến tới từ nhánh “KHÔNG” của bước 624 hay bước 626, chức năng tái tạo 310 bổ sung R-Tag vào gói tin (bước 628) và thủ tục tiến tới bước 604 trong đó gói tin là sẵn sàng để gửi.

Trở lại bước 616, nếu “UseInitSeqNum” không được cài đặt là “ĐÚNG”, thì chức năng tái tạo 310 vô hiệu “initSeqFlag” (chẳng hạn, cài đặt nó là “0”) (bước 630), bổ sung số trình tự (seq_num) bằng “GenSeqNum” vào gói tin (bước 632), và tăng “GenSeqSum” (bước 634). Thủ tục sau đó tiến tới bước 628 trong đó R-Tag được bổ sung vào gói tin và sau đó gói tin là sẵn sàng được gửi.

Fig.7 là lưu đồ minh họa sự vận hành của chức năng tái tạo 310 ở nút TX 302 theo một phương án thực hiện khác của sáng chế. Phương án thực hiện này là tương tự với phương án thực hiện trên Fig.6. Như được minh họa, chức năng tái tạo 310 xác định rằng chức năng sinh ra trình tự 310 ở nút TX 302 đã được cài đặt lại (bước 700). Đáp ứng với bước xác định rằng chức năng sinh ra trình tự 310 ở nút TX 302 đã được cài đặt lại, chức năng tái tạo 310 truyền các gói tin thứ nhất trong dòng các gói tin, trong đó: (a) mỗi một gói tin trong số các gói tin thứ nhất bao gồm số trình tự tương ứng từ không gian số trình tự tuyến tính và (b) ít nhất một gói tin thứ nhất từ trong số các gói tin thứ nhất mà được gửi sau khi cài đặt lại còn bao gồm bộ chỉ báo rõ ràng của sự cài đặt lại (bước 702). Chức năng tái tạo 310 xác định rằng đầu cuối của không gian số trình tự tuyến tính đã được đạt tới hoặc theo cách khác rằng việc sử dụng không gian số trình tự tuyến tính đã bị vô hiệu hóa (bước 704). Đáp ứng với bước xác định rằng đầu cuối của không gian số trình tự tuyến tính đã được đạt tới hoặc theo cách khác rằng việc sử dụng không gian số trình tự tuyến tính đã bị vô hiệu hóa, chức năng tái tạo 310 truyền các gói tin thứ hai trong dòng các gói tin, trong đó (a) mỗi một gói tin trong số các gói tin thứ hai bao gồm số trình tự tương ứng của không gian số trình tự tuần hoàn (bước 706).

Theo một phương án thực hiện, mỗi một gói tin trong số các gói tin thứ nhất còn bao gồm sự chỉ báo rõ ràng rằng không gian số trình tự tuyến tính đang được sử dụng.

Theo một phương án thực hiện, mạng nêu trên là mạng TSN. Ngoài ra, theo một phương án thực hiện, phương pháp nêu trên còn bao gồm bước cài đặt lại chức năng sinh ra trình tự 310, trong đó bước cài đặt lại chức năng sinh ra trình tự 310 bao gồm cài đặt lại lịch sử số trình tự, cửa sổ lịch sử (chẳng hạn, cài đặt lại “RecovSeqNum”, vốn là

trung điểm của cửa sổ lịch sử), hoặc cả lịch sử số trình tự và cửa sổ lịch sử. Theo một phương án thực hiện, các bước gồm xác định (700) rằng chức năng sinh ra trình tự 310 ở nút TX 302 đã được cài đặt lại, truyền (702) các gói tin thứ nhất trong dòng các gói tin, xác định (704) rằng đầu cuối của không gian số trình tự tuyến tính đã được đạt tới hoặc theo cách khác rằng việc sử dụng không gian số trình tự tuyến tính đã bị vô hiệu hóa, và truyền (706) các gói tin thứ hai trong dòng các gói tin được thực hiện bằng chức năng FRER 308 của nút TX 302 và, cụ thể hơn, bằng chức năng tái tạo 310 của nút TX 302.

Theo một phương án thực hiện khác, mạng nêu trên là mạng DetNet. Ngoài ra, theo một phương án thực hiện, các bước gồm xác định (700) rằng chức năng sinh ra trình tự 310 ở nút TX 302 đã được cài đặt lại, truyền (702) các gói tin thứ nhất trong dòng các gói tin, xác định (704) rằng đầu cuối của không gian số trình tự tuyến tính đã được đạt tới hoặc theo cách khác rằng việc sử dụng không gian số trình tự tuyến tính đã bị vô hiệu hóa, và truyền (706) các gói tin thứ hai trong dòng các gói tin được thực hiện bởi PRER của nút TX 302.

Fig.8 minh họa sơ đồ trạng thái cho một phương án thực hiện ví dụ của chức năng loại bỏ 318 ở nút RX 304. Như được minh họa, khi trong trạng thái thứ nhất, chức năng loại bỏ 318 sử dụng không gian số trình tự ban đầu. Khi trong trạng thái gói tin thứ nhất và gói tin được thu nhận với *InitSeqNumFlag* = 0 hoặc gói tin được thu nhận và không gian số trình tự ban đầu bị vô hiệu, chức năng loại bỏ 318 vẫn ở trạng thái thứ nhất. Tuy nhiên, khi ở trạng thái thứ nhất và gói tin được thu nhận với *InitSeqNumFlag* = 1 và không gian số trình tự ban đầu được cho phép, chức năng loại bỏ 318 chuyển tiếp sang trạng thái thứ hai trong đó chức năng loại bỏ 318 sử dụng không gian số trình tự ban đầu. Khi ở trạng thái thứ hai và gói tin được thu nhận với *InitSeqNumFlag* = 1 và không gian số trình tự ban đầu được cho phép, chức năng loại bỏ 318 vẫn ở trạng thái thứ hai. Khi ở trạng thái thứ hai và gói tin được thu nhận với *InitSeqNumFlag* = 0 hoặc gói tin được thu nhận và không gian số trình tự ban đầu bị vô hiệu, chức năng loại bỏ 318 chuyển tiếp sang trạng thái thứ nhất.

Fig.9A và Fig.9B cung cấp lưu đồ minh họa sự vận hành của chức năng loại bỏ 318 ở nút RX 304 theo một phương án thực hiện sáng chế. Các bước tùy chọn được thể

hiện bằng các đường nét đứt. Như được minh họa, chức năng loại bỏ 318 thu nhận gói tin (chẳng hạn, từ lớp (các lớp) thấp hơn trong ngăn xếp giao thức) cần được xử lý (bước 900). Gói tin nêu trên có giá trị “seq_num”. Chức năng loại bỏ 318 xác định xem liệu “UsingResetFlag” có được cho phép hay không (bước 902). Nếu không (bước 902, KHÔNG), chức năng loại bỏ 318 xác định xem liệu số trình tự của gói tin có nằm trong cửa sổ lịch sử (HSW) hay không (bước 904). Nếu không (bước 904, KHÔNG), chức năng loại bỏ 318 xác định xem liệu “TakeAny” có được cho phép hay không (bước 904). Nếu không (bước 906, KHÔNG), chức năng loại bỏ 318 thả rơi gói tin (bước 908). Theo cách khác, nếu “TakeAny” được cho phép (bước 906, CÓ), thủ tục tiến tới bước 912, mà được mô tả bên dưới. Trở lại bước 904, nếu số trình tự của gói tin nằm trong HSW (bước 904, CÓ), chức năng loại bỏ 318 xác định xem liệu số trình tự của gói tin đã nằm trong lịch sử hay chưa (bước 910). Nếu vậy (bước 910, CÓ), chức năng loại bỏ 318 thả rơi gói tin (bước 908). Theo cách khác, nếu số trình tự của gói tin chưa nằm trong lịch sử (bước 910, KHÔNG), thủ tục tiến tới bước 912, mà được mô tả bên dưới.

Trở lại bước 902, nếu “UsingResetFlag” được cho phép (bước 902, CÓ), chức năng loại bỏ 318 xác định xem liệu việc sử dụng “InitSeqNumSpace” có được cho phép hay không (bước 920). Nếu không (bước 920, KHÔNG), chức năng loại bỏ 318 xác định xem liệu “SeqResetFlag” có được cho phép hay không và số trình tự của gói tin đã thu nhận có nằm ngoài RIR hay không (bước 922). Nếu không (bước 922, KHÔNG), thủ tục tiến tới bước 904. Theo cách khác (bước 922, CÓ), thủ tục tiến tới bước 912.

Dù tiến tới từ nhánh CÓ của bước 906, nhánh KHÔNG của bước 910, hay nhánh CÓ của bước 922, thì chức năng loại bỏ 318 sau đó vẫn cập nhật “RecovSeqNum” (bước 912), cập nhật “SequenceHistory” (bước 914), xóa “TakeAny” (bước 916), và chấp nhận gói tin (bước 918).

Trở lại bước 920, nếu việc sử dụng “InitSeqNumSpace” được cho phép (bước 920, CÓ), thì chức năng loại bỏ 318 xác định xem liệu “InitSeqFlag” có được cho phép hay không (bước 924). Nếu không (bước 924, KHÔNG), quá trình tiến tới bước 922. Theo cách khác (bước 924, CÓ), chức năng loại bỏ 318 xác định xem liệu “SeqResetFlag” có được cho phép hay không và số trình tự của gói tin đã thu nhận có nằm ngoài iRIR hay không (bước 926). Nếu vậy (bước 926, CÓ), thủ tục tiến tới bước 932, mà được mô tả

bên dưới. Theo cách khác (bước 926, KHÔNG), chức năng loại bỏ 318 xác định xem liệu số trình tự của gói tin có nằm trong iHSW hay không (bước 928). Nếu vậy (bước 928, CÓ), chức năng loại bỏ 318 xác định xem liệu gói tin đã nằm trong lịch sử hay chưa (bước 930). Nếu không (bước 930), thủ tục tiến tới bước 932. Theo cách khác (bước 932, KHÔNG), chức năng loại bỏ 318 thả rơi gói tin (bước 944). Trở lại bước 928, nếu số trình tự của gói tin không nằm trong iHSW (bước 928, KHÔNG), thì chức năng loại bỏ 318 xác định xem liệu “InitTakeAny” có được cho phép hay không (bước 942). Nếu không (bước 942, KHÔNG), chức năng loại bỏ 318 loại bỏ gói tin (bước 944). Theo cách khác (bước 942, CÓ), quá trình tiến tới bước 932.

