



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

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

(11)



1-0053776

(51)^{2021.01} G06F 9/50; G06N 20/00; G06F 13/28

(13) B

(21) 1-2022-06132

(22) 29/03/2021

(86) PCT/US2021/070327 29/03/2021

(87) WO 2021/203125 A1 07/10/2021

(30) 16/834,986 30/03/2020 US

(45) 25/11/2025 452

(43) 26/12/2022 417A

(73) Qualcomm Incorporated (US)

Attn: International IP Administration, 5775 Morehouse Drive, San Diego, California
92121-1714 (US)

(72) PARK, Hee Jun (US); HOFMANN, Richard Gerard (US).

(74) Công ty TNHH Quốc tế D & N (D&N INTERNATIONAL CO.,LTD.)

(54) PHƯƠNG PHÁP VÀ HỆ THỐNG XỬ LÝ ĐỂ THỰC HIỆN XỬ LÝ DỮ LIỆU
SONG SONG

(21) 1-2022-06132

(57) Sáng chế đề cập đến phương pháp thực hiện xử lý dữ liệu song song, phương pháp này bao gồm các bước: nhận dữ liệu để xử lý song song từ bộ yêu cầu xử lý dữ liệu; tạo nhiều khối dữ liệu con; xác định nhiều phần dữ liệu trong mỗi khối dữ liệu con trong số nhiều khối dữ liệu con; thay đổi thứ tự của nhiều phần dữ liệu trong ít nhất một khối dữ liệu con trong số nhiều khối dữ liệu con; cung cấp nhiều khối dữ liệu con, gồm ít nhất một khối dữ liệu con bao gồm thứ tự thay đổi của nhiều phần dữ liệu, cho nhiều khối xử lý để xử lý song song; và nhận dữ liệu được xử lý được liên kết với nhiều khối dữ liệu con từ nhiều khối xử lý. Ngoài ra, các khía cạnh của sáng chế cũng đề cập đến hệ thống xử lý để thực hiện xử lý dữ liệu song song.

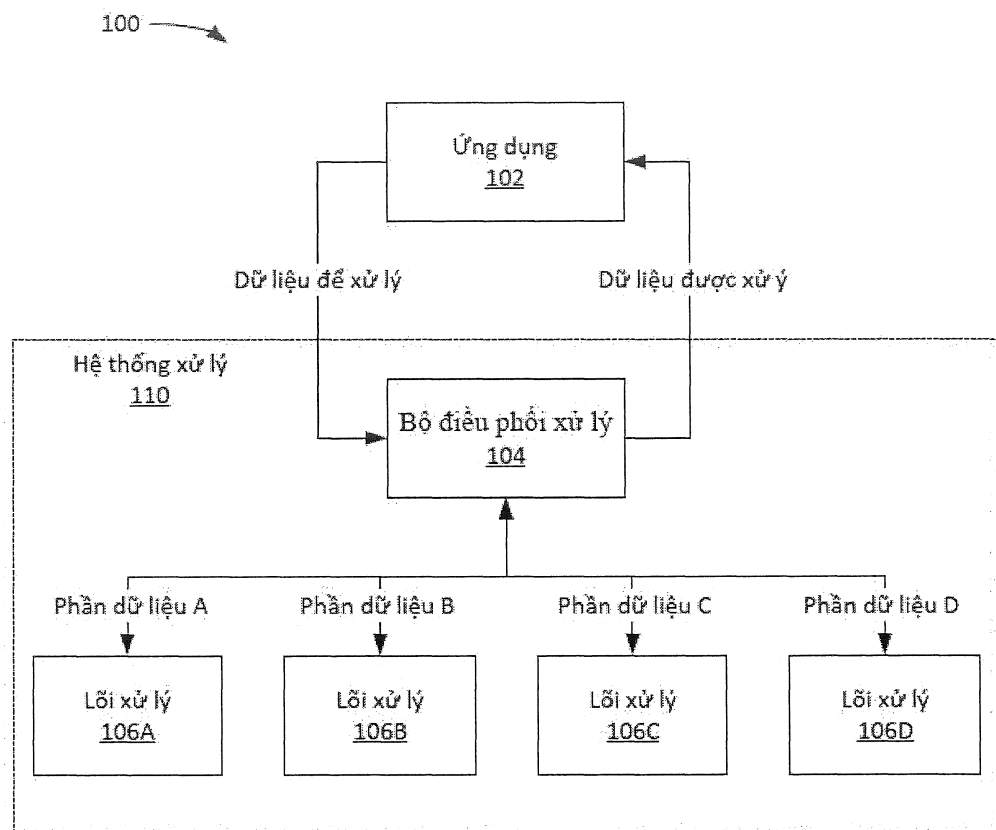


FIG. 1

Lĩnh vực kỹ thuật được đề cập

Các khía cạnh của sáng chế đề cập đến các kỹ thuật để sửa đổi các luồng dữ liệu xử lý để giảm tác động gián đoạn điện năng và ngoài ý muốn trong khi xử lý song song.

Tình trạng kỹ thuật của sáng chế

Xử lý song song ngày càng được triển khai để cải thiện hiệu suất của các tác vụ xử lý dữ liệu lớn. Ví dụ, hiệu suất của hệ thống học máy và trí tuệ nhân tạo thường có thể được cải thiện bằng cách thực hiện xử lý song song trong một hệ thống xử lý bao gồm nhiều khối xử lý hoặc trên nhiều hệ thống xử lý.

Mặc dù việc xử lý song song có thể tạo ra lợi ích hiệu suất đáng kể trong nhiều bối cảnh xử lý dữ liệu lớn, nó cũng có thể tạo ra áp lực đáng kể đối với các hệ thống xử lý. Áp lực như vậy có thể dẫn đến các tác động gián đoạn điện năng, chẳng hạn như tăng đột biến dòng điện và độ dốc điện áp, ảnh hưởng tiêu cực đến hiệu suất của hệ thống xử lý. Ví dụ, nếu nhiều khối xử lý của một hệ thống xử lý, chẳng hạn như một hệ thống trên chip (system on a chip - SoC), khởi tạo tác vụ xử lý dữ liệu chuyên sâu đồng thời, toàn bộ hệ thống xử lý có thể sẽ xuống cấp hoặc không ổn định trong thời gian dòng điện tăng đột biến và/hoặc độ dốc điện áp sau đó. Trong những trường hợp như vậy, lợi ích của kiến trúc xử lý song song bị giảm đi đáng kể và hiệu suất cho người dùng cuối bị ảnh hưởng. Những vấn đề này có thể đặc biệt nghiêm trọng ở các thiết bị di động khi các cân nhắc về nguồn điện được khuếch đại.

Do đó, cần có các hệ thống và phương pháp sửa đổi luồng dữ liệu xử lý để giảm tác động gián đoạn điện năng trong quá trình xử lý song song.

Bản chất kỹ thuật của sáng chế

Theo một số khía cạnh nhất định, sáng chế đề xuất phương pháp thực hiện xử lý dữ liệu song song, phương pháp này bao gồm các bước: nhận dữ liệu để xử lý song song từ bộ yêu cầu; tạo nhiều khối dữ liệu con; xác định nhiều phần dữ liệu trong mỗi khối dữ liệu con trong số nhiều khối dữ liệu con; thay đổi thứ tự của nhiều phần dữ

liệu trong ít nhất một khối dữ liệu con trong số nhiều khối dữ liệu con; cung cấp nhiều khối dữ liệu con, gồm ít nhất một khối dữ liệu con bao gồm thứ tự đã thay đổi của nhiều phần dữ liệu, cho nhiều khối xử lý để xử lý song song; và nhận dữ liệu được xử lý được liên kết với nhiều khối dữ liệu con từ nhiều khối xử lý.

Theo các khía cạnh khác, sáng chế đề xuất phương pháp thực hiện xử lý dữ liệu song song, phương pháp này bao gồm các bước: nhận dữ liệu để xử lý song song từ bộ yêu cầu; xác định nhiều phạm vi mật độ, trong đó mỗi phạm vi mật độ của nhiều phạm vi mật độ khác biệt với từng phạm vi mật độ khác của nhiều phạm vi mật độ; xác định nhiều phần dữ liệu của dữ liệu dựa trên các phạm vi mật độ; tạo nhiều khối dữ liệu con có các phần dữ liệu được tạo chuỗi lại; cung cấp nhiều khối dữ liệu con cho nhiều khối xử lý để xử lý song song; và nhận dữ liệu được xử lý được liên kết với nhiều khối dữ liệu con từ nhiều khối xử lý.

Theo các khía cạnh khác, sáng chế đề xuất phương tiện bất biến đọc được bằng máy tính bao gồm các lệnh mà, khi được thực thi bởi một hoặc nhiều bộ xử lý của hệ thống xử lý, thực hiện các phương pháp nêu trên cũng như các phương pháp khác được mô tả ở đây.

Theo các khía cạnh khác, sáng chế đề xuất sản phẩm chương trình máy tính được đặt trên phương tiện lưu trữ đọc được bằng máy tính bao gồm mã để thực hiện các phương pháp nêu trên cũng như các phương pháp khác được mô tả ở đây.

Các khía cạnh khác còn đề xuất hệ thống xử lý với một hoặc nhiều bộ xử lý và/hoặc các phương tiện khác được tạo cấu hình để thực hiện các phương pháp nêu trên cũng như các phương pháp khác được mô tả ở đây.

Phần mô tả sau đây và các hình vẽ liên quan trình bày chi tiết các tính năng minh họa nhất định của một hoặc nhiều ví dụ.

Mô tả vắn tắt các hình vẽ

Các hình vẽ kèm theo mô tả các khía cạnh nhất định của một hoặc nhiều ví dụ và do đó không được coi là giới hạn phạm vi của sáng chế này.

Fig.1 mô tả kiến trúc xử lý dữ liệu song song ví dụ.

Fig.2 mô tả ví dụ tạo nhiều khối dữ liệu con dựa trên khối dữ liệu nhận được bởi bộ điều phối xử lý.

Fig.3 mô tả kiến trúc xử lý dữ liệu song song ví dụ khác.

Fig.4 mô tả ví dụ việc tạo chuỗi lại động của dữ liệu để xử lý song song để giảm các tác động về điện năng.

Fig.5 mô tả phương pháp ví dụ dữ liệu tạo chuỗi lại động để xử lý song song.

Fig.6 mô tả ví dụ khác về việc tạo chuỗi lại động của dữ liệu để xử lý song song để giảm các tác động về điện năng.

Fig.7 mô tả phương pháp ví dụ về tạo chuỗi lại động của dữ liệu để xử lý song song sử dụng tạo bin mật độ.

Fig.8 mô tả hệ thống xử lý ví dụ được tạo cấu hình để thực hiện xử lý song song sử dụng tạo chuỗi lại dựa trên mật độ.

Fig.9 mô tả biểu đồ mô tả mối quan hệ ví dụ giữa mật độ trong dữ liệu mạng neuron tích chập và mức tiêu thụ điện của khối xử lý.

Fig.10 mô tả ví dụ xử lý song song các khối dữ liệu đa chiều.

Các hình vẽ từ Fig.11A đến Fig.11.C mô tả các tác động về điện năng ví dụ liên quan đến xử lý song song.

Fig.12 mô tả ví dụ sử dụng tạo bin mật độ trên dữ liệu học máy.

Fig.13 mô tả sự so sánh xử lý thông thường của các khối dữ liệu học máy với các nhân so với xử lý tạo chuỗi lại mật độ của các khối dữ liệu học máy.

Các hình vẽ từ Fig.14A đến Fig.14C mô tả các tác động về điện năng ví dụ liên quan đến xử lý song song sử dụng tạo bin mật độ.

Fig.15 mô tả thuộc tính hữu dụng các hoạt động học máy nhất định khi sử dụng dữ liệu đầu vào tạo chuỗi lại theo mật độ.

Fig.16 mô tả thiết bị điện tử ví dụ có thể được tạo cấu hình để thực hiện các phương pháp được mô tả ở đây.

Fig.17 mô tả hệ thống xử lý nhiều bộ xử lý ví dụ có thể được triển khai với các ví dụ được mô tả ở đây.

Để dễ hiểu, các số tham chiếu giống nhau được sử dụng, nếu có thể, để chỉ định các phần tử giống nhau chung trong các hình vẽ. Cần hiểu rằng các yếu tố và đặc điểm của một ví dụ có thể được kết hợp một cách có lợi trong các ví dụ khác mà không cần đọc lại thêm.

Mô tả chi tiết sáng chế

Các khía cạnh được mô tả ở đây liên quan tới các hệ thống và các phương pháp để sửa đổi các luồng dữ liệu xử lý để giảm các tác động về điện năng không mong muốn và gián đoạn trong khi xử lý song song. Các hệ thống và các phương pháp cải thiện hiệu suất của các hệ thống xử lý song song, chẳng hạn như các hệ thống bao gồm nhiều khối xử lý và/hoặc các hệ thống điều phối nhiều hệ thống xử lý.