Dù tiến tới từ nhánh CÓ của bước 926, nhánh KHÔNG của bước 930, hay nhánh CÓ của bước 942, thì chức năng loại bỏ 318 vẫn cập nhật “InitRecovSeqNum” (bước 932), cập nhật “InitSequenceHistory” (bước 934), và xóa “InitTakeAny” (bước 936). Chức năng loại bỏ 318 xác định xem liệu số trình tự nằm trong STAR hay không (nghĩa là, phạm vi {“GenSeqSpace” – d; ...; “GenSeqSpace” – 2 x d}) (bước 938). Nếu vậy (bước 938, CÓ), chức năng loại bỏ 318 cài đặt “TakeAny” là ĐÚNG (bước 940) và gói tin được chấp nhận (bước 918). Theo cách khác (bước 938, KHÔNG), chức năng loại bỏ 318 chấp nhận gói tin (bước 918).

Một tùy chọn có thể có để mã hóa “SeqResetFlag”, “InitSeqFlag”, và “số trình tự mới thuộc về không gian số trình tự tuyến tính” trong R-TAG được mô tả dưới đây:

- trong trường dành riêng (các byte thứ 2 và thứ 3 của R-TAG), và/hoặc
- trong trường số trình tự (các byte thứ 4 và thứ 5 của R-TAG).

Trên Fig.10, vốn minh họa định dạng R-TAG (Fig.7-4 trong IEEE 802.1CB):

- “SeqResetFlag” và “InitSeqFlag” có thể được mã hóa trong R-TAG sử dụng 2 bit của trường dành riêng, một bit cho mỗi cờ. Theo 802.1CB-2017: “Trường này sẽ được truyền với tất cả các số không và sẽ được bỏ qua khi thu nhận.” “Số trình tự mới thuộc về không gian số trình tự tuyến tính” có thể được mã hóa trong trường số trình tự của R-TAG. “InitSeqFlag” chỉ báo cách để diễn dịch trường số trình tự.

Các phương pháp mã hóa khác cũng có thể được tạo dựng. Ví dụ, “SeqResetFlag” có thể được mã hóa trong R-TAG sử dụng các trường dành riêng sử dụng 1 bit, và “số

trình tự mới thuộc về không gian số trình tự tuyến tính” có thể được mã hóa trong 15 bit còn lại của các trường dành riêng. Việc mã hóa nêu trên cũng bao gồm thông tin của không gian số trình tự, vì vậy không cần phải mã hóa một cách rõ ràng “*InitSeqFlag*”.

Fig.11 là sơ đồ khối giản lược của nút mạng 1100 theo một vài phương án thực hiện của sáng chế. Nút mạng 1100 có thể là nút TX 302 hoặc nút RX 304 đã mô tả trên đây. Như được minh họa, nút mạng 1100 bao gồm một hoặc nhiều bộ xử lý 1104 (chẳng hạn, các đơn vị xử lý trung tâm (CPU - Central Processing Unit), các mạch tích hợp chuyên dụng (ASIC - Application Specific Integrated Circuit), các mảng cổng lập trình được bằng trường (FPGA - Field Programmable Gate Array), và/hoặc tương tự), bộ nhớ 1106, và giao diện mạng 1108. Một hoặc nhiều bộ xử lý 1104 ở đây cũng được xem như hệ mạch xử lý. Một hoặc nhiều bộ xử lý 1104 vận hành để cung cấp một hoặc nhiều chức năng của nút mạng 1100 như được mô tả ở đây (chẳng hạn, một hoặc nhiều chức năng của nút TX 302 hoặc một hoặc nhiều chức năng của nút RX 304, như được mô tả ở đây). Theo một vài phương án thực hiện, chức năng (các chức năng) được thực hiện trong phần mềm mà được lưu trữ, chẳng hạn, trong bộ nhớ 1106 và được thực thi bởi một hoặc nhiều bộ xử lý 1104.

Fig.12 là sơ đồ khối giản lược minh họa phương án thực hiện được ảo hóa của nút mạng 1100 theo một vài phương án thực hiện của sáng chế. Phần thảo luận này có thể áp dụng tương đương với các kiểu khác của các nút mạng. Ngoài ra, các kiểu khác của các nút mạng có thể có các kiến trúc ảo hóa tương tự. Một lần nữa, các dấu hiệu tùy chọn được thể hiện bởi các hộp nét đứt.

Như được sử dụng ở đây, nút mạng “được ảo hóa” là một cách thức thực hiện của nút mạng 1100 trong đó ít nhất một phần tính năng của nút mạng 1100 được thực hiện dưới dạng bộ phận ảo (các bộ phận ảo) (chẳng hạn, qua máy ảo (các máy ảo) thực thi trên nút xử lý vật lý (các nút xử lý vật lý) trong mạng (các mạng)). Như được minh họa, theo ví dụ này, nút mạng 1100 bao gồm một hoặc nhiều nút xử lý 1200 được ghép với hoặc được bao gồm dưới dạng một phần của mạng (các mạng) 1202. Mỗi nút xử lý 1200 bao gồm một hoặc nhiều bộ xử lý 1204 (chẳng hạn, các CPU, các ASIC, các FPGA, và/hoặc tương tự), bộ nhớ 1206, và giao diện mạng 1208. Theo ví dụ này, các chức năng 1210 của nút mạng 1100 đã mô tả ở đây (chẳng hạn, một hoặc nhiều chức năng của nút

TX 302 hoặc một hoặc nhiều chức năng của nút RX 304, như được mô tả ở đây) được thực hiện ở một nút xử lý trong số các nút xử lý 1200 hoặc được phân bố ngang qua hai hoặc nhiều nút xử lý 1200 theo cách mong muốn bất kỳ. Theo một vài phương án thực hiện cụ thể, một vài hoặc tất cả các chức năng 1210 của nút mạng 1100 đã mô tả ở đây được thực hiện dưới dạng các bộ phận ảo được thực thi bởi một hoặc nhiều máy ảo được thực hiện trong môi trường ảo (các môi trường ảo) được chứa bởi nút xử lý (các nút xử lý) 1200.

Theo một vài phương án thực hiện, chương trình máy tính bao gồm các lệnh mà, khi được thực thi bởi ít nhất một bộ xử lý, làm cho ít nhất một bộ xử lý thực hiện tính năng của nút mạng 1100 hoặc nút (chẳng hạn, nút xử lý 1200) thực hiện một hoặc nhiều chức năng 1210 của nút mạng 1100 trong môi trường ảo theo phương án thực hiện bất kỳ trong số các phương án thực hiện đã mô tả ở đây được đề xuất. Theo một vài phương án thực hiện, sóng mang bao gồm sản phẩm chương trình máy tính đã nói trên đây được đề xuất. Sóng mang là một thành phần trong số tín hiệu điện tử, tín hiệu quang học, tín hiệu vô tuyến, hoặc phương tiện lưu trữ máy tính đọc được (chẳng hạn, phương tiện lưu trữ máy tính đọc được không chuyển tiếp như bộ nhớ).

Fig.13 là sơ đồ khối giản lược của nút mạng 1100 theo một vài phương án thực hiện khác của sáng chế. Nút mạng 1100 bao gồm một hoặc nhiều môđun 1300, mỗi một môđun trong số chúng được thực hiện trong phần mềm. Môđun (các môđun) 1300 cung cấp tính năng của nút mạng 1100 304 đã mô tả ở đây (chẳng hạn, một hoặc nhiều chức năng của nút TX 302 hoặc một hoặc nhiều chức năng của nút RX 304, như được mô tả ở đây). Phần mô tả này được áp dụng tương đương với nút xử lý 1200 trên Fig.12 trong đó các môđun 1300 có thể được thực hiện ở một nút xử lý trong số các nút xử lý 1200 hoặc được phân bố ngang qua nhiều nút xử lý 1200.

Các bước, các phương pháp, các dấu hiệu, các chức năng, hoặc các lợi ích thích hợp bất kỳ đã bộc lộ ở đây có thể được thực hiện thông qua một hoặc nhiều đơn vị chức năng hoặc môđun của một hoặc nhiều thiết bị ảo. Mỗi thiết bị ảo có thể bao gồm nhiều đơn vị chức năng nêu trên. Các đơn vị chức năng nêu trên có thể được thực hiện qua hệ mạch xử lý, mà có thể bao gồm một hoặc nhiều bộ vi xử lý hoặc bộ vi điều khiển, cũng như phần cứng kỹ thuật số khác, mà có thể bao gồm bộ xử lý tín hiệu số (DSP - Digital

signal processor), logic số chuyên dụng, và tương tự. Hệ mạch xử lý có thể được tạo cấu hình để thực thi mã chương trình lưu trữ trong bộ nhớ, mà có thể bao gồm một hoặc một vài kiểu bộ nhớ như bộ nhớ chỉ đọc (ROM - Read Only Memory), bộ nhớ truy nhập ngẫu nhiên (RAM - Random Access Memory), bộ nhớ cạc, các thiết bị nhớ tác động nhanh, các thiết bị lưu trữ quang học, v.v. Mã chương trình được lưu trữ trong bộ nhớ bao gồm các lệnh chương trình để thực thi một hoặc nhiều giao thức viễn thông và/hoặc giao thức truyền thông dữ liệu cũng như các lệnh để thực hiện một hoặc nhiều kỹ thuật mô tả ở đây. Theo một vài cách thức thực hiện, hệ mạch xử lý có thể được sử dụng để làm cho đơn vị chức năng tương ứng thực hiện các chức năng tương ứng theo một hoặc nhiều phương án thực hiện của sáng chế.

Mặc dù các quá trình trên các hình vẽ có thể thể hiện thứ tự cụ thể của các vận hành thực hiện bởi các phương án nhất định của sáng chế này, nhưng cần phải hiểu rằng thứ tự như vậy chỉ là ví dụ (nghĩa là, các phương án thay thế có thể thực hiện các vận hành đó theo thứ tự khác, kết hợp các vận hành nhất định, phối hợp các vận hành nhất định, v.v.).

Một vài phương án thực hiện ví dụ của sáng chế là như sau.