Nói chung, khả năng xử lý song song có thể được cung cấp trong một bộ xử lý với nhiều khối xử lý (ví dụ, nhiều lõi xử lý của bộ xử lý), bởi nhiều bộ xử lý, mà mỗi bộ xử lý có thể có một hoặc nhiều khối xử lý, bởi nhiều hệ thống xử lý, mỗi hệ thống xử lý có các bộ xử lý của chúng, và bằng cách kết hợp chúng. Nói chung, nhiều bộ xử lý được sử dụng để xử lý song song có thể là thiết kế đồng nhất hoặc không đồng nhất, như là nhiều khối xử lý trung tâm (central processing unit - CPU), hoặc một khối xử lý trung tâm, khối xử lý đồ họa (graphical processing unit - GPU), khối xử lý tín hiệu nơron (neural signal processing unit NPU), bộ xử lý tín hiệu kỹ thuật số (digital signal processor - DSP), và tương tự hoạt động phối hợp. NPU, như được mô tả ở đây, nói chung là một mạch chuyên biệt được tạo cấu hình để triển khai tất cả logic số học và điều khiển cần thiết để thực thi các thuật toán học máy, chẳng hạn như các thuật toán xử lý mạng nơron nhân tạo (artificial neural network - ANN), mạng nơron sâu (deep neural network - DNN), rừng ngẫu nhiên (random forest - RF), và tương tự.

Như được mô tả chi tiết hơn dưới đây, sửa đổi các luồng dữ liệu xử lý để giảm các tác động gián đoạn điện năng trong khi xử lý song song có thể được thực hiện bằng cách phân tích dữ liệu để xử lý song song và tạo chuỗi lại (hoặc sắp xếp lại, sắp xếp thứ tự lại, hoặc thay đổi) dữ liệu tránh tăng đột biến năng lượng dòng đồng thời từ nhiều khối xử lý. Do đó, các luồng dữ liệu xử lý được tạo chuỗi lại có thể làm giảm

hoặc loại bỏ các tác động về điện năng không mong muốn, chẳng hạn như tăng đột biến công suất xử lý, độ dốc điện áp, quá nhiệt, dòng điện rò rỉ bổ sung và các tác động khác, trong hệ thống xử lý, giúp cải thiện hiệu suất và độ tin cậy của hệ thống xử lý.

Hơn nữa, các phương pháp được mô tả ở đây có thể làm giảm dòng điện cực đỉnh một cách có lợi trong quá trình xử lý song song mà không sử dụng các phương pháp làm giảm hiệu suất thông thường, bất lợi, chẳng hạn như điều chỉnh tần số và dùng lệnh, để nêu một vài ví dụ.

Hơn nữa, tránh tăng đột biến công suất xử lý lớn có thể có lợi làm giảm tải nhiệt trong hệ thống (từ dòng điện quá độ, cao). Giảm tải nhiệt có thể làm giảm yêu cầu làm mát và do đó ảnh hưởng tích cực đến khả năng đóng gói. Ngoài ra, giảm tải nhiệt có thể làm giảm dòng rò do nhiệt và do đó ảnh hưởng tích cực đến hiệu quả sử dụng điện.

Các lợi ích nói trên có thể cải thiện đáng kể hiệu suất của các thiết bị di động nói riêng do những hạn chế về thiết kế nguồn vốn có của chúng.

Kiến trúc xử lý dữ liệu song song ví dụ

Fig.1 mô tả kiến trúc xử lý dữ liệu song song ví dụ 100. Theo ví dụ được mô tả, ứng dụng 102 cung cấp dữ liệu để xử lý cho hệ thống xử lý dữ liệu 110, bao gồm bộ điều phối xử lý dữ liệu 104 để điều phối xử lý song song. Ứng dụng 102 có thể là ứng dụng nội bộ (ví dụ, trên hệ thống xử lý) hoặc ứng dụng từ xa (ví dụ, trong kiến trúc máy khách-máy chủ) và có thể thường được gọi là bộ yêu cầu xử lý dữ liệu.

Bộ điều phối xử lý 104 có thể thường là chức năng được triển khai trong hệ thống xử lý được tạo cấu hình để điều phối xử lý song song dữ liệu, như là trong hệ thống tính toán nhiều bộ xử lý (ví dụ, máy tính, máy chủ, thiết bị di động, hệ thống trên chip (SoC), thiết bị internet vạn vật (internet of things - IoT) hoặc hoặc thiết bị tiên tiến, hoặc nói chung trong bất kỳ hệ thống nhiều khối xử lý nào.

Trong ví dụ này, bộ điều phối xử lý dữ liệu 104 nhận dữ liệu từ ứng dụng 102 và điều phối xử lý song song dữ liệu bằng cách cung cấp các khối dữ liệu con A-D cho các lõi xử lý 106A-D. Các khối dữ liệu con A-D có thể là dữ liệu thuộc loại bất kỳ để

xử lý song song. Theo một số ví dụ, các khối dữ liệu con A-D có thể gồm dữ liệu đầu vào mô hình học máy hoặc mô hình trí tuệ nhân tạo, như là dữ liệu hình ảnh, dữ liệu video, dữ liệu âm thanh, hoặc dữ liệu định lượng khác.

Các khối dữ liệu con có thể thường liên quan tới các khối, ô, tensor, hoặc tương tự của dữ liệu được gửi tới nhiều khối xử lý (như là các lõi xử lý 106A-D) để xử lý song song. Theo một số ví dụ, mỗi khối dữ liệu con có thể biểu diễn tập con duy nhất của tập dữ liệu để xử lý, trong khi theo các ví dụ khác, mỗi khối dữ liệu con có thể biểu diễn tập hợp dữ liệu được lặp lại cần được xử lý theo cách khác trên mỗi trong số các khối xử lý, như là xử lý cùng một dữ liệu với các mô hình khác nhau. Do đó, việc tạo các khối dữ liệu con có thể bao gồm bước tạo tập hợp con dữ liệu loại trừ lẫn nhau, hoặc các tập dữ liệu lặp lại, để phân phối tới nhiều khối xử lý.

Fig.2 mô tả ví dụ tạo nhiều khối dữ liệu con bởi bộ điều phối xử lý dữ liệu, như là bộ điều phối xử lý 104 trên Fig.1.

Trong ví dụ này, khối dữ liệu đa chiều 202 được cung cấp cho bộ điều phối xử lý 104 để xử lý bởi nhiều thiết bị xử lý (ví dụ, để xử lý song song), như là nhiều lõi xử lý 106A-D. Theo một số ví dụ, khối dữ liệu 202 có thể là tensor. Nói chung, tensor bậc n trong không gian m chiều là một đối tượng toán học có n chỉ số và m^n thành phần và tuân theo các quy tắc biến đổi nhất định. Mỗi chỉ số của một tensor dao động trên số kích thước của không gian. Do đó, tensor có thể là dạng tổng quát của đại lượng vô hướng (không có chỉ số), vector (có chính xác một chỉ số) và ma trận (có chính xác hai chỉ số) cho số lượng chỉ số tùy ý.

Khối dữ liệu 202 gồm các phần khác nhau 202A-D, mà có các mật độ khác nhau (hoặc ngược lại, độ thưa thớt), mà được chỉ báo thường bằng các hình dạng khác nhau của các phần của chúng. Trong ví dụ này, phần 202B là phần được tô bóng tối nhất, và do đó, là phần dữ liệu dày đặc nhất của dữ liệu 202, trong khi phần 202A và 202C có màu sáng hơn một bậc và do đó ít dày đặc hơn phần 202B, nhưng dày đặc hơn phần 202D. Phần dữ liệu 202D là bóng sáng nhất và ít dày đặc nhất (hoặc thưa thớt nhất) trong số các phần dữ liệu trong khối dữ liệu 202.

Theo một ví dụ, mật độ của phiên bản dữ liệu đa chiều có thể được đo bằng tỷ lệ các mục nhập không rỗng trong các chiều khác nhau của dữ liệu, chẳng hạn như

trong mảng, vectơ, ma trận hoặc định dạng dữ liệu đa chiều khác. Tương tự, độ thừa thớt của phiên bản dữ liệu đa chiều có thể được đo bằng tỷ lệ các mục nhập rỗng trong các chiều khác nhau của dữ liệu. Theo ví dụ đơn giản, vectơ dữ liệu có các mục [1, 0, 1] sẽ có mật độ là 2/3 và độ thừa thớt là 1/3. Nói chung, dữ liệu dày đặc hơn đòi hỏi nhiều sức mạnh xử lý hơn, trong khi dữ liệu thưa hơn yêu cầu ít sức mạnh xử lý hơn.

Đáng chú ý là mật độ và cách bố trí khác nhau của các phần dữ liệu (202A-D) trên Fig.2 chỉ là một ví dụ đơn giản để dễ giải thích. Dữ liệu có thể có các đặc điểm mật độ phức tạp hơn đáng kể.

Thứ tự xử lý 203 của dữ liệu 202 chỉ báo rằng trong khi xử lý dữ liệu 202, các mật độ dữ liệu khác nhau sẽ gặp phải tại các thời điểm khác nhau bởi các lỗi xử lý 106A-D.

Trong ví dụ này, bộ điều phối xử lý 104 chia dữ liệu 202 vào trong bốn khối con 204A-D, mà mỗi khối con bao gồm tập con của các phần 202A-D như được chỉ báo bởi các dải bóng mờ phù hợp. Như được mô tả trong biểu đồ 206, xử lý các khối con dữ liệu 204A-D bằng các lỗi xử lý 106A-D theo hướng xử lý 203 dẫn đến tổng công suất hệ thống xử lý tăng đột biến đáng kể ở mức 208, tương ứng với phân đoạn thời gian 2 dọc theo trục thời gian. Điều này là do mỗi lỗi xử lý (106A-D) tiến đến phần dữ liệu mật độ cao hơn 202B của các khối con dữ liệu 204A-D đồng thời (hoặc gần như vậy).

Mức tăng đột biến công suất ở mức 208 là một ví dụ về tác động về điện năng không mong muốn, có thể gây ra các vấn đề phụ khác nhau cho hệ thống xử lý, chẳng hạn như độ dốc điện áp trên bus nguồn của hệ thống xử lý, có thể dẫn đến suy giảm liên quan, chẳng hạn như điều chỉnh tần số bộ xử lý, lệnh đình trệ, hệ thống không ổn định, và tương tự. Hơn nữa, để giảm thiểu khả năng xảy ra những vấn đề này, có thể yêu cầu điện áp hoạt động cao hơn, làm tăng mức tiêu hao điện.

Như được mô tả chi tiết hơn, bên dưới, vấn đề được mô tả trong biểu đồ 206 có thể được khắc phục bằng cách xác định mật độ dữ liệu trong các khối dữ liệu con (ví dụ, trong các khối dữ liệu con 204A-D) và tạo chuỗi lại dữ liệu để tránh đồng thời các nhu cầu xử lý cao điểm tại mỗi lỗi xử lý (ví dụ, 106A-D trong ví dụ này).

Kiến trúc xử lý dữ liệu song song với tạo chuỗi lại dữ liệu

Fig.3 mô tả ví dụ khác về kiến trúc xử lý dữ liệu song song 300. Theo ví dụ được mô tả, ứng dụng 302 cung cấp dữ liệu để xử lý cho hệ thống xử lý dữ liệu 310, bao gồm bộ điều phối xử lý dữ liệu 304, bộ phân tích mật độ dữ liệu 305, và bộ tạo chuỗi dữ liệu 307. Một lần nữa, ứng dụng 302 có thể được gọi là bộ yêu cầu xử lý dữ liệu và có thể là cục bộ hoặc từ xa đối với hệ thống xử lý 310.

Như trên, bộ điều phối xử lý dữ liệu 304 có thể thường được tạo cấu hình để điều phối xử lý song song dữ liệu trong hệ thống xử lý dữ liệu nhiều khối xử lý, như là hệ thống xử lý 310. Trong ví dụ này, bộ điều phối xử lý dữ liệu 304 nhận dữ liệu từ ứng dụng 302 và tạo các khối dữ liệu con. Như trên, dữ liệu nhận được có thể là tensor.

Theo một số ví dụ, bộ điều phối xử lý 304 có thể được triển khai là một phần của trình biên dịch được tạo cấu hình để xác định cách chia và phân bổ các khối dữ liệu con và các lõi xử lý trong thời gian biên dịch. Theo các ví dụ khác, bộ điều phối xử lý 304 có thể được triển khai như một phần của bộ lập lịch tác vụ hoặc nhân trong hệ điều hành được tạo cấu hình để xác định cách chia và phân bổ các khối dữ liệu con và các lõi xử lý trong thời gian chạy. Đáng chú ý, đây chỉ là hai cách triển khai ví dụ và những cách khác đều có thể thực hiện được.