Phương án thực hiện 1: Phương pháp được thực hiện bởi nút truyền (302) để mạng tái tạo gói tin hoặc khung, phương pháp này bao gồm một hoặc nhiều bước sau đây:

- xác định (700) rằng chức năng sinh ra trình tự (310) ở nút truyền (302) đã được cài đặt lại;
- đáp ứng với bước xác định (700) rằng chức năng sinh ra trình tự (310) ở nút truyền (302) đã được cài đặt lại, truyền (702) các gói tin thứ nhất trong dòng các gói tin, trong đó: (a) mỗi một gói tin trong số các gói tin thứ nhất bao gồm số trình tự tương ứng từ không gian số trình tự tuyến tính và (b) ít nhất một gói tin thứ nhất từ trong số các gói tin thứ nhất mà được gửi sau khi cài đặt lại còn bao gồm bộ chỉ báo rõ ràng của sự cài đặt lại;
- xác định (704) rằng đầu cuối của không gian số trình tự tuyến tính đã được đạt tới hoặc theo cách khác rằng việc sử dụng không gian số trình tự tuyến tính đã bị vô hiệu hóa; và

- đáp ứng với bước xác định (704) rằng đầu cuối của không gian số trình tự tuyến tính đã được đạt tới hoặc theo cách khác rằng việc sử dụng không gian số trình tự tuyến tính đã bị vô hiệu hóa, truyền (706) các gói tin thứ hai trong dòng các gói tin, trong đó (a) mỗi một gói tin trong số các gói tin thứ hai bao gồm số trình tự tương ứng của không gian số trình tự tuần hoàn.

Phương án thực hiện 2: Phương pháp theo phương án thực hiện 1 trong đó mỗi một gói tin trong số các gói tin thứ nhất còn bao gồm sự chỉ báo rõ ràng rằng không gian số trình tự tuyến tính đang được sử dụng.

Phương án thực hiện 3: Phương pháp theo phương án thực hiện 1 hoặc 2 trong đó mạng nêu trên là mạng nối mạng nhảy với thời gian (TSN).

Phương án thực hiện 4: Phương pháp theo phương án thực hiện bất kỳ trong số các phương án thực hiện từ 1 tới 3 còn bao gồm bước cài đặt lại chức năng sinh ra trình tự (310), trong đó bước cài đặt lại chức năng sinh ra trình tự (310) bao gồm cài đặt lại lịch sử số trình tự, cửa sổ lịch sử, hoặc cả lịch sử số trình tự và cửa sổ lịch sử.

Phương án thực hiện 5: Phương pháp theo phương án thực hiện 3 trong đó các bước gồm xác định (700) rằng chức năng sinh ra trình tự (310) ở nút truyền (302) đã được cài đặt lại, truyền (702) các gói tin thứ nhất trong dòng các gói tin, xác định (704) rằng đầu cuối của không gian số trình tự tuyến tính đã được đạt tới hoặc theo cách khác rằng việc sử dụng không gian số trình tự tuyến tính đã bị vô hiệu hóa, và truyền (706) các gói tin thứ hai trong dòng các gói tin được thực hiện bởi chức năng tái tạo và loại bỏ khung cho độ tin cậy (FRER) (308) của nút truyền (302).

Phương án thực hiện 6: Phương pháp theo phương án thực hiện 1 hoặc 2 trong đó mạng nêu trên là mạng nối mạng tất định (DetNet).

Phương án thực hiện 7: Phương pháp theo phương án thực hiện 6 trong đó các bước gồm xác định (700) rằng chức năng sinh ra trình tự (310) ở nút truyền (302) đã được cài đặt lại, truyền (702) các gói tin thứ nhất trong dòng các gói tin, xác định (704) rằng đầu cuối của không gian số trình tự tuyến tính đã được đạt tới hoặc theo cách khác rằng việc sử dụng không gian số trình tự tuyến tính đã bị vô hiệu hóa, và truyền (706) các gói tin thứ hai trong dòng các gói tin được thực hiện bởi chức năng tái tạo và loại bỏ gói tin (PRER) của nút truyền (302).

Phương án thực hiện 8: Phương pháp được thực hiện bởi nút thu nhận (304) để mạng loại bỏ gói tin hoặc khung, phương pháp này bao gồm một hoặc nhiều bước sau đây:

- thu nhận (900) gói tin từ một dòng trong số các dòng đã tái tạo mà đi qua các đường dẫn khác nhau từ nút truyền (302) tới nút thu nhận (304) qua mạng, gói tin này bao gồm số trình tự;
- xác định (924, CÓ) rằng gói tin bao gồm sự chỉ báo rõ ràng rằng không gian số trình tự tuyến tính đang được sử dụng;
- xác định (926) xem liệu gói tin có bao gồm sự chỉ báo rõ ràng rằng chức năng sinh ra trình tự (310) ở nút truyền (302) đã được cài đặt lại hay không;
- xác định (926) xem liệu số trình tự của gói tin có nằm bên ngoài phạm vi bỏ qua cài đặt lại (iRIR) được kết hợp với việc sử dụng không gian số trình tự tuyến tính hay không;
- đáp ứng với bước xác định rằng gói tin bao gồm sự chỉ báo rõ ràng rằng không gian số trình tự tuyến tính đang được sử dụng (924, CÓ) và rằng gói tin bao gồm sự chỉ báo rõ ràng rằng chức năng sinh ra trình tự (310) ở nút truyền (302) đã được cài đặt lại và số trình tự của gói tin nằm bên ngoài iRIR (926, CÓ), chấp nhận (918) gói tin.

Phương án thực hiện 9: Phương pháp theo phương án thực hiện 8 còn bao gồm, đáp ứng với bước xác định rằng gói tin bao gồm sự chỉ báo rõ ràng rằng không gian số trình tự tuyến tính đang được sử dụng (924, CÓ) và rằng gói tin bao gồm sự chỉ báo rõ ràng rằng chức năng sinh ra trình tự (310) ở nút truyền (302) đã được cài đặt lại và số trình tự của gói tin nằm bên ngoài iRIR (926, CÓ), cập nhật (932; 934) iHSW và lịch sử được kết hợp với không gian số trình tự tuyến tính dựa trên số trình tự của gói tin.

Phương án thực hiện 10: Phương pháp theo phương án thực hiện 8 còn bao gồm, khi xác định rằng gói tin bao gồm sự chỉ báo rõ ràng rằng không gian số trình tự tuyến tính đang được sử dụng (924, CÓ) và rằng gói tin hoặc không bao gồm sự chỉ báo rõ ràng rằng chức năng sinh ra trình tự (310) ở nút truyền (302) đã được cài đặt lại hoặc số trình tự của gói tin không nằm bên ngoài iRIR (926, KHÔNG):

- xác định (928) xem liệu số trình tự của gói tin có nằm trong cửa sổ lịch sử (iHSW) được kết hợp với việc sử dụng không gian số trình tự tuyến tính hay không;
- xác định (930) xem liệu số trình tự của gói tin đã nằm trong lịch sử được kết hợp với việc sử dụng không gian số trình tự tuyến tính mà bao gồm các số trình tự vốn đã được đã thu nhận hay chưa; và
- đáp ứng với bước xác định (928, CÓ) rằng số trình tự của gói tin nằm trong iHSW và xác định (930, KHÔNG) rằng số trình tự của gói tin chưa nằm trong lịch sử, chấp nhận (918) gói tin.

Phương án thực hiện 11: Phương pháp theo phương án thực hiện 10 còn bao gồm, đáp ứng với bước xác định (928, CÓ) rằng số trình tự của gói tin nằm trong iHSW và xác định (930, CÓ) rằng số trình tự của gói tin đã nằm trong lịch sử, loại bỏ (918) gói tin.

Phương án thực hiện 12: Phương pháp theo phương án thực hiện 10 còn bao gồm, đáp ứng với bước xác định (928, CÓ) rằng số trình tự của gói tin nằm trong iHSW và xác định (930, KHÔNG) rằng số trình tự của gói tin chưa nằm trong lịch sử, cập nhật (932; 934) iHSW và lịch sử được kết hợp với không gian số trình tự tuyến tính dựa trên số trình tự của gói tin.

Phương án thực hiện 13: Phương pháp được thực hiện bởi nút truyền (302) để mạng tái tạo gói tin hoặc khung, phương pháp này bao gồm một hoặc nhiều bước sau đây:

- thu được (600) gói tin cần được gửi;
- xác định (602, KHÔNG) rằng gói tin chưa được tái tạo bởi nút mạng trước trong mạng (chẳng hạn, xác định rằng R-Tag chưa xuất hiện trong gói tin);
- xác định (608) xem liệu chức năng sinh ra trình tự (310) ở nút truyền (302) đã được cài đặt lại hay chưa;
- cho phép (610) sự chỉ báo rõ ràng trong gói tin mà chỉ báo rằng chức năng sinh ra trình tự (310) đã được cài đặt lại đáp ứng với bước xác định (608) rằng chức năng sinh ra trình tự (310) ở nút truyền (302) đã được cài đặt lại; theo cách khác vô hiệu hóa (612) sự chỉ báo rõ ràng trong gói tin;
- xác định (616, CÓ) rằng không gian số trình tự tuyến tính sẽ được sử dụng;

- đáp ứng với bước xác định (616, CỎ) rằng không gian số trình tự tuyến tính sẽ được sử dụng:
 - cho phép (618) sự chỉ báo rõ ràng trong gói tin mà chỉ báo rằng không gian số trình tự tuyến tính đang được sử dụng; và
 - bổ sung (620) số trình tự từ không gian số trình tự tuyến tính vào gói tin;
- bổ sung (628) thẻ vào gói tin mà chỉ báo rằng nó đã được được tái tạo;
- cung cấp (604) gói tin để tái tạo và truyền.

Phương án thực hiện 14: Nút mạng (302; 304; 1100) được làm thích ứng để thực hiện phương pháp theo phương án thực hiện bất kỳ trong số các phương án thực hiện từ 1 tới 13.

Người có hiểu biết trung bình trong lĩnh vực kỹ thuật này sẽ nhận ra các cải tiến và các biến thể với các phương án thực hiện của sáng chế. Tất cả các cải tiến và các biến thể nêu trên đều nằm trong phạm vi bảo hộ của sáng chế.

YÊU CẦU BẢO HỘ

1. Phương pháp được thực hiện bởi nút truyền (302) để tái tạo gói tin hoặc khung, phương pháp này bao gồm các bước:

xác định (700) rằng chức năng sinh ra trình tự (310) ở nút truyền (302) đã được cài đặt lại;

đáp ứng với bước xác định (700) rằng chức năng sinh ra trình tự (310) ở nút truyền (302) đã được cài đặt lại, truyền (702) các gói tin thứ nhất trong dòng các gói tin qua ít nhất hai đường dẫn tách rời thông qua mạng, dòng các gói tin này là một dòng đã tái tạo trong số các dòng đã tái tạo, trong đó: (a) mỗi gói tin trong số các gói tin thứ nhất bao gồm số trình tự tương ứng từ không gian số trình tự tuyến tính và sự chỉ báo rõ ràng rằng không gian số trình tự tuyến tính đang được sử dụng và (b) ít nhất gói tin thứ nhất từ trong số các gói tin thứ nhất mà được gửi sau khi cài đặt lại còn bao gồm bộ chỉ báo rõ ràng của sự cài đặt lại;

xác định (704) rằng đầu cuối của không gian số trình tự tuyến tính đã được đạt tới hoặc theo cách khác rằng việc sử dụng không gian số trình tự tuyến tính đã bị vô hiệu hóa; và

đáp ứng với bước xác định (704) rằng đầu cuối của không gian số trình tự tuyến tính đã được đạt tới hoặc theo cách khác rằng việc sử dụng không gian số trình tự tuyến tính đã bị vô hiệu hóa, truyền (706) các gói tin thứ hai trong dòng các gói tin qua ít nhất hai đường dẫn tách rời thông qua mạng, trong đó (a) mỗi gói tin trong số các gói tin thứ hai bao gồm số trình tự tương ứng của không gian số trình tự tuần hoàn.

2. Phương pháp theo điểm 1, trong đó mạng là mạng nối mạng nhạy với thời gian (TSN - Time-Sensitive Networking).

3. Phương pháp theo điểm 1 hoặc 2, còn bao gồm bước cài đặt lại chức năng sinh ra trình tự (310), trong đó bước cài đặt lại chức năng sinh ra trình tự (310) bao gồm cài đặt lại lịch sử số trình tự, cửa sổ lịch sử, hoặc cả lịch sử số trình tự và cửa sổ lịch sử.

4. Phương pháp theo điểm 2 hoặc 3, trong đó các bước gồm xác định (700) rằng chức năng sinh ra trình tự (310) ở nút truyền (302) đã được cài đặt lại, truyền (702) các gói tin thứ nhất trong dòng các gói tin, xác định (704) rằng đầu cuối của không gian số trình

tự tuyến tính đã được đạt tới hoặc theo cách khác rằng việc sử dụng không gian số trình tự tuyến tính đã bị vô hiệu hóa, và truyền (706) các gói tin thứ hai trong dòng các gói tin được thực hiện bởi chức năng tái tạo và loại bỏ khung cho độ tin cậy (FRER - Frame Replication and Elimination for Reliability) (308) của nút truyền (302).

5. Phương pháp theo điểm 1, trong đó mạng là mạng nối mạng tất định (DetNet - Deterministic Networking).

6. Phương pháp theo điểm 5, trong đó các bước gồm xác định (700) rằng chức năng sinh ra trình tự (310) ở nút truyền (302) đã được cài đặt lại, truyền (702) các gói tin thứ nhất trong dòng các gói tin, xác định (704) rằng đầu cuối của không gian số trình tự tuyến tính đã được đạt tới hoặc theo cách khác rằng việc sử dụng không gian số trình tự tuyến tính đã bị vô hiệu hóa, và truyền (706) các gói tin thứ hai trong dòng các gói tin được thực hiện bởi chức năng tái tạo và loại bỏ gói tin (PREF - Gói tin Replication and Elimination Function) của nút truyền (302).

7. Nút truyền (302; 1100) để tái tạo gói tin hoặc khung, nút truyền (302; 1100) này bao gồm:

giao diện mạng (1108; 1208); và

hệ mạch xử lý (1104; 1204) được kết hợp với giao diện mạng (1108; 1208), hệ mạch xử lý (1104; 1204) này được tạo cấu hình để làm cho nút truyền (302; 1100):

xác định (700) rằng chức năng sinh ra trình tự (310) ở nút truyền (302; 1100) đã được cài đặt lại;

đáp ứng với bước xác định (700) rằng chức năng sinh ra trình tự (310) ở nút truyền (302; 1100) đã được cài đặt lại, truyền (702) các gói tin thứ nhất trong dòng các gói tin qua ít nhất hai đường dẫn tách rời thông qua mạng, dòng các gói tin là một dòng đã tái tạo trong số các dòng đã tái tạo, trong đó: (a) mỗi gói tin trong số các gói tin thứ nhất bao gồm số trình tự tương ứng từ không gian số trình tự tuyến tính và sự chỉ báo rõ ràng rằng không gian số trình tự tuyến tính đang được sử dụng và (b) ít nhất gói tin thứ nhất từ trong số các gói tin thứ nhất mà được gửi sau khi cài đặt lại còn bao gồm bộ chỉ báo rõ ràng của sự cài đặt lại;

xác định (704) rằng đầu cuối của không gian số trình tự tuyến tính đã được đạt tới hoặc theo cách khác rằng việc sử dụng không gian số trình tự tuyến tính đã bị vô hiệu hóa; và

đáp ứng với xác định (704) rằng đầu cuối của không gian số trình tự tuyến tính đã được đạt tới hoặc theo cách khác rằng việc sử dụng không gian số trình tự tuyến tính đã bị vô hiệu hóa, truyền (706) các gói tin thứ hai trong dòng các gói tin qua ít nhất hai đường dẫn tách rời thông qua mạng, trong đó (a) mỗi gói tin trong số các gói tin thứ hai bao gồm số trình tự tương ứng của không gian số trình tự tuần hoàn.

8. Nút truyền (302; 1100) theo điểm 7, trong đó hệ mạch xử lý (1104; 1204) còn được tạo cấu hình để làm cho nút truyền (302; 1100) thực hiện phương pháp theo điểm bất kỳ trong số các điểm từ 2 tới 6.

9. Phương pháp được thực hiện bởi nút truyền (302) để tái tạo gói tin hoặc khung, phương pháp này bao gồm các bước:

thu được (600) gói tin cần được gửi;

xác định (602, KHÔNG) rằng gói tin này không bao gồm thẻ mà chỉ báo rằng gói tin này đã được tái tạo;

đáp ứng với bước xác định (602, KHÔNG) rằng gói tin này không bao gồm thẻ mà chỉ báo rằng gói tin này đã được tái tạo:

xác định (616, CÓ) rằng không gian số trình tự tuyến tính sẽ được sử dụng;

đáp ứng với bước xác định (616, CÓ) rằng không gian số trình tự tuyến tính sẽ được sử dụng:

cho phép (618) sự chỉ báo rõ ràng trong gói tin mà chỉ báo rằng không gian số trình tự tuyến tính đang được sử dụng; và

bổ sung (620) số trình tự từ không gian số trình tự tuyến tính vào gói tin; và

bổ sung (628) thẻ vào gói tin mà chỉ báo rằng gói tin đã được tái tạo; và

cung cấp (604) gói tin để tái tạo và truyền.

10. Phương pháp theo điểm 9, còn bao gồm các bước, đáp ứng với bước xác định (602, KHÔNG) rằng gói tin không bao gồm thẻ mà chỉ báo rằng gói tin đã được tái tạo:

xác định (608) rằng chức năng sinh ra trình tự (310) ở nút truyền (302) đã được cài đặt lại; và

cho phép (610) sự chỉ báo rõ ràng trong gói tin mà chỉ báo rằng chức năng sinh ra trình tự (310) đã được cài đặt lại đáp ứng với bước xác định (608) rằng chức năng sinh ra trình tự (310) ở nút truyền (302) đã được cài đặt lại, theo cách khác vô hiệu hóa (612) sự chỉ báo rõ ràng trong gói tin mà chỉ báo rằng chức năng sinh ra trình tự (310) đã được cài đặt lại.

11. Phương pháp theo điểm 9 hoặc 10, còn bao gồm, đáp ứng với bước xác định (616, CÓ) rằng không gian số trình tự tuyến tính sẽ được sử dụng, bước tăng (622) số trình tự từ không gian số trình tự tuyến tính.

12. Phương pháp theo điểm 11, còn bao gồm bước lặp lại phương pháp theo điểm 13 cho các gói tin bổ sung cho đến khi không gian số trình tự tuyến tính được dùng hết hoặc theo cách khác việc sử dụng không gian số trình tự tuyến tính bị vô hiệu hóa.

13. Phương pháp theo điểm 12, còn bao gồm các bước, sau khi không gian số trình tự tuyến tính được dùng hết hoặc theo cách khác sau khi việc sử dụng không gian số trình tự tuyến tính bị vô hiệu hóa:

thu được (600) gói tin khác nữa cần được gửi;

xác định (602, KHÔNG) rằng gói tin khác nữa này không bao gồm thẻ mà chỉ báo rằng gói tin khác nữa này đã được tái tạo;

đáp ứng với bước xác định (602, KHÔNG) rằng gói tin khác nữa không bao gồm thẻ mà chỉ báo rằng gói tin khác nữa đã được tái tạo:

xác định (616, KHÔNG) rằng không gian số trình tự tuần hoàn, thay vì không gian số trình tự tuyến tính, sẽ được sử dụng;

đáp ứng với bước xác định (616, KHÔNG) rằng không gian số trình tự tuần hoàn sẽ được sử dụng:

vô hiệu hóa (630) sự chỉ báo rõ ràng trong gói tin khác nữa mà chỉ báo rằng không gian số trình tự tuyến tính đang được sử dụng; và

bổ sung (632) số trình tự từ không gian số trình tự tuần hoàn vào gói tin khác nữa; và

bổ sung (628) thẻ vào gói tin khác nữa mà chỉ báo rằng gói tin khác nữa đã được tái tạo; và

cung cấp (604) gói tin khác nữa để tái tạo và truyền.