Không giống ví dụ trên Fig.1 và Fig.2, ở đây các khối dữ liệu con sau đó được phân tích bởi bộ phân tích mật độ 305 trước khi xử lý. Bộ phân tích mật độ 305 được tạo cấu hình để xác định các mật độ (hoặc, cách khác là độ thừa thớt) của các khối dữ liệu con đến. Theo một số ví dụ, việc xác định mật độ dẫn đến việc nhận dạng nhiều phần dữ liệu khác được liên kết với các mật độ khác (hoặc phạm vi mật độ). Theo một số ví dụ, các phần dữ liệu được sắp xếp theo hướng xử lý.

Bộ phân tích mật độ 305 cung cấp các khối dữ liệu con và các mật độ được xác định cho bộ tạo chuỗi 307, mà sau đó tạo chuỗi lại các khối dữ liệu con riêng lẻ được dựa trên các mật độ được xác định để giảm hoặc loại bỏ sự trùng hợp của các phần dữ liệu mật độ đỉnh đang được xử lý bởi nhiều lõi xử lý cùng một lúc.

Theo một ví dụ, bộ tạo chuỗi 307 có thể tạo ra các khối dữ liệu con được tạo chuỗi lại bằng cách thay đổi (ví dụ, bằng cách xáo trộn hoặc dịch chuyển) thứ tự của

các phần dữ liệu (ví dụ, theo thứ tự hoặc thời gian xử lý) trong mỗi khối con để giảm sự đồng nhất của mật độ đỉnh trong thứ tự xử lý của các phần dữ liệu trên nhiều khối xử lý, chẳng hạn như các lõi xử lý 306A-D.

Trong ví dụ khác, bộ tạo chuỗi 307 có thể tạo ra các khối dữ liệu con được tạo chuỗi lại bằng cách truy xuất dữ liệu từ các bin mật độ khác nhau theo thứ tự cụ thể. Ví dụ, bộ tạo chuỗi 307 có thể tạo chuỗi lại các khối dữ liệu con bằng cách vẽ dữ liệu từ các bin mật độ theo thứ tự vòng tròn của các mật độ dữ liệu, như được mô tả chi tiết hơn với tham chiếu tới Fig.6. Điều này cũng làm giảm sự đồng nhất của mật độ đỉnh trong thứ tự xử lý trên nhiều khối xử lý, như là các lõi xử lý 306A-D.

Bộ tạo chuỗi 307 có thể được tạo cấu hình không những để giảm sự trùng hợp của các phần dữ liệu mật độ đỉnh, mà còn để giảm sự chênh lệch trung bình trong mật độ dữ liệu (hoặc độ thừa thớt) giữa các phần dữ liệu liên tiếp (ví dụ, theo thứ tự xử lý). Theo một ví dụ, điều này có thể được thực hiện bằng cách tạo chuỗi lại các phần dữ liệu khác nhau theo thứ tự mật độ giảm hoặc tăng sao cho mật độ của phần dữ liệu x_i là $x_{i-1} < x_i < x_{i+1}$ hoặc $x_{i-1} > x_i > x_{i+1}$. Thứ tự này dựa trên mật độ (hoặc độ thừa thớt) này có lợi làm giảm kích thước của sự thay đổi công suất xử lý (ví dụ, trong hiện tại) giữa các phần dữ liệu có mật độ khác nhau.

Theo các phương án được mô tả, bộ tạo chuỗi 307 cung cấp các khối dữ liệu con được tạo chuỗi lại lần lượt đến xử lý các lõi 306A-D, được mô tả và minh họa với tham chiếu tới Fig.4. Như còn được định nghĩa với tham chiếu tới Fig.6, theo các ví dụ khác, mỗi lõi xử lý (ví dụ, 306A-D) có thể được gán thứ tự dựa trên mật độ mặc định và được tạo cấu hình để đọc các phần dữ liệu dựa trên thứ tự dựa trên mật độ mặc định.

Theo một số ví dụ, sau khi xử lý, bộ tạo chuỗi 307 có thể nhận dữ liệu được xử lý và tạo chuỗi lại để khớp với thứ tự chuỗi gốc. Theo cách này, việc xác định mật độ và việc tạo chuỗi có thể được triển khai mà không ảnh hưởng đến chức năng phối hợp xử lý dữ liệu song song hiện có. Tuy nhiên, theo các ví dụ khác, các lõi xử lý 306A-D có thể cung cấp dữ liệu được xử lý trực tiếp tới bộ điều phối xử lý 304 nếu chuỗi các kết quả xử lý dữ liệu không quan trọng với kết quả xử lý dữ liệu tổng thể.

Các ví dụ về bộ phân tích mật độ (305) và/hoặc bộ tạo chuỗi (307) có thể được triển khai dưới dạng logic phần cứng và/hoặc logic phần mềm. Ví dụ, theo một số ví dụ, như được mô tả chi tiết hơn dưới đây với tham chiếu tới Fig.8, bộ phân tích mật độ (305) và bộ tạo chuỗi (307) được triển khai dưới dạng một phần của DMA, bao gồm logic phần cứng và firmware cấp thấp.

Mặc dù được hiển thị dưới dạng các khía cạnh riêng trong ví dụ này, trong các ví dụ khác, các chức năng của bộ phân tích mật độ 305 và bộ tạo chuỗi 307 có thể được hợp nhất trong một phần tử duy nhất, chẳng hạn như trong bộ điều phối xử lý 304, hoặc theo khía cạnh khác của hệ thống xử lý 310. Các chức năng khác nhau này là được thể hiện độc lập trên Fig.3 cho rõ ràng.

Ví dụ về việc tạo chuỗi lại động của dữ liệu để xử lý song song

Fig.4 mô tả ví dụ việc tạo chuỗi lại động của dữ liệu để xử lý song song để giảm các tác động về điện năng.

Theo ví dụ được mô tả, khối dữ liệu đa chiều 302 được cung cấp bởi ứng dụng, như là được mô tả ở trên Fig.3, để xử lý dữ liệu song song. Một lần nữa, khối dữ liệu 302 có các phần (ví dụ, 302A-D) của mật độ dữ liệu (hoặc độ thưa thớt) khác nhau. Trong ví dụ này, phần 302B là phần được tô đậm nhất, và do đó là phần dữ liệu dày đặc nhất, trong khi các phần 302A và 302C có màu sáng hơn một bậc và do đó ít dày đặc hơn phần 302B, nhưng dày đặc hơn phần 302D. Phần dữ liệu 302D là được tô nhẹ nhất và ít dày đặc nhất (hoặc thưa thớt nhất) trong số các phần dữ liệu trong khối dữ liệu 302.

Đáng chú ý, trong ví dụ này, mật độ của mỗi phần 302A-D là đồng nhất, nhưng theo các ví dụ khác, mật độ trên một hoặc nhiều chiều của khối dữ liệu đa chiều 302 có thể không đồng nhất. Ví dụ, khối dữ liệu 302 có thể biểu diễn dữ liệu hình ảnh và mỗi trong các phần dữ liệu 302A-D có thể biểu diễn kênh (hoặc lớp) của hình ảnh khác với mật độ có thể đo được của riêng nó. Như trên, mật độ có thể được đo bằng số lượng mục nhập rỗng hoặc bằng không trong một phần dữ liệu, chẳng hạn như lát của dữ liệu hình ảnh có kích thước chiều rộng (w) x chiều cao (h) x chiều sâu (d).

Bộ điều phối xử lý dữ liệu 304 nhận khối dữ liệu 302 và tạo nhiều khối dữ liệu con để xử lý song song, sau đó được cung cấp cho bộ phân tích mật độ 305. Bộ phân tích mật độ 305 phân tích các khối dữ liệu con và, ví dụ, nhận dạng các phần của các khối dữ liệu con có mật độ khác nhau. Theo một ví dụ, các phần có thể tương ứng với nhiều phạm vi mật độ xác định trước.

Bộ phân tích mật độ 305 sau đó cung cấp các khối dữ liệu con và dữ liệu mật độ cho bộ tạo chuỗi 307, tạo chuỗi lại các khối dữ liệu con theo cách động được dựa trên dữ liệu mật độ được xác định.

Trong ví dụ được mô tả trên Fig.4, thứ tự tương đối ban đầu của các phần dữ liệu (302A-D) được duy trì theo các khối dữ liệu con được tạo chuỗi lại 404A-D khi được xem xét theo kiểu vòng tròn, nhưng thời gian xử lý của mỗi phần dữ liệu được dịch trong mỗi khối dữ liệu con bởi việc tạo chuỗi lại. Cụ thể, trong ví dụ này, thứ tự xử lý của các phần dữ liệu trong khối dữ liệu con 404A như trong khối dữ liệu gốc 302 trong khi thứ tự xử lý của các phần dữ liệu trong mỗi khối dữ liệu con tiếp theo được tăng thêm một vị trí, hoặc nói cách khác, một vị trí cho khối dữ liệu con 404B, hai vị trí cho khối dữ liệu con 404C, và ba vị trí cho khối dữ liệu con 404D. Như được mô tả, khi phần dữ liệu đến cuối khối con, phần tăng tiếp theo của nó sẽ trở lại vị trí ban đầu kiểu vòng tròn. Việc tạo chuỗi lại này của các phần dữ liệu theo thứ tự xử lý của khối dữ liệu con có thể thường được gọi là bước tạo chuỗi lại thời gian xử lý. Việc tạo chuỗi lại thời gian xử lý có thể là bước được xác định để xác định có bao nhiêu vị trí để dịch một phần cụ thể. Trong ví dụ được mô tả, bước là 1.

Kết quả có lợi của bước tạo chuỗi lại thời gian xử lý trên Fig.4 là các khối dữ liệu con được tạo chuỗi lại 404A-D có các phần dữ liệu của các mật độ khác nhau trong các phân đoạn thời gian khác của thứ tự xử lý 406. Sự phân bố của các phần dữ liệu mật độ đỉnh (và do đó công suất xử lý đỉnh) theo các phân đoạn thời gian khác cho các khối dữ liệu con khác giảm theo cách có lợi công suất hệ thống xử lý tổng cộng đỉnh khi được so sánh với sơ đồ 206 trên Fig.2. Cụ thể, bước tạo chuỗi lại thời gian xử lý trong ví dụ này dẫn đến phần dữ liệu dày đặc nhất đang được xử lý trong phân đoạn thời gian thứ hai cho khối dữ liệu con 404A, trong phân đoạn thời gian thứ ba cho khối dữ liệu con 404B, trong phân đoạn thời gian thứ tư cho khối dữ liệu con 404C, và

trong phân đoạn thời gian thứ nhất trong khối dữ liệu con 404D (đã được xoay trở lại phân đoạn đầu tiên của thứ tự xử lý). Hơn nữa, tổng công suất xử lý của hệ thống trên các phân đoạn thời gian 1-4 vẫn nhất quán trong biểu đồ 408 so với có đỉnh quan trọng trong biểu đồ 206 trên Fig.2.

Nói chung, bước tạo chuỗi lại thời gian xử lý tạo nhiều thứ tự xử lý cho các phần dữ liệu khác trong các khối dữ liệu con 404A-D. Trong ví dụ này, mỗi khối dữ liệu con 404A-D có thứ tự xử lý các phần dữ liệu khác nhau, nhưng theo các ví dụ khác có thể có nhiều khối con hơn các phần dữ liệu cho mỗi khối phụ, trong đó trường hợp các thứ tự xử lý nhất định có thể được lặp lại (như được mô tả trên Fig.6).

Đáng chú ý, trong khi số lượng phần dữ liệu bằng số lượng khối dữ liệu con trong ví dụ này, điều này không cần thiết và được sử dụng trong ví dụ này để dễ giải thích. Hơn nữa, trong khi trong ví dụ này số lượng khối dữ liệu con (404A-D) khớp với số lượng lõi xử lý (306A-D), điều này cũng không cần thiết. Trong một số hệ thống xử lý, số lượng lõi khả dụng có thể là động và việc xử lý này có thể được thực hiện theo các tài nguyên xử lý khả dụng theo cách động.

Mặc dù không được mô tả trên Fig.4, theo các ví dụ khác, dữ liệu các phần của các khối con 404A-D có thể còn được tạo chuỗi lại theo thứ tự mật độ tăng hoặc giảm. Nói cách khác, khối dữ liệu con thứ nhất có thể được tạo chuỗi lại như là phần dữ liệu thứ nhất (ở đầu thứ tự xử lý) có mật độ cao nhất và mỗi phần dữ liệu tiếp theo có mật độ thấp hơn phần dữ liệu trước đó, tức là theo thứ tự giảm dần mật độ. Các khối dữ liệu con tiếp theo sau đó có thể được tạo chuỗi lại thời gian như được mô tả ở trên. Theo cách khác, khối dữ liệu con thứ nhất có thể được tạo chuỗi lại sao cho phần dữ liệu thứ nhất có mật độ thấp nhất và mỗi phần dữ liệu tiếp theo có mật độ cao hơn phần dữ liệu trước đó, tức là theo thứ tự tăng dần mật độ. Điều này có thể thường được gọi là tạo chuỗi lại thứ tự mật độ và có thể làm giảm một cách có lợi sự thay đổi về mật độ giữa các phần dữ liệu liên kề và do đó thay đổi công suất xử lý. Khi khối dữ liệu con thứ nhất có các phần dữ liệu của nó được sắp xếp theo mật độ, thì việc tạo chuỗi thời gian xử lý có thể được áp dụng như mô tả ở trên.