14. Phương pháp theo điểm 13, còn bao gồm các bước, đáp ứng với bước xác định (602, KHÔNG) rằng gói tin khác nữa không bao gồm thẻ mà chỉ báo rằng gói tin khác nữa đã được tái tạo:

xác định (608) xem liệu chức năng sinh ra trình tự (310) ở nút truyền (302) đã được cài đặt lại hay chưa; và

cho phép (610) sự chỉ báo rõ ràng trong gói tin khác nữa mà chỉ báo rằng chức năng sinh ra trình tự (310) đã được cài đặt lại đáp ứng với bước xác định (608) rằng chức năng sinh ra trình tự (310) ở nút truyền (302) đã được cài đặt lại, theo cách khác vô hiệu hóa (612) sự chỉ báo rõ ràng trong gói tin khác nữa mà chỉ báo rằng chức năng sinh ra trình tự (310) đã được cài đặt lại.

15. Nút truyền (302; 1100) để tái tạo gói tin hoặc khung, nút truyền (302; 1100) này bao gồm:

giao diện mạng (1108; 1208); và

hệ mạch xử lý (1104; 1204) được kết hợp với giao diện mạng (1108; 1208), hệ mạch xử lý (1104; 1204) được tạo cấu hình để làm cho nút truyền (302; 1100):

thu được (600) gói tin cần được gửi;

xác định (602, KHÔNG) rằng gói tin này không bao gồm thẻ mà chỉ báo rằng gói tin đã được tái tạo;

đáp ứng với xác định (602, KHÔNG) rằng gói tin này không bao gồm thẻ mà chỉ báo rằng gói tin đã được tái tạo:

xác định (616, CÓ) rằng không gian số trình tự tuyến tính sẽ được sử dụng;

đáp ứng với bước xác định (616, CÓ) rằng không gian số trình tự tuyến tính sẽ được sử dụng:

cho phép (618) sự chỉ báo rõ ràng trong gói tin mà chỉ báo rằng không gian số trình tự tuyến tính đang được sử dụng; và

bổ sung (620) số trình tự từ không gian số trình tự tuyến tính vào gói tin; và

bổ sung (628) thẻ vào gói tin mà chỉ báo rằng gói tin đã được tái tạo; và cung cấp (604) gói tin để tái tạo và truyền.

16. Nút truyền (302; 1100) theo điểm 15, trong đó hệ mạch xử lý (1104; 1204) còn được tạo cấu hình để làm cho nút truyền (302; 1100) thực hiện phương pháp theo điểm bất kỳ trong số các điểm từ 10 tới 14.

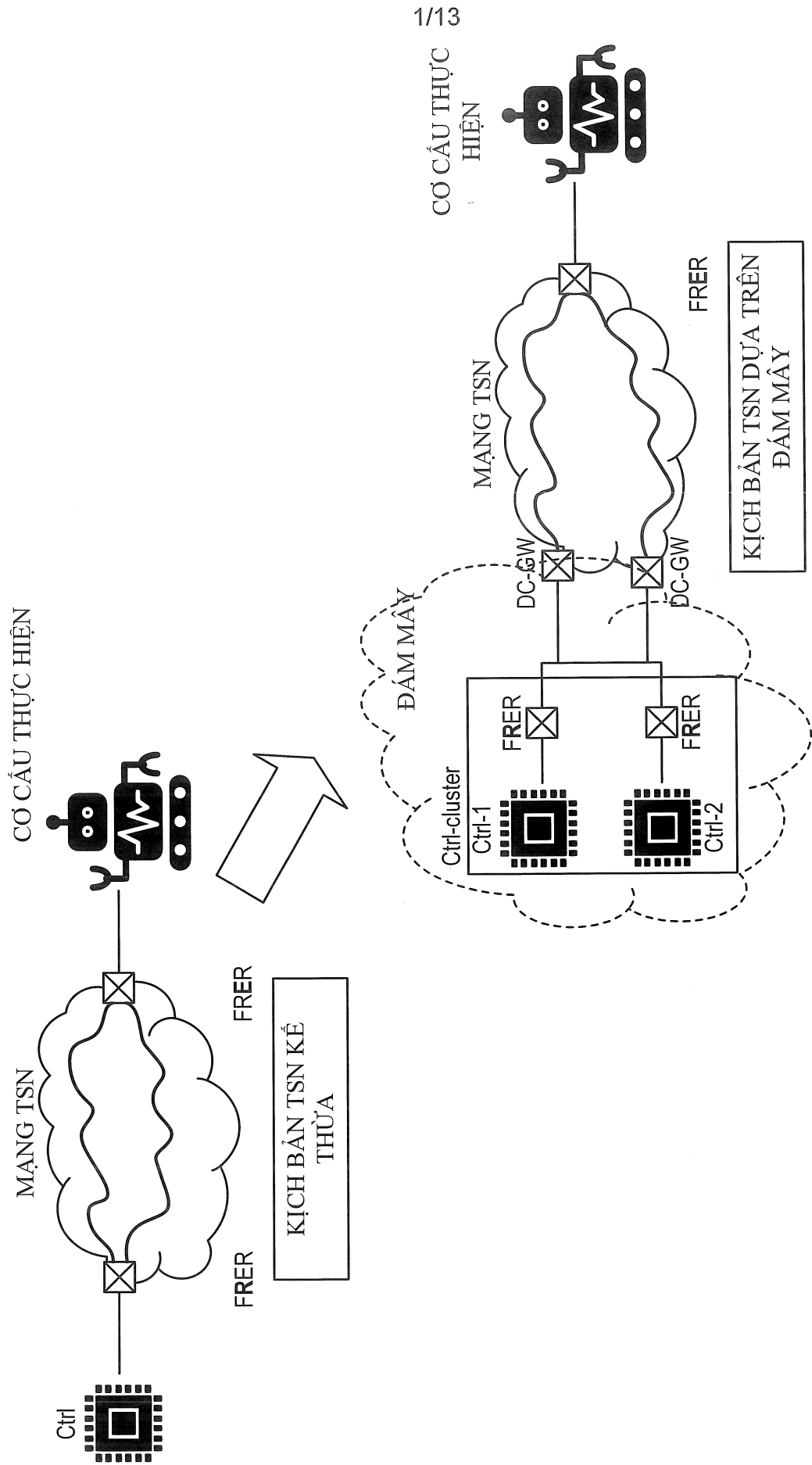
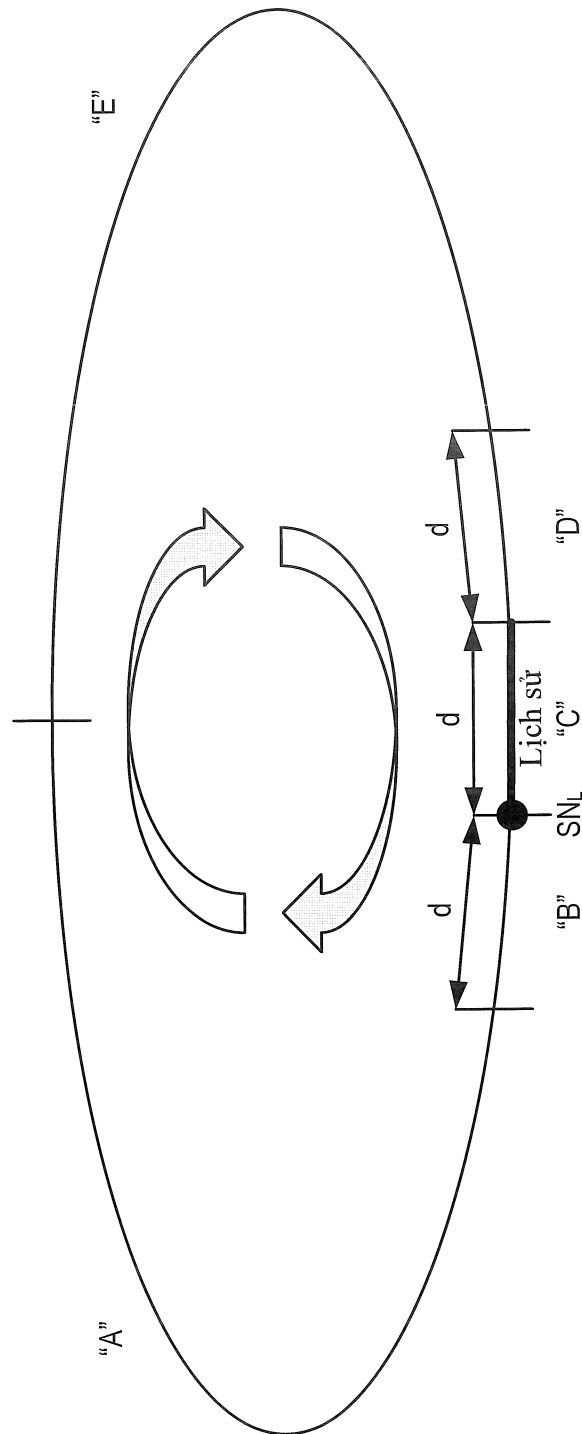