Phương pháp ví dụ về việc tạo chuỗi lại động của dữ liệu để xử lý song song

Fig.5 mô tả ví dụ về phương pháp 500 để tạo chuỗi lại dữ liệu theo cách động để xử lý song song. Phương pháp 500 có thể được thực hiện, ví dụ, bởi hệ thống xử lý, như là hệ thống xử lý 310 trên Fig.3 và 804 trên Fig.8. Như được mô tả ở đây, theo một số phương án, các khía cạnh của phương pháp 500 có thể được thực hiện bởi bộ điều khiển DMA, như được mô tả với tham chiếu tới Fig.8.

Phương pháp 500 bắt đầu ở bước 502 nhận dữ liệu để xử lý song song từ bộ yêu cầu xử lý dữ liệu, như là được mô tả với tham chiếu tới các Fig.3, Fig.4, và Fig.6. Theo các phương án khác nhau, bộ yêu cầu xử lý dữ liệu có thể là ứng dụng cục bộ, ứng dụng từ xa, hệ thống con xử lý cục bộ (như là bộ xử lý khác trong cùng hệ thống xử lý), xử lý cục bộ hoặc từ xa, hoặc tương tự.

Phương pháp 500 sau đó tiến hành bước 504 với việc tạo nhiều khối dữ liệu con, như là được mô tả với tham chiếu tới các Fig.3, Fig.4, và Fig.6. Như trên, theo một số khía cạnh, mỗi khối dữ liệu con biểu diễn một tập con của dữ liệu cần được xử lý theo cùng cách thức, trong khi theo các khía cạnh khác, mỗi khối con là tập sao chép của dữ liệu để được xử lý theo các cách thức khác nhau.

Phương pháp 500 sau đó tiến hành bước 506 với việc xác định nhiều phần dữ liệu trong mỗi khối dữ liệu con trong số nhiều khối dữ liệu con. Như trên, các phần dữ liệu có thể được xác định với tham chiếu tới nhiều mật độ hoặc phạm vi mật độ, như là bằng cách tạo bin theo mật độ.

Theo một số khía cạnh, việc xác định nhiều phần dữ liệu trong mỗi khối dữ liệu con trong số nhiều khối dữ liệu con bao gồm các bước: xác định nhiều phạm vi mật độ, trong đó mỗi phạm vi mật độ của nhiều phạm vi mật độ khác biệt với từng phạm vi mật độ khác của nhiều phạm vi mật độ; và xác định nhiều phần dữ liệu dựa trên nhiều phạm vi mật độ.

Phương pháp 500 sau đó tiến hành bước 508 với việc thay đổi (hoặc tạo chuỗi lại) thứ tự của nhiều phần dữ liệu trong ít nhất một khối dữ liệu con trong số nhiều khối dữ liệu con. Trong ví dụ này, việc thay đổi thứ tự của các phần dữ liệu trong các khối dữ liệu con tạo thành khối dữ liệu con được tạo chuỗi lại.

Theo một số khía cạnh, việc thay đổi thứ tự của nhiều phần dữ liệu cho ít nhất một khối dữ liệu con trong số nhiều khối dữ liệu con bao gồm bước: dịch thời gian mỗi phần dữ liệu trong số nhiều phần dữ liệu theo hướng thời gian xử lý theo thứ tự vòng tròn, như là được mô tả với tham chiếu tới các Fig.4 và Fig.6.

Theo một số khía cạnh, việc thay đổi thứ tự của nhiều phần dữ liệu cho ít nhất một khối dữ liệu con trong số nhiều khối dữ liệu con bao gồm bước: sắp xếp nhiều phần dữ liệu theo một trong số thứ tự mật độ tăng hoặc thứ tự mật độ giảm, như là được mô tả với tham chiếu tới các Fig.4 và Fig.6.

Phương pháp 500 sau đó tiến hành bước 510 với việc cung cấp nhiều khối dữ liệu con, gồm ít nhất một khối dữ liệu con bao gồm thứ tự nhiều phần dữ liệu đã thay đổi, tới nhiều khối xử lý để xử lý song song.

Phương pháp 500 sau đó tiến hành bước 512 với việc nhận dữ liệu được xử lý được liên kết với nhiều khối dữ liệu con từ nhiều khối xử lý.

Phương pháp 500 sau đó tiến hành bước 514 với việc cung cấp dữ liệu được xử lý cho bộ yêu cầu xử lý dữ liệu.

Theo một số khía cạnh, phương pháp 500 còn gồm bước thay đổi thứ tự của dữ liệu được xử lý được liên kết với nhiều khối dữ liệu con dựa trên thứ tự gốc của nhiều phần dữ liệu trong mỗi khối dữ liệu con trong số nhiều khối dữ liệu con trước khi cung cấp dữ liệu được xử lý cho bộ yêu cầu xử lý dữ liệu.

Theo một số khía cạnh, phương pháp 500 còn gồm bước thay đổi thứ tự của nhiều phần dữ liệu trong mỗi khối dữ liệu con trong số nhiều khối dữ liệu con.

Theo một số khía cạnh, việc xác định nhiều phần dữ liệu trong mỗi khối dữ liệu con trong số nhiều khối dữ liệu con được thực hiện bởi Bộ điều khiển truy cập bộ nhớ trực tiếp (Direct Memory Access - DMA) truyền thông dữ liệu với nhiều khối xử lý, như là được mô tả với tham chiếu tới Fig.8.

Theo một số khía cạnh, việc thay đổi thứ tự của nhiều phần dữ liệu trong ít nhất một khối dữ liệu con trong số nhiều khối dữ liệu con được thực hiện bởi bộ điều khiển DMA, như là được mô tả với tham chiếu tới Fig.8.

Theo một số khía cạnh, dữ liệu để xử lý song song bao gồm dữ liệu học máy được liên kết với mô hình học máy.

Theo một số khía cạnh, nhiều khối xử lý bao gồm ít nhất một khối xử lý tín hiệu nơron hoặc đơn vị gia tốc AI, như là được mô tả với tham chiếu tới Fig.16.

Theo một số khía cạnh, phương pháp 500 có thể được sử dụng để thực hiện xử lý tích chập mạng nơron trên nhiều khối xử lý tín hiệu nơron (NPU) (hoặc các lõi gia tốc AI). Tương tự, phương pháp 500 có thể được sử dụng để thực hiện xử lý tích chập mạng nơron trên nhiều đơn vị logic số học (arithmetic logic unit - ALU) trong GPU, hoặc nhiều GPU hoặc trên nhiều CPU. Ví dụ, phương pháp 500 có thể được sử dụng để thực hiện xử lý hình ảnh chung trên nhiều ALU bên trong GPU hoặc nhiều GPU.

Ví dụ về việc tạo chuỗi lại động của dữ liệu để xử lý song song được dựa trên tạo bin mật độ

Fig.6 mô tả ví dụ khác về việc tạo chuỗi lại động của dữ liệu để xử lý song song để giảm các tác động về điện năng.

Cụ thể, Fig.6 mô tả bộ điều phối xử lý 604, bao gồm bộ phân tích mật độ 606 và bộ tạo chuỗi 610. Trong ví dụ này, bộ điều phối xử lý 604 nhận khối dữ liệu đa chiều 602, theo một số khía cạnh có thể là tensor. Bộ điều phối xử lý 604 sau đó sử dụng bộ phân tích mật độ 606 để kiểm tra khối dữ liệu 602.

Bộ phân tích mật độ 606 nhận dạng các phần của khối dữ liệu 602 có các mật độ trong các phạm vi định trước được liên kết với các bin mật độ 608A-C. Bộ phân tích mật độ 606 sau đó đặt các phần của khối dữ liệu được nhận dạng 602, gồm 602A-D, vào trong các bin 608A-C được dựa trên các mật độ được xác định. Nói chung, nhiều bin mật độ sẽ bao gồm toàn bộ phạm vi mật độ có thể để tất cả các phần dữ liệu có thể được gán cho bin duy nhất.

Như được mô tả trong ví dụ này, phần dữ liệu 602B là phần dữ liệu mật độ cao nhất và được lưu trữ trong bin 608A dựa trên việc mật độ được xác định của nó nằm trong phạm vi mật độ gắn với bin 608A (ví dụ, mật độ > 66%); phần dữ liệu 602A là phần dữ liệu mật độ trung bình và được lưu trữ trong bin 608B dựa trên việc mật độ được xác định của nó nằm trong phạm vi mật độ gắn với bin 608B (ví dụ, 33% < mật

độ < 66%); và các phần dữ liệu 602C và 602D là các phần dữ liệu mật độ thấp hơn được lưu trữ trong bin 608C dựa trên việc các mật độ được xác định của chúng nằm trong phạm vi mật độ gần với bin 608C (ví dụ, mật độ < 33%).

Trong ví dụ này, bộ tạo chuỗi 610 sau đó tạo các khối dữ liệu con được tạo chuỗi lại 612A-D bằng cách vẽ các phần dữ liệu từ các bin mật độ (608A-C) theo thứ tự xác định, được sửa đổi (ví dụ, tăng dần) cho mỗi khối dữ liệu con.

Theo ví dụ được mô tả, bản đồ chuỗi bin mật độ 611 được sử dụng bởi bộ tạo chuỗi 610 để ánh xạ mỗi lỗi xử lý tới chuỗi bin (hoặc thứ tự) theo kiểu vòng tròn. Ví dụ, lỗi 1 (616A) được ánh xạ tới chuỗi 1 – 2 – 3 (nghĩa là, các bin 608A – 608B – 608C), lỗi 2 (616B) được ánh xạ tới chuỗi 3 – 1 – 2 (nghĩa là, các bin 608C – 608A – 608B), và v.v.. Trong ví dụ này, mỗi chuỗi trong số các chuỗi bin được tăng thêm một bin cho mỗi lỗi xử lý tiếp theo theo thứ tự vòng tròn. Hơn nữa, trong ví dụ này, chuỗi lặp lại bởi lỗi xử lý thứ tư vì chỉ có ba bin. Theo các khía cạnh khác, có thể có nhiều bin hơn hoặc ít bin hơn và nhiều lỗi xử lý hơn hoặc ít lỗi xử lý hơn.

Theo các ví dụ khác, các lỗi xử lý 616A-D có thể được tạo cấu hình để đọc các phần dữ liệu từ các bin mật độ 608A-C theo thứ tự các bin định trước (như là được thể hiện trong 611) mà không cần tác động bởi bộ tạo chuỗi 610. Trong các ví dụ, các phép gán thứ tự bin mật độ định trước có thể được tạo cấu hình tại thời điểm cài đặt trong, ví dụ, ảnh firmware lỗi xử lý.

Sau khi xử lý bởi các lỗi xử lý 616A-D, dữ liệu được xử lý có thể được tạo chuỗi lại và kết hợp lại để tạo thành đầu ra dữ liệu đã xử lý. Theo một số ví dụ, phép ánh xạ chuỗi bin có thể được sử dụng để tạo chuỗi lại dữ liệu được xử lý để khớp chuỗi gốc của đầu vào dữ liệu. Theo cách này, bộ yêu cầu xử lý dữ liệu không cần phải sửa đổi để nhận dữ liệu được xử lý.

Phương pháp ví dụ tạo chuỗi lại động của dữ liệu để xử lý song song sử dụng tạo bin mật độ

Fig.7 mô tả ví dụ về phương pháp 700 để tạo chuỗi lại động dữ liệu để xử lý song song sử dụng tạo bin mật độ.

Phương pháp 700 bắt đầu ở bước 702 với việc nhận dữ liệu để xử lý song song từ bộ yêu cầu xử lý dữ liệu.

Phương pháp 700 sau đó tiến hành bước 704 với việc xác định nhiều phạm vi mật độ, trong đó mỗi phạm vi mật độ của nhiều phạm vi mật độ khác biệt với từng phạm vi mật độ khác của nhiều phạm vi mật độ.

Phương pháp 700 sau đó tiến hành bước 706 với việc xác định nhiều phần dữ liệu của dữ liệu dựa trên các phạm vi mật độ.

Theo một số khía cạnh, việc xác định nhiều phần dữ liệu của dữ liệu dựa trên các phạm vi mật độ bao gồm các bước: xác định mật độ cho mỗi phần dữ liệu tương ứng trong số nhiều phần dữ liệu; và gán mỗi phần dữ liệu tương ứng trong số nhiều phần dữ liệu cho một bin mật độ trong số nhiều bin mật độ dựa trên mật độ được xác định cho phần dữ liệu tương ứng, trong đó mỗi bin mật độ được liên kết với một phạm vi mật độ trong số nhiều phạm vi mật độ, như là được mô tả ở trên với tham chiếu tới Fig.6.

Theo một số khía cạnh, việc xác định nhiều phần dữ liệu của dữ liệu dựa trên các phạm vi mật độ được thực hiện bởi bộ điều khiển truy cập bộ nhớ trực tiếp (DMA) truyền thông dữ liệu với nhiều khối xử lý, như là được mô tả với tham chiếu tới Fig.8. Lợi ích của việc tính toán mật độ (hoặc độ thưa thớt) trong DMA là nó ẩn độ trễ tính toán liên quan, có thể được thực hiện song song với việc di chuyển và đệm dữ liệu cho mỗi phần dữ liệu.

Phương pháp 700 sau đó tiến hành bước 708 với việc tạo nhiều khối dữ liệu con có các phần dữ liệu được tạo chuỗi lại.

Theo một số khía cạnh, việc tạo nhiều khối dữ liệu con có các phần dữ liệu được tạo chuỗi lại bao gồm bước: thêm các phần dữ liệu vào nhiều khối dữ liệu con theo nhiều chuỗi bin, trong đó mỗi chuỗi bin trong nhiều chuỗi bin bao gồm chuỗi duy nhất của nhiều bin mật độ, như là được mô tả với tham chiếu tới Fig.6.

Theo một số khía cạnh, việc tạo nhiều khối dữ liệu con có các phần dữ liệu được tạo chuỗi lại được thực hiện bởi bộ điều khiển DMA, như là được mô tả với tham chiếu tới Fig.8.

Phương pháp 700 sau đó tiến hành bước 710 với việc cung cấp nhiều khối dữ liệu con cho nhiều khối xử lý để xử lý song song.

Theo một số khía cạnh, việc cung cấp nhiều khối dữ liệu con cho nhiều khối xử lý để xử lý song song được thực hiện bởi bộ điều khiển DMA, như là được mô tả với tham chiếu tới Fig.8.

Phương pháp 700 sau đó tiến hành bước 712 với việc nhận dữ liệu được xử lý được liên kết với nhiều khối dữ liệu con từ nhiều khối xử lý.

Theo một số khía cạnh, nhận dữ liệu được xử lý được liên kết với nhiều khối dữ liệu con từ nhiều khối xử lý được thực hiện bởi bộ điều khiển DMA, như là được mô tả với tham chiếu tới Fig.8.

Phương pháp 700 sau đó tiến hành bước 714 với việc cung cấp dữ liệu được xử lý cho bộ yêu cầu xử lý dữ liệu. Theo các phương án khác nhau, bộ yêu cầu xử lý dữ liệu có thể là ứng dụng cục bộ (như là được mô tả với tham chiếu tới các Fig.3 và Fig.6), ứng dụng từ xa, hệ thống con xử lý cục bộ (như là bộ xử lý khác trong cùng hệ thống xử lý), xử lý cục bộ hoặc từ xa, hoặc tương tự.

Theo một số khía cạnh, phương pháp 700 còn bao gồm gán mỗi chuỗi bin trong nhiều chuỗi bin cho một hoặc nhiều khối xử lý trong số nhiều khối xử lý, trong đó mỗi khối dữ liệu con tương ứng trong số nhiều khối dữ liệu con được cung cấp cho một khối xử lý trong số nhiều khối xử lý dựa trên chuỗi bin được gán của khối dữ liệu con tương ứng.

Theo một số khía cạnh, phương pháp 700 còn bao gồm bước xác định nhiều chuỗi bin theo một trong số thứ tự mật độ tăng hoặc thứ tự mật độ giảm; và dịch thời gian thứ tự xử lý.

Theo một số khía cạnh, dữ liệu để xử lý song song là dữ liệu đầu vào mô hình học máy cho mô hình mạng nơron tích chập.

Theo một số khía cạnh, mỗi phần dữ liệu trong số nhiều phần dữ liệu bao gồm bản đồ đặc trưng. Theo một số phương án, bản đồ đặc trưng có thể biểu diễn đầu ra của một bộ lọc được áp dụng cho lớp trước đó của mạng nơron, như là mạng nơron tích chập. Trong một số trường hợp, các bản đồ đặc trưng có thể được gọi là các bản

đồ đặc trưng trung gian, có thể là đại diện cho đầu ra từ các lớp trung gian trong mạng nơron sâu, như là mạng nơron tích chập.

Theo một số khía cạnh, phương pháp 700 còn bao gồm bước tạo chuỗi lại nhân tích chập (convoluntional kernel) (hoặc bộ lọc) cho mỗi khối dữ liệu con tương ứng trong số nhiều khối dữ liệu con dựa trên chuỗi các phần dữ liệu trong khối dữ liệu con tương ứng; và áp dụng nhân tích chập được tạo chuỗi lại cho khối dữ liệu con tương ứng, như là được mô tả với tham chiếu tới Fig.15. Theo các phương án khác nhau, nhân tích chập hoặc bộ lọc là một tập hợp các giá trị hoặc tham số có thể huấn luyện được áp dụng cho dữ liệu đang được xử lý bởi các lớp của mô hình mạng nơron tích chập để tạo dữ liệu đầu ra lớp, như là bản đồ đặc trưng. Nhân tích chập, như là được mô tả trên Fig.15, có thể bao gồm nhiều lớp, có thể được tạo chuỗi lại để khớp dữ liệu đầu vào được tạo chuỗi lại, như là bản đồ đặc trưng đầu vào được tạo chuỗi lại.

Theo một số khía cạnh, nhiều khối xử lý bao gồm ít nhất một khối xử lý tín hiệu nơron hoặc đơn vị gia tốc AI, như là được mô tả với tham chiếu tới Fig.16.

Theo một số khía cạnh, phương pháp 700 có thể được sử dụng để thực hiện xử lý tích chập mạng nơron trên nhiều khối xử lý tín hiệu nơron (NPU) (hoặc các lõi gia tốc AI). Tương tự, phương pháp 700 có thể được sử dụng để thực hiện xử lý tích chập mạng nơron trên nhiều đơn vị logic số học (ALU) trong GPU, hoặc nhiều GPU hoặc trên nhiều CPU. Ví dụ, phương pháp 700 có thể được sử dụng để thực hiện xử lý hình ảnh chung trên nhiều ALU bên trong GPU, hoặc nhiều GPU.

Hệ thống ví dụ để xử lý song song sử dụng tạo chuỗi lại dựa trên mật độ

Fig.8 mô tả hệ thống xử lý ví dụ 804 được tạo cấu hình để thực hiện xử lý song song sử dụng tạo chuỗi lại dựa trên mật độ, như được thảo luận ở đây.

Trong ví dụ này, hệ thống xử lý 804 nhận dữ liệu từ ứng dụng 802 để được xử lý song song sử dụng nhiều khối xử lý, bao gồm nhiều khối xử lý tín hiệu nơron (NPU) 812A-F, như được mô tả chi tiết hơn bên dưới, có thể cách khác được gọi là các lõi hoặc các đơn vị gia tốc AI. Đáng chú ý, theo các ví dụ khác, các số và/hoặc loại khác của các khối xử lý có thể được sử dụng ngoài hoặc thay cho các NPU 812A-F,

như là các loại khác nhau của các khối xử lý được mô tả dưới đây với tham chiếu tới Fig.16.

Dữ liệu từ ứng dụng 802 nhận được bởi bộ điều khiển truy cập bộ nhớ trực tiếp (DMA) 806, mà trong ví dụ này gồm bộ phân tích mật độ 808 và bộ tạo chuỗi 810, như là được mô tả ở trên. Đáng chú ý, theo các ví dụ khác, một hoặc nhiều của bộ phân tích mật độ 808 và bộ tạo chuỗi 810 có thể được tách từ bộ điều khiển DMA 806. Hơn nữa, như trên Fig.6, bộ phân tích mật độ 808 và bộ tạo chuỗi 810 có thể là một phần của bộ điều phối xử lý (không được minh họa). Cách khác, bộ điều phối xử lý riêng có thể được cung cấp và chức năng như được mô tả ở đây.

Theo các ví dụ khác, mỗi NPU có thể được gán thứ tự đọc bin mật độ mặc định mà có thể được theo sau cho mỗi khối con, do đó loại bỏ sự cần thiết để bộ tạo chuỗi 810 để thực hiện chức năng đó. Việc gán thứ tự bin mật độ có thể được định nghĩa, ví dụ, tại thời điểm cài đặt trong, ví dụ, ảnh firmware NPU. Tuy nhiên, bộ tạo chuỗi 810 có thể được sử dụng để ghi đè thứ tự mặc định theo một số phương án.

Trong ví dụ này, bộ điều khiển DMA 806 thực hiện phân tích mật độ và chức năng tạo chuỗi lại (như được mô tả ở trên) và cung cấp các khối dữ liệu con được tạo chuỗi lại (không được minh họa) cho một hoặc nhiều trong số NPU 812A-F để xử lý song song. Các khối dữ liệu con được tạo chuỗi lại có thể được lưu trữ, ví dụ, trong các bộ nhớ được liên kết với từng NPU, chẳng hạn như các bộ nhớ được kết hợp chặt chẽ (tightly coupled memory - TCM) 814A-F. Nói chung, TCM cung cấp bộ nhớ độ trễ thấp để bộ xử lý có thể sử dụng mà không có đặc điểm không thể dự đoán cho bộ nhớ đệm. Ví dụ, một TCM có thể được sử dụng để giữ các quy trình quan trọng, chẳng hạn như các quy trình xử lý gián đoạn hoặc các tác vụ thời gian thực mà tính không xác định của bộ nhớ đệm là không mong muốn.

Theo một ví dụ, hệ thống xử lý 804 có thể được tạo cấu hình để thực hiện xử lý tích chập mạng nơ-ron trên các NPU 812A-F.

Đáng chú ý, Fig.8 chỉ là một ví dụ về các khía cạnh nhất định của hệ thống xử lý, và các khía cạnh khác có thể có mặt và có thể có các cấu hình khác.

Ví dụ về xử lý song song dữ liệu mạng nơron sử dụng việc tạo chuỗi lại dựa trên mật độ

Xử lý song song dữ liệu học máy, như là đối với mạng nơron, đòi hỏi trên các hệ thống xử lý. Nhiều yếu tố có thể dẫn đến tăng cường độ dòng điện đỉnh cũng như tăng cường độ dòng điện trong thời gian ngắn, gây bất lợi cho hiệu suất của hệ thống xử lý như được mô tả ở trên. Ví dụ, các hệ thống xử lý cho dữ liệu học máy có thể thường hoạt động ở tần số cao để xử lý nhanh các lệnh và các luồng khác nhau liên quan đến các hoạt động học máy. Hơn nữa, dữ liệu học máy có thể thường dày đặc.

Để minh họa tác động của dữ liệu học máy dày đặc trên công suất xử lý, Fig.9 mô tả biểu đồ 900 mô tả mối quan hệ ví dụ giữa mật độ trong dữ liệu mạng nơron tích chập và mức tiêu thụ điện của khối xử lý. Như được mô tả, lượng công suất xử lý được sử dụng thường tăng lên khi mật độ của dữ liệu mạng nơron tích chập tăng (nghĩa là, độ thưa thớt giảm). Hơn nữa, mức tiêu thụ điện tăng lên khi mật độ tăng còn tăng lên nữa khi kích thước dữ liệu và độ phức tạp tính toán tăng lên (ví dụ, phép nhân tích lũy dấu phẩy động tiêu thụ nhiều điện năng hơn phép nhân tích lũy số nguyên). Trong ví dụ cụ thể này, việc xử lý dữ liệu tích chập dấu phẩy động 16 bit tiêu thụ nhiều điện hơn 40% so với xử lý dữ liệu tích chập số nguyên 8 bit ở mật độ đầy đủ, chẳng hạn như trong tensor bão hòa.

Fig.10 mô tả ví dụ xử lý song song các khối dữ liệu con đa chiều 1002A-D. Trong ví dụ này, mỗi khối dữ liệu con 1002A-D bao gồm nhiều bản đồ đặc trưng, được xử lý với nhiều nhân tích chập 1004A-D (ví dụ, trong lớp tích chập của mạng nơron). Trong ví dụ này, mật độ của các khối dữ liệu con 1002A-D thay đổi theo các bản đồ đặc trưng khác nhau trong mỗi khối dữ liệu con như được mô tả bằng các nét gạch khác nhau các bản đồ đặc trưng theo chuỗi xử lý tích chập 1010 của mỗi khối con, do đó mỗi bản đồ đặc trưng trong ví dụ này có thể được coi là phần khối dữ liệu con tương ứng của nó.