FIG.1

2/13

**FIG.2**

3/13

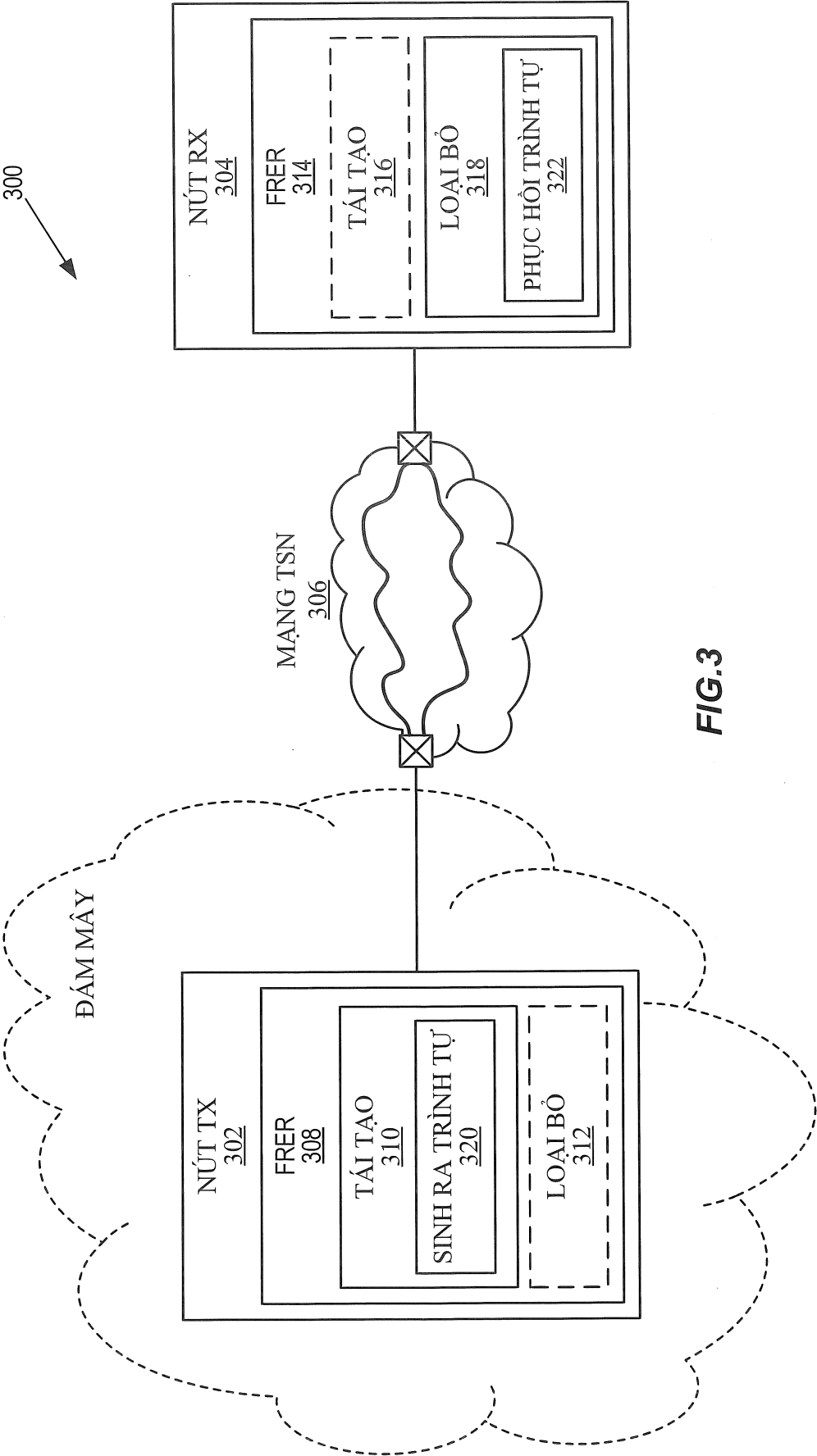
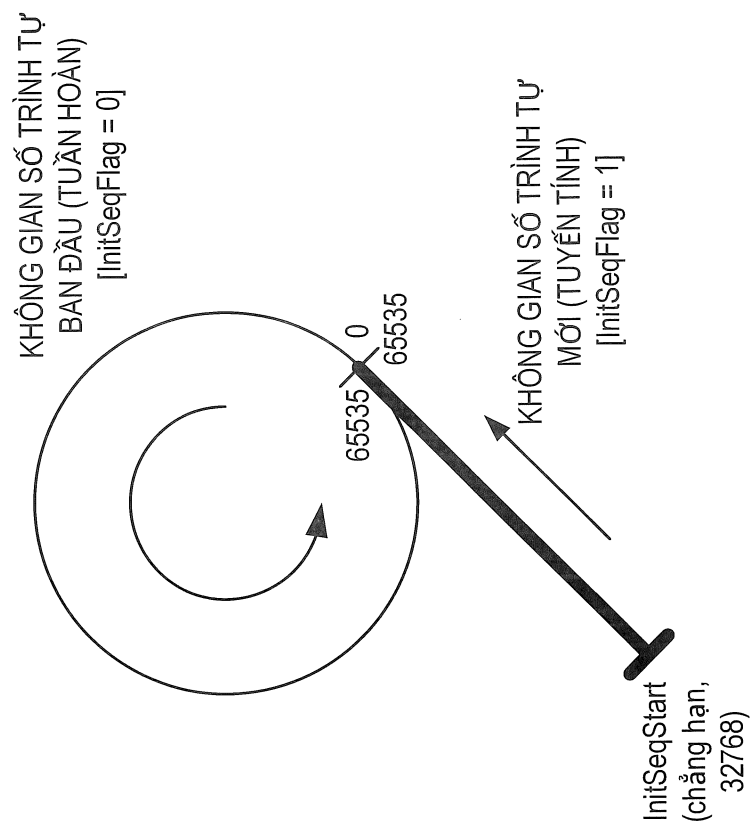


FIG.3

4/13

**FIG.4**

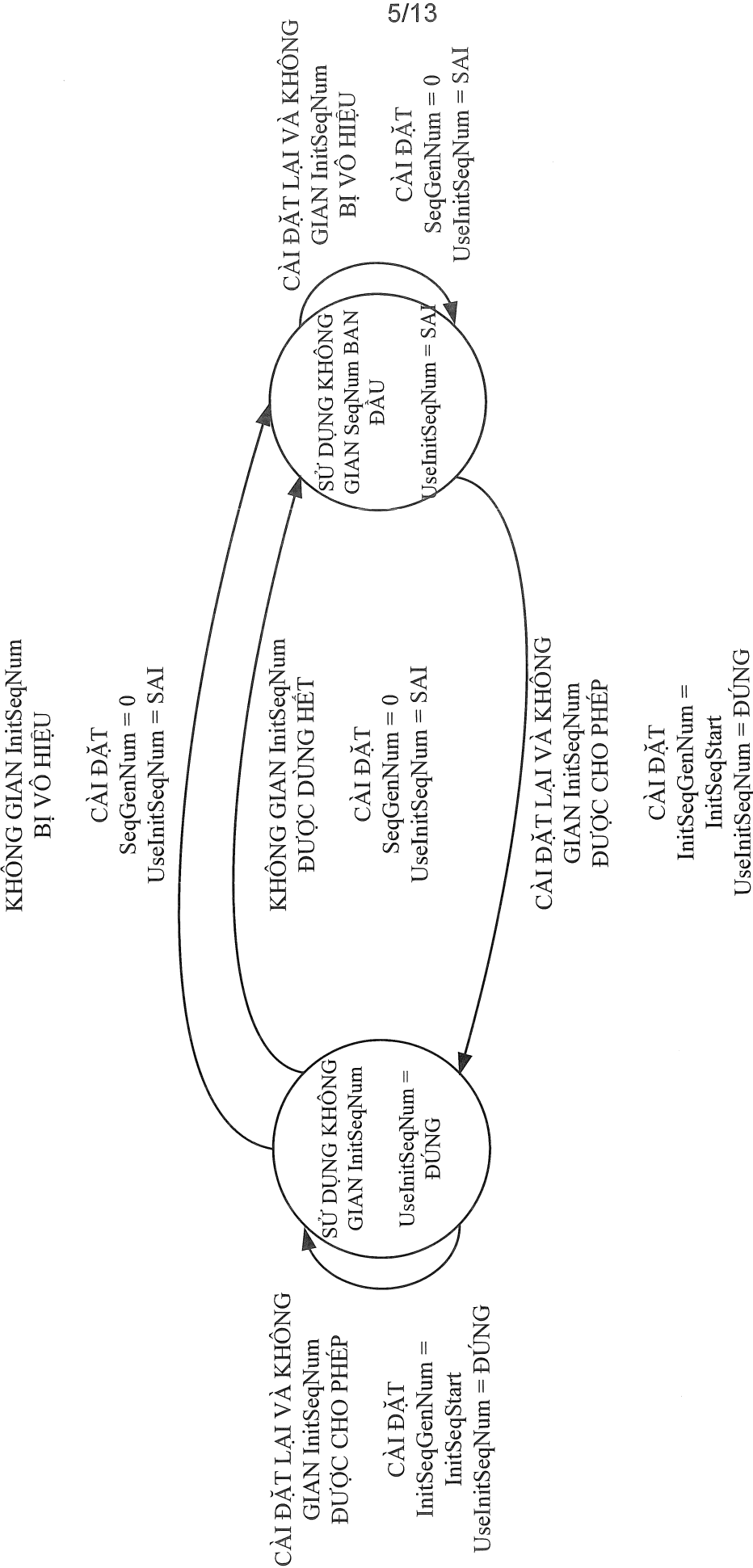
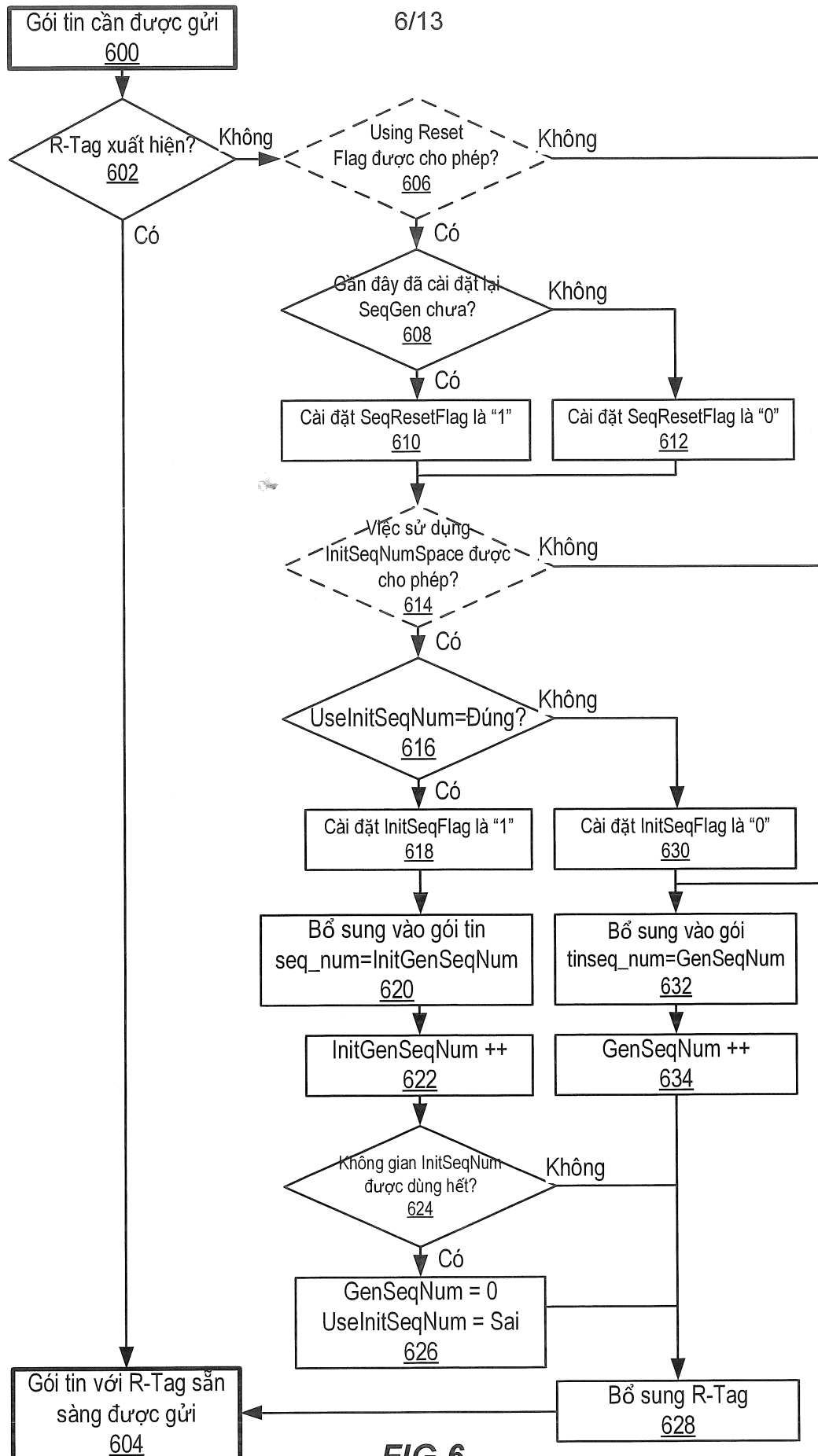
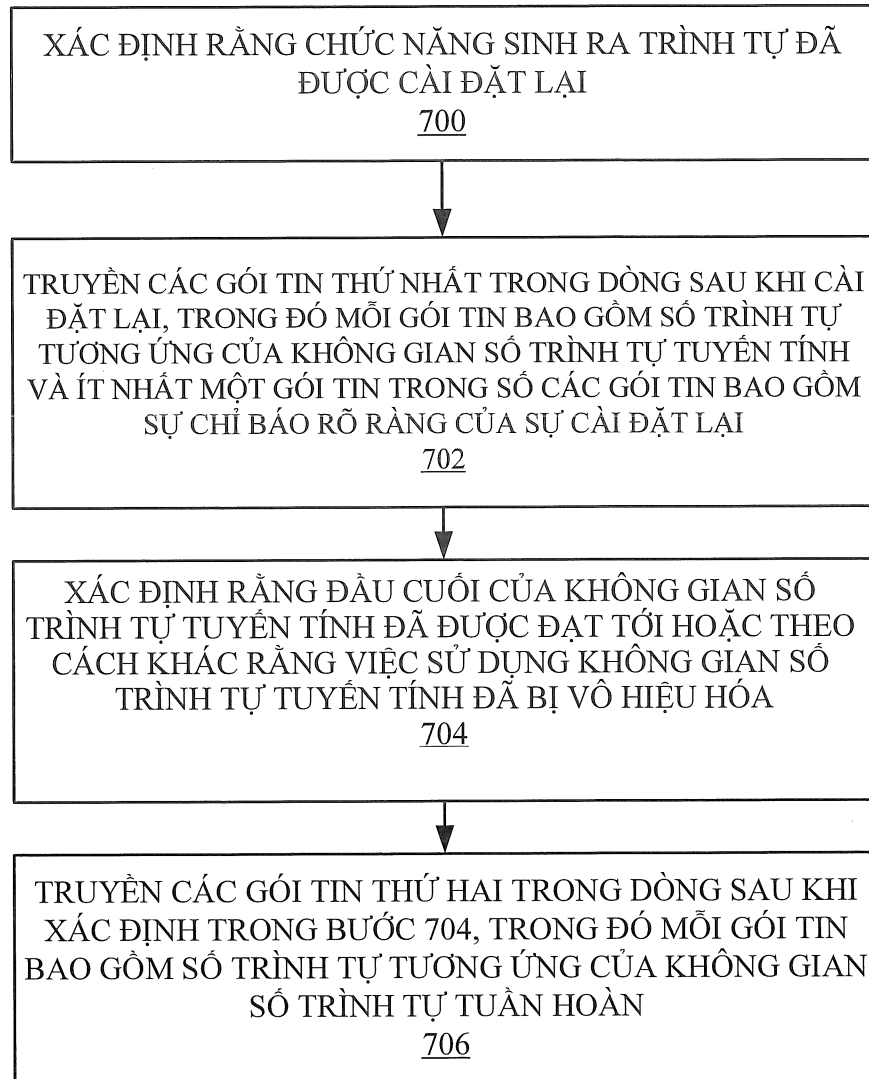