Biểu đồ 1006 mô tả mức tiêu thụ điện của bốn các khối xử lý 1008A-D. Do mỗi trong số các khối dữ liệu con 1002A-D có các phần dày đặc hơn (ví dụ, các bản đồ đặc trưng) tại cùng vị trí theo chuỗi xử lý tích chập 1010 của các khối dữ liệu con, mức tiêu thụ điện của mỗi khối xử lý 1008A-D là cao nhất tại cùng thời điểm, do đó gây ra

sự gia tăng lớn về công suất xử lý tổng thể, như được mô tả ở 1012. Như đã mô tả ở trên, sự gia tăng lớn này gây bất lợi cho hiệu suất hệ thống vì nó có thể gây ra tác động về điện năng gián đoạn, chẳng hạn như độ dốc điện áp (như được mô tả chi tiết hơn với tham chiếu tới các Fig.11A đến Fig.11C), sau đó có thể ảnh hưởng đến các bộ phận khác của hệ thống, chẳng hạn như gây điều chỉnh các hoạt động xử lý.

Fig.11A mô tả ví dụ biểu đồ mức tiêu thụ điện cho nhiều lỗi xử lý, tương tự như biểu đồ 1006 trên Fig.10.

Fig.11B mô tả ví dụ về tổng mức tiêu thụ điện xử lý 1102 của các lỗi 0-3 trên Fig.11A. Như được mô tả, công suất xử lý tổng thường thay đổi theo thời gian, nhưng nhưng đạt mức tối đa trong mức tăng đột biến công suất ở mức 1105, tương ứng với mức tăng thời gian 4 trên Fig.11A. Hơn nữa, sự thay đổi của dòng điện 1104 (ví dụ, Δi) so với sự thay đổi về thời gian 1106 (ví dụ, Δt) dẫn đến đoạn dốc 1107 (ví dụ, $\Delta i / \Delta t$) của mức tiêu thụ điện. Thay đổi bất ngờ này trong mức tiêu thụ điện, mà do thay đổi bất ngờ trong dòng điện, có thể làm cho các tác động gián đoạn điện năng, như là độ dốc điện áp được mô tả trên Fig.11C.

Fig.11C mô tả ví dụ về điện áp hệ thống 1008 theo thời gian, tương ứng với thời gian với các Fig.11A và Fig.11B. Như được mô tả, độ dốc điện áp 1112 xảy ra tại thời điểm 4 trùng với đỉnh tiêu thụ điện 1105 trên Fig.11B. Đáng chú ý, độ dốc điện áp 1112 đưa điện áp hệ thống ra khỏi dải điện áp hệ thống ổn định 1110, điều này có thể gây ra sự mất ổn định của hệ thống và các hành động làm giảm hiệu suất trên một phần của hệ thống để duy trì sự ổn định, chẳng hạn như điều chỉnh tần số.

Fig.12 mô tả ví dụ sử dụng tạo bin mật độ trên dữ liệu học máy, như là các bản đồ đặc điểm được mô tả trên Fig.10. Cụ thể, bin 1 (1202A) gồm các con trỏ địa chỉ tới dữ liệu bản đồ đặc trưng 1204A, có mật độ 0-25%. Tương tự, bin 2 (1202B) gồm các con trỏ địa chỉ cho dữ liệu bản đồ đặc trưng 1204B, có mật độ 26-50%; bin 3 (1202C) gồm các con trỏ địa chỉ dữ liệu bản đồ đặc trưng 1204C, có mật độ 51-75%; và bin 4 (1202D) gồm các con trỏ địa chỉ tới dữ liệu bản đồ đặc trưng 1204D, có mật độ 76-100%. Việc tạo bin các con trỏ địa chỉ tới dữ liệu bản đồ đặc trưng khác nhau có mật độ khác nhau có thể được thực hiện bởi, ví dụ, bộ phân tích mật độ, như là được mô tả ở trên (ví dụ, với tham chiếu tới Fig.6).

Nhiều khối xử lý có thể sau đó xử lý các bản đồ đặc trưng từ các bin theo cách hiệu quả về điện năng, như là bằng cách giảm độ dốc công suất đỉnh ($\Delta i / \Delta t$). Ví dụ, khối xử lý thứ nhất có thể xử lý một bản đồ đặc trưng từ mỗi bin (ví dụ, 1202A) trước khi xử lý các bản đồ đặc trưng từ bin tiếp theo (ví dụ, 1202B), sao cho thứ tự xử lý thu được trong ví dụ này là Bin 4 (công suất cao nhất), Bin 3, Bin 2, Bin 1 (công suất thấp nhất).

Hơn nữa, như được mô tả ở trên, nhiều khối xử lý có thể làm xáo trộn thứ tự bắt đầu xử lý bin để giảm thiểu độ dốc công suất ban đầu và tổng dòng điện, điều này giảm thiểu tổng công suất xử lý đỉnh cao và công suất rò rỉ. Sử dụng ví dụ trên Fig.12, các thứ tự bắt đầu xử lý cho bốn khối xử lý có thể là: Đơn vị xử lý 1: Bin 4, Bin 3, Bin 2, Bin 1; Đơn vị xử lý 2: Bin 3, Bin 2, Bin 1, Bin 4; Đơn vị xử lý 3: Bin 2, Bin 1, Bin 4, Bin 3; và Đơn vị xử lý 4: Bin 1, Bin 4, Bin 3, Bin 2. Đáng chú ý, các thứ tự này chỉ là một ví dụ, và các thứ tự khác là khả thi.

Khi quá trình xử lý bin bắt đầu bởi các khối xử lý song song, hoạt động tạo bin tiếp theo có thể được kết nối với quá trình xử lý khối xử lý hiện tại bằng cách sử dụng, ví dụ, một bộ đệm tròn.

Fig.13 mô tả sự so sánh xử lý thông thường của các khối dữ liệu con học máy 1002A-D với các hạt nhân 1004A-D với xử lý tạo chuỗi lại mật độ của các khối dữ liệu con học máy 1302A-D với các hạt nhân 1304A-D sử dụng các khối xử lý 1308A-D. Đáng chú ý đây là ví dụ trong đó mỗi khối dữ liệu con 1002A-D và 1302A- có cùng dữ liệu được vận hành bởi một hạt nhân khác nhau.

Như được mô tả, sự phân bố của các mật độ dữ liệu trong các khối con 1002A-D là đồng nhất, dẫn đến công suất tăng đột biến cho các phần dữ liệu dày đặc nhất (được tô bóng tối nhất) như được mô tả ở trên. Ngược lại, sự phân bố của các mật độ trong các khối dữ liệu con 1302A-D thay đổi do việc tạo chuỗi lại dựa trên mật độ của chúng, tạo ra lợi ích được mô tả ở trên và còn với tham chiếu tới các Fig.14A đến Fig.14C. Cụ thể, trong ví dụ này, các khối dữ liệu con 1302A-D có cả thời gian tạo chuỗi lại và tạo chuỗi lại thứ tự mật độ được áp dụng.

Fig.14A mô tả ví dụ biểu đồ của mức tiêu thụ điện cho nhiều lõi xử lý để xử lý các khối dữ liệu con 1302A-D trên Fig.13. Đáng chú ý, công suất đỉnh cho mỗi lõi

được phân bổ là kết quả của việc phân bổ các phần dày đặc nhất của dữ liệu được xử lý, như được mô tả trong các khối dữ liệu con 1302A-D trên Fig.13. Điều này trái ngược với nồng độ công suất xử lý đỉnh trên Fig.11A được dựa trên sự phân bổ của các mật độ trong các khối dữ liệu con 1002A-D trên Fig.10.

Fig.14B mô tả ví dụ về tổng mức tiêu thụ điện xử lý 1402 của các lõi 0-3 từ Fig.14A. Như được mô tả, công suất xử lý tổng vẫn thay đổi theo thời gian, nhưng mức tăng đột biến công suất được mô tả ở mức trên Fig.11B đã được triệt tiêu. Hơn nữa, sự thay đổi của dòng điện 1404 (ví dụ, Δi) so với sự thay đổi về thời gian 1406 (ví dụ, Δt) dẫn đến độ dốc công suất thoải hơn 1407 (ví dụ, $\Delta i / \Delta t$) của mức tiêu thụ điện. Các đặc tính mức tiêu thụ điện năng được cải thiện giúp tránh các tác động gián đoạn điện năng, như là độ dốc điện áp, được mô tả ở trên với tham chiếu tới Fig.11C.

Fig.14C mô tả ví dụ về điện áp hệ thống 1408 theo thời gian, tương ứng với thời gian với các Fig.14A và Fig.14B. Như được mô tả, không có độ dốc điện áp đáng kể và điện áp hệ thống vẫn nằm trong dải hoạt động an toàn 1410, khi được so sánh với độ dốc điện áp 1112 được mô tả trên Fig.11C. Do đó, tránh được việc hệ thống không ổn định và các hành động làm giảm hiệu suất trên một phần của hệ thống trong khi xử lý cùng lượng dữ liệu, chỉ theo một thứ tự được tạo chuỗi lại được dựa trên mật độ.

Fig.15 mô tả thuộc tính hữu dụng các hoạt động học máy nhất định khi sử dụng dữ liệu đầu vào tạo chuỗi lại theo mật độ.

Như được mô tả, việc áp dụng chuỗi gốc của các bản đồ đặc trưng 1502A cho nhân 1504A dẫn đến bản đồ đặc trưng đầu ra 1506. Tương tự, nhân được tạo chuỗi lại 1504B có thể được áp dụng cho bản đồ đặc trưng đầu vào 1502B được tạo chuỗi lại (ví dụ, được tạo chuỗi lại theo thời gian và/hoặc mật độ) để có cùng một bản đồ đặc trưng đầu ra 1506. Do đó, các kết quả toán học sau khi tạo chuỗi lại giống như kết quả trước đó và bản đồ đặc trưng đầu ra không cần phải được tạo chuỗi lại một cách độc lập trong kịch bản này. Hơn nữa, các kết quả toán học tương tự cũng đạt được với hiệu suất hệ thống được cải thiện như mô tả ở trên.

Thiết bị điện tử ví dụ để xử lý dữ liệu song song với việc tạo chuỗi lại dữ liệu

Fig.16 mô tả thiết bị điện tử ví dụ 1600 có thể được tạo cấu hình để thực hiện xử lý dữ liệu song song với việc tạo chuỗi lại dữ liệu, như được mô tả ở đây ví dụ với tham chiếu tới các Fig.5 và Fig.7.

Thiết bị điện tử 1600 bao gồm khối xử lý trung tâm (CPU) 1602, theo một số ví dụ có thể là CPU đa lõi. Các lệnh được thực thi tại CPU 1602 có thể được tải, ví dụ, từ bộ nhớ chương trình được liên kết với CPU.

Thiết bị điện tử 1600 cũng bao gồm thành phần xử lý bổ sung phù hợp với các chức năng cụ thể, như là khối xử lý đồ họa (GPU) 1604, bộ xử lý tín hiệu kỹ thuật số (DSP) 1606, khối xử lý tín hiệu nơron (NPU) 1608, khối xử lý đa phương tiện 1610, và thành phần kết nối không dây 1612.

NPU, chẳng hạn như 1608, nói chung là một mạch chuyên biệt được tạo cấu hình để thực hiện tất cả các logic số học và điều khiển cần thiết để thực thi các thuật toán học máy, chẳng hạn như các thuật toán xử lý mạng nơron nhân tạo (ANN), mạng nơron sâu (DNN), rừng ngẫu nhiên (RF), và tương tự. NPU đôi khi có thể được gọi theo cách khác là bộ xử lý tín hiệu nơron (NSP), bộ xử lý tensor (TPU), bộ xử lý mạng nơron (NNP), bộ xử lý thông minh (IPU), bộ xử lý thị giác (VPU) hoặc bộ xử lý đồ thị.

NPU, chẳng hạn như 1608, được tạo cấu hình để tăng tốc hiệu suất của các tác vụ học máy thông thường, chẳng hạn như phân loại hình ảnh, dịch máy, phát hiện đối tượng và nhiều mô hình dự đoán khác nhau khác. Trong một số ví dụ, nhiều NPU có thể được khởi tạo trên một chip, chẳng hạn như hệ thống trên chip (SoC), trong khi trong các ví dụ khác, chúng có thể là một phần của bộ gia tốc mạng nơron chuyên dụng.

NPU có thể được tối ưu hóa cho huấn luyện hoặc suy luận, hoặc trong một số trường hợp được tạo cấu hình để cân bằng hiệu suất giữa cả hai. Đối với các NPU có khả năng thực hiện cả huấn luyện và suy luận, hai tác vụ thường vẫn có thể được thực hiện độc lập.

Các NPU được thiết kế để đẩy nhanh quá trình huấn luyện thường được tạo cấu hình để đẩy nhanh quá trình quản lý các mô hình mới, đây là một hoạt động đòi hỏi

tính toán cao liên quan đến việc nhập một tập dữ liệu hiện có (thường được gắn nhãn hoặc gắn thẻ), lập lại tập dữ liệu và sau đó điều chỉnh các thông số mô hình, chẳng hạn như trọng số và độ lệch, để cải thiện hiệu suất của mô hình. Nói chung, việc tối ưu hóa dựa trên một dự đoán sai bao gồm việc lan truyền ngược lại qua các lớp của mô hình và xác định gradient để giảm lỗi dự đoán.

Các NPU được thiết kế để tăng tốc suy luận thường được tạo cấu hình để hoạt động trên các mô hình hoàn chỉnh. Do đó, các NPU như vậy có thể được tạo cấu hình để nhập một phần dữ liệu mới và nhanh chóng xử lý nó thông qua một mô hình đã được huấn luyện để tạo ra đầu ra mô hình (ví dụ, suy luận).

Trong một triển khai, NPU 1608 là một phần của một hoặc nhiều CPU 1602, GPU 1604 và/hoặc DSP 1606.

Theo một số ví dụ, thành phần kết nối không dây 1612 có thể gồm các thành phần con, ví dụ, cho kết nối thế hệ thứ ba (3G), kết nối thế hệ thứ tư (4G) (ví dụ, 4G LTE), kết nối thế hệ thứ năm (ví dụ, 5G hoặc NR), kết nối Wi-Fi, kết nối Bluetooth và các tiêu chuẩn truyền dữ liệu không dây khác. Thành phần xử lý kết nối không dây 1612 được kết nối thêm với một hoặc nhiều anten 1614.

Thiết bị điện tử 1600 có thể cũng gồm một hoặc nhiều các khối xử lý cảm biến 1616 được liên kết với bất kỳ cách thức nào của cảm biến, một hoặc nhiều bộ xử lý tín hiệu ảnh (ISP) 1618 được liên kết với bất kỳ cách thức nào của cảm biến hình ảnh, và/hoặc bộ xử lý điều hướng 1620, có thể bao gồm vệ tinh các thành phần của hệ thống định vị dựa trên vệ tinh (ví dụ, GPS hoặc GLONASS) cũng như các thành phần của hệ thống định vị quán tính.

Thiết bị điện tử 1600 có thể cũng gồm một hoặc nhiều thiết bị đầu vào và/hoặc thiết bị đầu ra 1622, như là màn hình, bề mặt cảm ứng (bao gồm cả màn hình cảm ứng), nút vật lý, loa, micro, v.v..

Theo một số ví dụ, một hoặc nhiều bộ xử lý của thiết bị điện tử 1600 có thể được dựa trên tập lệnh ARM hoặc RISC-V.

Thiết bị điện tử 1600 cũng gồm bộ nhớ 1624, biểu diễn một hoặc nhiều bộ nhớ tĩnh và/hoặc bộ nhớ động, như là bộ nhớ truy cập ngẫu nhiên động, bộ nhớ tĩnh dựa

trên flash, và tương tự. Trong ví dụ này, bộ nhớ 1624 gồm thành phần thực thi được bởi máy tính, có thể được thực thi bởi một hoặc nhiều trong số các bộ xử lý nói trên của thiết bị điện tử 1600. Mặc dù không được mô tả trên Fig.16, thiết bị điện tử có thể gồm bộ điều khiển truy cập bộ nhớ trực tiếp (DMA).

Cụ thể, trong ví dụ này, bộ nhớ 1624 bao gồm thành phần gửi 1624A, thành phần nhận 1624B, thành phần xử lý 1624C, thành phần xác định 1624D, thành phần đầu ra 1624E, thành phần khối con 1624F, thành phần ước lượng mật độ 1624G, thành phần trình tự (hoặc thứ tự) 1624H và thành phần tọa độ 1624I. Các thành phần được mô tả và các thành phần khác không được mô tả, có thể được tạo cấu hình để thực hiện các khía cạnh khác nhau của các phương pháp được mô tả ở đây.

Nói chung, thiết bị điện tử 1600 và/hoặc các thành phần của nó có thể được tạo cấu hình để thực hiện các phương pháp được mô tả ở đây.

Fig.17 mô tả hệ thống xử lý nhiều bộ xử lý ví dụ 1700 có thể được triển khai với các ví dụ được mô tả ở đây. Ví dụ, hệ thống đa xử lý 1700 có thể biểu diễn các bộ xử lý khác nhau của thiết bị điện tử 1600 trên Fig.16.

Trong ví dụ này, hệ thống 1700 gồm các bộ xử lý 1701, 1703, và 1705, nhưng theo các ví dụ khác, số lượng bất kỳ của các bộ xử lý riêng có thể được sử dụng. Hơn nữa, mặc dù được mô tả tương tự, các bộ xử lý 1701, 1703, và 1705 có thể biểu diễn các loại bộ xử lý khác nhau trong thiết bị điện tử, như là các CPU, các GPU, các DSP, các NPU, và tương tự như được mô tả ở đây.

Mỗi trong các bộ xử lý 1701, 1703, và 1705 gồm bộ lập lịch lệnh, các thành phần phụ con cứng khác nhau (ví dụ, phần cứng X, phần cứng Y và phần cứng Z) và bộ nhớ cục bộ. Trong một số ví dụ, bộ nhớ cục bộ có thể là bộ nhớ kết hợp chặt chẽ (TCM). Lưu ý rằng trong khi các thành phần của mỗi bộ xử lý 1701, 1703 và 1705 được thể hiện là giống nhau trong ví dụ này, trong các ví dụ khác, một số hoặc mỗi bộ xử lý 1701, 1703 và 1705 có thể có cấu hình phần cứng khác nhau, các phần tử phần cứng khác nhau, v.v..

Mỗi trong các bộ xử lý 1701, 1703, và 1705 cũng truyền thông dữ liệu với bộ nhớ chung, chẳng hạn như bộ nhớ DDR hoặc các loại bộ nhớ hoạt động khả biến khác. Ví dụ, bộ nhớ chung 1707 có thể là đại diện cho bộ nhớ 1624 của trên Fig.16.

Trong một số triển khai, trong hệ thống xử lý nhiều bộ xử lý như là 1700, một trong các bộ xử lý có thể hoạt động như bộ xử lý chính. Ví dụ, bộ xử lý 1701 có thể là bộ xử lý chính trong ví dụ này. Bộ xử lý chính có thể gồm trình biên dịch mà, khi được thực thi, có thể xác định cách một mô hình, chẳng hạn như mạng nơ-ron, sẽ được xử lý bởi các thành phần khác nhau của hệ thống xử lý 1700. Ví dụ, sự song song của phần cứng có thể được thực hiện bằng cách ánh xạ các phần của quá trình xử lý một mô hình tới phần cứng khác nhau (ví dụ, phần cứng X, phần cứng Y và phần cứng Z) trong một bộ xử lý nhất định (ví dụ, bộ xử lý 1701) cũng như ánh xạ các phần của quá trình xử lý của mô hình tới các bộ xử lý khác (ví dụ, bộ xử lý 1703 và 1705) và phần cứng liên quan của chúng. Ví dụ, các khối song song trong kiến trúc xử lý khối song song được mô tả ở đây có thể được ánh xạ tới các phần khác nhau của phần cứng khác nhau trong các bộ xử lý 1701, 1703 và 1705.

Các ví dụ

Ví dụ 1: Phương pháp thực hiện xử lý dữ liệu song song, phương pháp này bao gồm các bước: nhận dữ liệu để xử lý song song từ bộ yêu cầu xử lý dữ liệu; tạo nhiều khối dữ liệu con; xác định nhiều phần dữ liệu trong mỗi khối dữ liệu con trong số nhiều khối dữ liệu con; thay đổi thứ tự của nhiều phần dữ liệu trong ít nhất một khối dữ liệu con trong số nhiều khối dữ liệu con; cung cấp nhiều khối dữ liệu con, gồm ít nhất một khối dữ liệu con bao gồm thứ tự thay đổi của nhiều phần dữ liệu, tới nhiều khối xử lý để xử lý song song; và nhận dữ liệu được xử lý được liên kết với nhiều khối dữ liệu con từ nhiều khối xử lý.

Ví dụ 2: Phương pháp theo ví dụ 1, trong đó việc xác định nhiều phần dữ liệu trong mỗi khối dữ liệu con trong số nhiều khối dữ liệu con bao gồm các bước: xác định nhiều phạm vi mật độ, trong đó mỗi phạm vi mật độ của nhiều phạm vi mật độ khác biệt với từng phạm vi mật độ khác của nhiều phạm vi mật độ; và xác định nhiều phần dữ liệu dựa trên nhiều phạm vi mật độ.

Ví dụ 3: Phương pháp theo bất kỳ một trong các ví dụ 1 hoặc 2, trong đó việc thay đổi thứ tự của nhiều phần dữ liệu cho ít nhất một khối dữ liệu con trong số nhiều khối dữ liệu con bao gồm bước: dịch theo thời gian mỗi phần dữ liệu trong số nhiều phần dữ liệu theo hướng thời gian xử lý theo thứ tự vòng tròn.

Ví dụ 4: Phương pháp theo bất kỳ một trong các ví dụ 1-3, trong đó việc thay đổi thứ tự của nhiều phần dữ liệu cho ít nhất một khối dữ liệu con trong số nhiều khối dữ liệu con bao gồm bước: sắp xếp nhiều phần dữ liệu theo một trong số thứ tự mật độ tăng hoặc thứ tự mật độ giảm.

Ví dụ 5: Phương pháp theo bất kỳ một trong các ví dụ 1-4, phương pháp này còn bao gồm các bước: thay đổi thứ tự của dữ liệu được xử lý được liên kết với nhiều khối dữ liệu con dựa trên thứ tự gốc của nhiều phần dữ liệu trong mỗi khối dữ liệu con trong số nhiều khối dữ liệu con; và cung cấp dữ liệu được xử lý cho bộ yêu cầu xử lý dữ liệu.

Ví dụ 6: Phương pháp theo bất kỳ một trong các ví dụ 1-5, phương pháp này còn bao gồm bước: thay đổi thứ tự của nhiều phần dữ liệu trong mỗi khối dữ liệu con trong số nhiều khối dữ liệu con.

Ví dụ 7: Phương pháp theo bất kỳ một trong các ví dụ 1-6, trong đó việc xác định nhiều phần dữ liệu trong mỗi khối dữ liệu con trong số nhiều khối dữ liệu con được thực hiện bởi bộ điều khiển truy cập bộ nhớ trực tiếp (DMA) truyền thông dữ liệu với nhiều khối xử lý.

Ví dụ 8: Phương pháp theo ví dụ 7, trong đó việc thay đổi thứ tự của nhiều phần dữ liệu trong ít nhất một khối dữ liệu con trong số nhiều khối dữ liệu con được thực hiện bởi bộ điều khiển DMA.

Ví dụ 9: Phương pháp theo bất kỳ một trong các Ví dụ 1-8, trong đó việc thay đổi thứ tự của nhiều phần dữ liệu trong ít nhất một khối dữ liệu con trong số nhiều khối dữ liệu con bao gồm bước đọc, bởi nhiều khối xử lý, nhiều phần dữ liệu theo thứ tự định trước.

Ví dụ 10: Phương pháp theo bất kỳ một trong các Ví dụ 1-9, trong đó nhiều khối xử lý bao gồm ít nhất một khối xử lý tín hiệu nơron.

Ví dụ 11: Hệ thống xử lý để thực hiện xử lý dữ liệu song song, hệ thống này bao gồm: bộ nhớ bao gồm các lệnh thực thi được bởi máy tính; một hoặc nhiều bộ xử lý được tạo cấu hình để thực thi các lệnh thực thi được bởi máy tính và khiến cho hệ thống xử lý thực hiện phương pháp theo ví dụ bất kỳ trong số các ví dụ 1-10.

Ví dụ 12: Phương tiện bất biến đọc được bằng máy tính bao gồm lệnh mà, khi được thực thi bởi một hoặc nhiều bộ xử lý của hệ thống xử lý, khiến cho hệ thống xử lý thực hiện phương pháp theo ví dụ bất kỳ trong số các ví dụ 1-10.

Ví dụ 23: Sản phẩm chương trình máy tính nằm trên phương tiện lưu trữ đọc được bằng máy tính và bao gồm mã để thực hiện phương pháp theo ví dụ bất kỳ trong số các ví dụ 1-10.