FIG.5



7/13

**FIG.7**

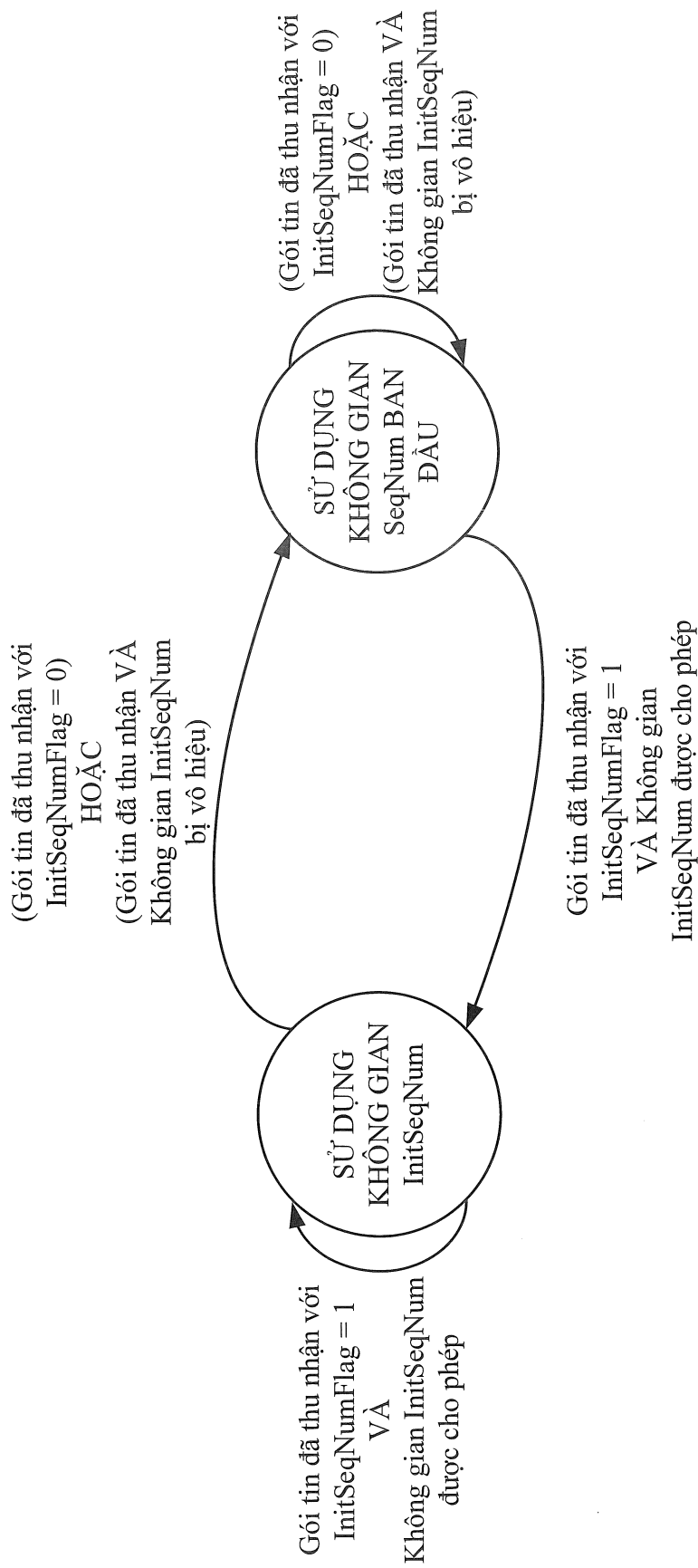
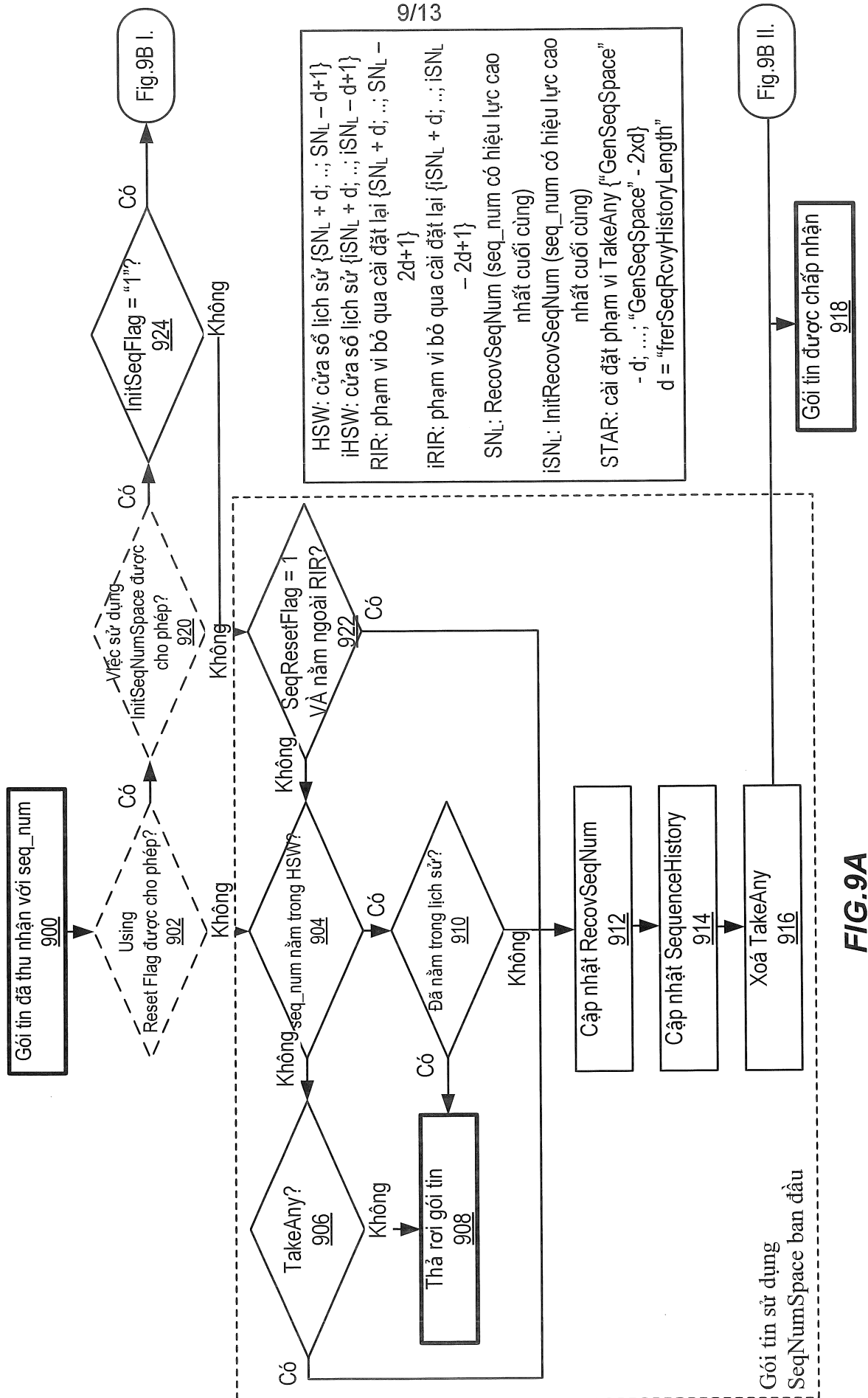


FIG.8



10/13

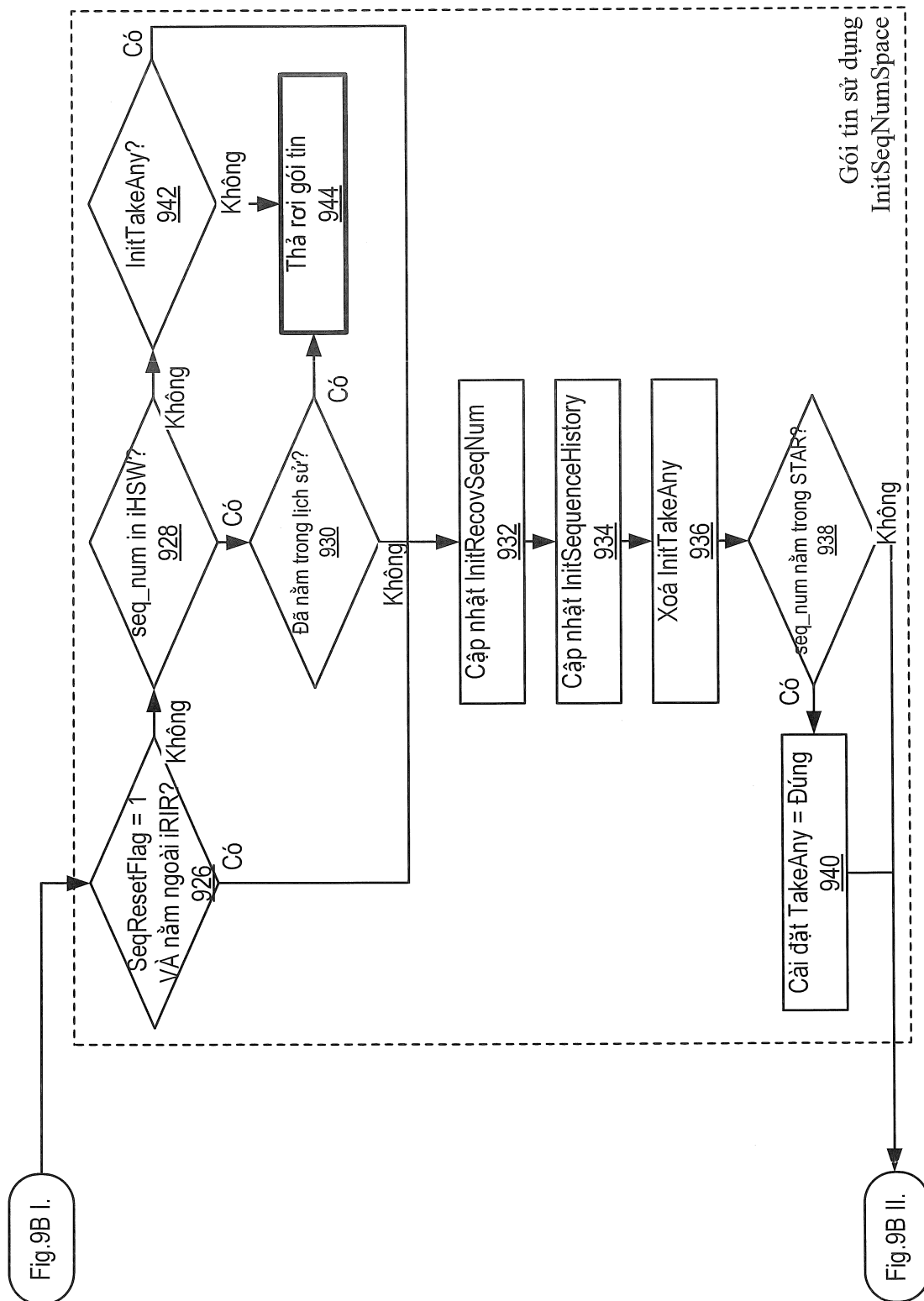


FIG.9B

Nhóm tám bit:

0	1	2	3	4	5
Kiểu Ether (xem Bảng 7-1)			Được dành riêng (7.8.2)		Số trình tự (7.8.2)

FIG.10

12/13

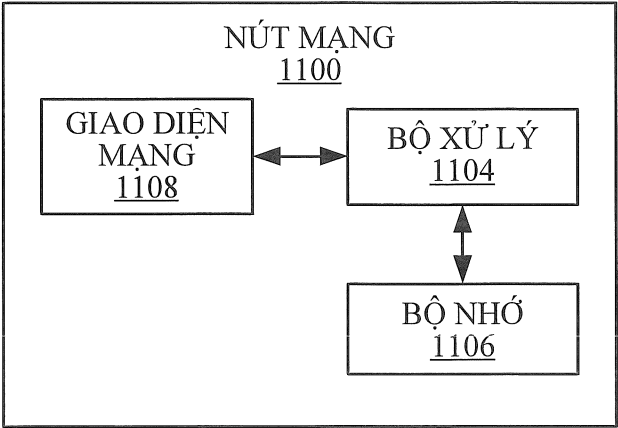


FIG.11

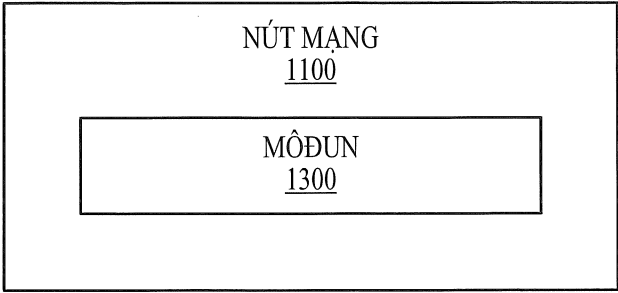


FIG.13

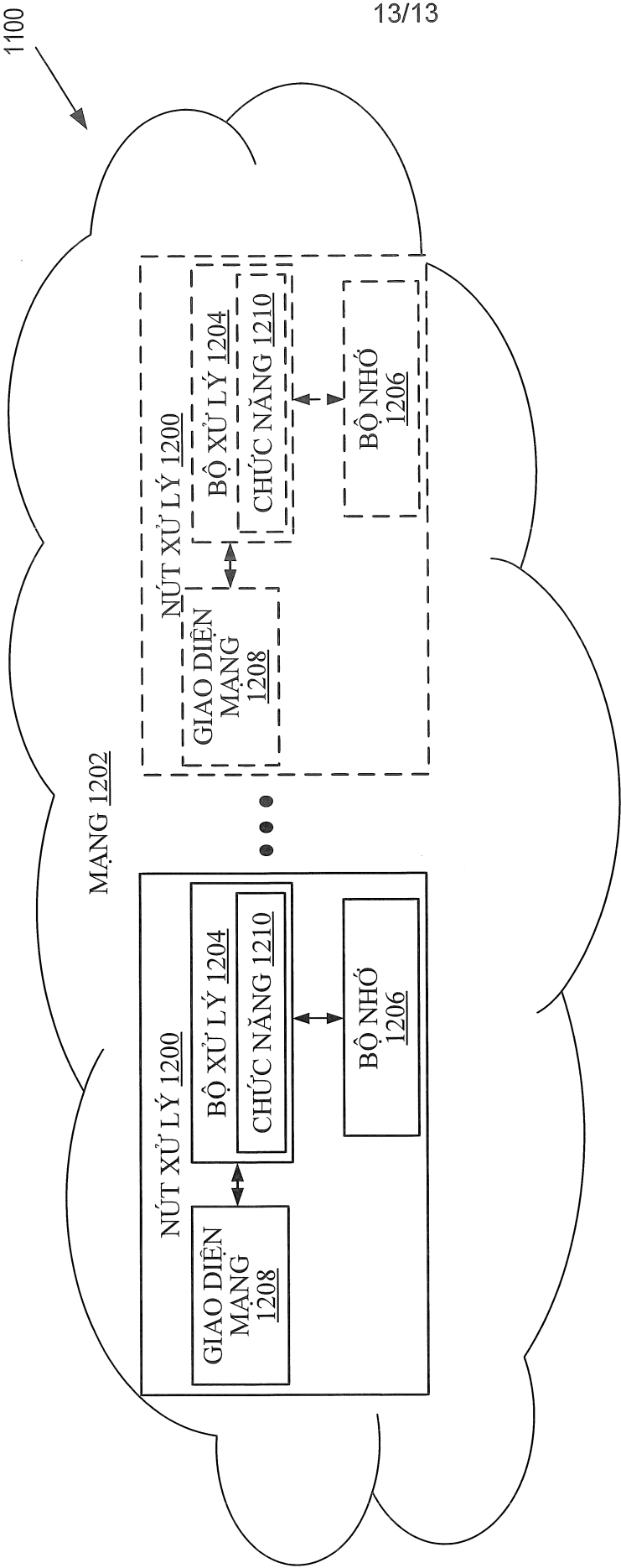


FIG.12