Ví dụ 14: Phương pháp thực hiện xử lý dữ liệu song song, phương pháp này bao gồm các bước: nhận dữ liệu để xử lý song song từ bộ yêu cầu xử lý dữ liệu; xác định nhiều phạm vi mật độ, trong đó mỗi phạm vi mật độ của nhiều phạm vi mật độ khác biệt với từng phạm vi mật độ khác của nhiều phạm vi mật độ; xác định nhiều phần dữ liệu của dữ liệu dựa trên các phạm vi mật độ; tạo nhiều khối dữ liệu con có các phần dữ liệu được tạo chuỗi lại; cung cấp nhiều khối dữ liệu con cho nhiều khối xử lý để xử lý song song; và nhận dữ liệu được xử lý được liên kết với nhiều khối dữ liệu con từ nhiều khối xử lý.

Ví dụ 15: Phương pháp theo ví dụ 14, trong đó việc xác định nhiều phần dữ liệu của dữ liệu dựa trên các phạm vi mật độ bao gồm các bước: xác định mật độ cho mỗi phần dữ liệu tương ứng trong số nhiều phần dữ liệu; và gán mỗi phần dữ liệu tương ứng trong số nhiều phần dữ liệu cho một bin mật độ trong số nhiều bin mật độ được dựa trên mật độ được xác định cho phần dữ liệu tương ứng, trong đó mỗi bin mật độ được liên kết với một phạm vi mật độ trong số nhiều phạm vi mật độ.

Ví dụ 16: Phương pháp theo ví dụ 15, trong đó việc tạo nhiều khối dữ liệu con có các phần dữ liệu được tạo chuỗi lại bao gồm: thêm các phần dữ liệu vào nhiều khối dữ liệu con theo nhiều chuỗi bin, trong đó mỗi chuỗi bin trong nhiều chuỗi bin bao gồm chuỗi duy nhất của nhiều bin mật độ.

Ví dụ 17: Phương pháp theo ví dụ 16, phương pháp này còn bao gồm các bước: gán mỗi chuỗi bin trong nhiều chuỗi bin cho một hoặc nhiều khối xử lý trong số nhiều khối xử lý, trong đó mỗi khối dữ liệu con tương ứng trong số nhiều khối dữ liệu con được cung cấp cho một khối xử lý trong số nhiều khối xử lý dựa trên chuỗi bin được gán của khối dữ liệu con tương ứng.

Ví dụ 18: Phương pháp theo ví dụ 16, phương pháp này còn bao gồm các bước: xác định nhiều chuỗi bin theo: một trong số thứ tự mật độ tăng hoặc thứ tự mật độ giảm; và dịch thời gian thứ tự xử lý.

Ví dụ 19: Phương pháp theo bất kỳ một trong các ví dụ 14-18, phương pháp này còn bao gồm bước: cung cấp dữ liệu được xử lý cho bộ yêu cầu xử lý dữ liệu.

Ví dụ 20: Phương pháp theo bất kỳ một trong các ví dụ 14-19, trong đó: dữ liệu để xử lý song song là dữ liệu đầu vào mô hình học máy cho mô hình mạng nơron tích chập, mỗi phần dữ liệu trong số nhiều phần dữ liệu bao gồm bản đồ đặc trưng, và phương pháp này còn bao gồm các bước: tạo chuỗi lại nhân tích chập cho mỗi khối dữ liệu con tương ứng trong số nhiều khối dữ liệu con dựa trên chuỗi các phần dữ liệu trong khối dữ liệu con tương ứng; và áp dụng nhân tích chập được tạo chuỗi lại cho khối dữ liệu con tương ứng.

Ví dụ 21: Phương pháp theo bất kỳ một trong các ví dụ 14-20, trong đó việc xác định nhiều phần dữ liệu của dữ liệu dựa trên các phạm vi mật độ được thực hiện bởi bộ điều khiển truy cập bộ nhớ trực tiếp (DMA) truyền thông dữ liệu với nhiều khối xử lý.

Ví dụ 22: Phương pháp theo ví dụ 21, trong đó việc tạo nhiều khối dữ liệu con có các phần dữ liệu được tạo chuỗi lại được thực hiện bởi bộ điều khiển DMA.

Ví dụ 23: Phương pháp theo bất kỳ của các ví dụ 14-20, trong đó nhiều khối xử lý bao gồm ít nhất một khối xử lý tín hiệu nơron.

Ví dụ 24: Hệ thống xử lý để thực hiện xử lý dữ liệu song song, hệ thống này bao gồm: bộ nhớ bao gồm các lệnh thực thi được bởi máy tính; một hoặc nhiều bộ xử lý được tạo cấu hình để thực thi các lệnh thực thi được bởi máy tính và khiến cho hệ thống xử lý thực hiện phương pháp theo ví dụ bất kỳ trong số các ví dụ 14-23.

Ví dụ 25: Phương tiện bất biến đọc được bằng máy tính bao gồm lệnh mà, khi được thực thi bởi một hoặc nhiều bộ xử lý của hệ thống xử lý, khiến cho hệ thống xử lý thực hiện phương pháp theo ví dụ bất kỳ trong số các ví dụ 14-23.

Ví dụ 26: Sản phẩm chương trình máy tính nằm trên phương tiện lưu trữ có thể đọc được bằng máy tính và bao gồm mã để thực hiện phương pháp theo ví dụ bất kỳ trong số các ví dụ 14-23.

Các xem xét bổ sung

Mô tả trước được cung cấp để cho phép bất kỳ người nào có hiểu biết trung bình trong lĩnh vực này thực hành các ví dụ khác nhau và các ví dụ được mô tả ở đây. Các ví dụ được mô tả ở đây không làm giới hạn phạm vi, khả năng áp dụng hoặc các ví dụ được nêu trong các yêu cầu bảo hộ. Những sửa đổi khác nhau đối với các ví dụ này sẽ dễ dàng nhận thấy đối với những người có hiểu biết trung bình trong lĩnh vực này và các nguyên tắc chung được xác định ở đây có thể được áp dụng cho các ví dụ khác. Ví dụ, các thay đổi có thể được thực hiện trong chức năng và cách sắp xếp của các yếu tố được mô tả mà không nằm ngoài phạm vi của sáng chế. Các ví dụ khác nhau có thể bỏ qua, thay thế hoặc thêm các thủ tục hoặc thành phần khác nhau nếu thích hợp. Ví dụ, các phương pháp được mô tả có thể được thực hiện theo thứ tự khác với thứ tự được mô tả và các bước khác nhau có thể được thêm vào, bỏ qua hoặc kết hợp. Ngoài ra, các tính năng được mô tả liên quan đến một số ví dụ có thể được kết hợp trong một số ví dụ khác. Ví dụ, thiết bị có thể được triển khai hoặc phương pháp có thể được thực hiện bằng cách sử dụng số lượng bất kỳ trong số các khía cạnh được nêu ở đây. Ngoài ra, phạm vi của sáng chế nhằm mục đích đề cập đến một thiết bị hoặc phương pháp được thực hành sử dụng cấu trúc, chức năng hoặc cấu trúc và chức năng khác ngoài, hoặc ngoài các khía cạnh khác nhau của sáng chế được nêu ở đây. Cần hiểu rằng bất kỳ khía cạnh nào của sáng chế được tiết lộ ở đây có thể được thể hiện bởi một hoặc nhiều yếu tố của một yêu cầu bảo hộ.

Như được sử dụng ở đây, từ "làm mẫu" có nghĩa là "làm ví dụ, ví dụ hoặc minh họa." Bất kỳ khía cạnh nào được mô tả ở đây là "làm mẫu" không nhất thiết phải được hiểu là ưu tiên hoặc có lợi hơn các khía cạnh khác.

Như được sử dụng ở đây, cụm từ đề cập đến “ít nhất một trong số” danh sách các mục đề cập đến bất kỳ sự kết hợp nào của các mục đó, bao gồm cả các thành viên đơn lẻ. Ví dụ, “ít nhất một trong số: a, b hoặc c” nhằm bao hàm a, b, c, a-b, a-c, b-c và a-b-c, cũng như bất kỳ kết hợp nào với bội số của cùng một phần tử (ví dụ: a-a, a-a-a, a-a-b, a-a-c, a-b-b, a -c-c, b-b, b-b-b, b-b-c, c-c và c-c-c hoặc bất kỳ thứ tự nào khác của a, b và c).

Như được sử dụng ở đây, thuật ngữ “xác định” bao gồm nhiều loại hành động. Ví dụ, “xác định” có thể bao gồm tính, tính toán, xử lý, suy ra, điều tra, tra cứu (ví dụ, tra cứu trong bảng, cơ sở dữ liệu hoặc cấu trúc dữ liệu khác), xác định và tương tự. Ngoài ra, “xác định” có thể bao gồm nhận (ví dụ, nhận thông tin), truy cập (ví dụ, truy cập dữ liệu trong bộ nhớ) và tương tự. Ngoài ra, “xác định” có thể bao gồm giải quyết, chọn, lựa chọn, thiết lập và tương tự.

Các phương pháp bộc lộ ở đây bao gồm một hoặc nhiều bước hoặc hành động để đạt được các phương pháp. Các bước phương pháp và/hoặc các hành động có thể được thay thế cho nhau mà không nằm ngoài phạm vi của các điểm yêu cầu bảo hộ. Nói cách khác, trừ khi một thứ tự cụ thể của các bước hoặc hành động được chỉ định, thứ tự và/hoặc việc sử dụng các bước và/hoặc hành động cụ thể có thể được sửa đổi mà không nằm ngoài phạm vi của các yêu cầu bảo hộ. Hơn nữa, các hoạt động khác nhau của các phương pháp được mô tả ở trên có thể được thực hiện bằng bất kỳ phương tiện thích hợp nào có khả năng thực hiện các chức năng tương ứng. Các phương tiện có thể bao gồm (các) thành phần phần cứng và/hoặc phần mềm và/hoặc (các) mô đun khác nhau, bao gồm, nhưng không giới hạn ở mạch, mạch tích hợp chuyên dụng (ASIC) hoặc bộ xử lý. Nói chung, nếu có các hoạt động được thể hiện trên các hình vẽ, thì các hoạt động này có thể có các thành phần phương tiện và chức năng tương đương với cách đánh số tương tự.

Các yêu cầu bảo hộ sau đây không nhằm giới hạn trong các ví dụ được trình bày ở đây, nhưng được cung cấp phạm vi đầy đủ phù hợp với ngôn ngữ của các yêu cầu bảo hộ. Trong yêu cầu bảo hộ, việc tham chiếu đến một phần tử ở dạng số ít không dự định có nghĩa là “một và chỉ một” trừ khi được quy định cụ thể như vậy, mà có nghĩa là “một hoặc nhiều”. Trừ khi có quy định cụ thể khác, thuật ngữ “một số” đề cập

đến một hoặc nhiều. Không thành phần yêu cầu bảo hộ được hiểu theo các quy định trong 35 U.S.C. §112(f), trừ phi thành phần đó được nêu ra rõ ràng bằng cách sử dụng cụm từ "phương tiện để", hoặc trong trường hợp điểm yêu cầu bảo hộ phương pháp, thành phần này được nêu ra bằng cách sử dụng cụm từ "bước để". Tất cả các điểm tương đương về cấu trúc và chức năng đối với các phần tử của các khía cạnh khác nhau được mô tả trong sáng chế này đã được biết đến hoặc sau này được biết đến với những người có hiểu biết thông thường trong lĩnh vực kỹ thuật này được đưa vào đây một cách rõ ràng bằng cách tham khảo và được bao hàm bởi các yêu cầu bảo hộ. Hơn nữa, không có gì được tiết lộ ở đây nhằm mục đích dành riêng cho công chúng bất kể việc tiết lộ đó có được nêu rõ ràng trong các yêu cầu bảo hộ hay không.

YÊU CẦU BẢO HỘ

1. Phương pháp thực hiện xử lý dữ liệu song song, phương pháp này bao gồm các bước:

nhận dữ liệu để xử lý song song từ bộ yêu cầu xử lý dữ liệu;

tạo nhiều khối dữ liệu con dựa trên dữ liệu;

xác định nhiều phần dữ liệu trong mỗi khối dữ liệu con trong số nhiều khối dữ liệu con;

thay đổi thứ tự của nhiều phần dữ liệu trong ít nhất một khối dữ liệu con trong số nhiều khối dữ liệu con;

cung cấp nhiều khối dữ liệu con, gồm ít nhất một khối dữ liệu con bao gồm thứ tự thay đổi của nhiều phần dữ liệu, cho nhiều khối xử lý để xử lý song song; và

nhận dữ liệu được xử lý được liên kết với nhiều khối dữ liệu con từ nhiều khối xử lý.

2. Phương pháp theo điểm 1, trong đó việc xác định nhiều phần dữ liệu trong mỗi khối dữ liệu con trong số nhiều khối dữ liệu con bao gồm các bước:

xác định nhiều phạm vi mật độ, trong đó mỗi phạm vi mật độ của nhiều phạm vi mật độ là khác biệt với từng phạm vi mật độ khác của nhiều phạm vi mật độ; và

xác định nhiều phần dữ liệu dựa trên nhiều phạm vi mật độ.

3. Phương pháp theo điểm 1, trong đó việc thay đổi thứ tự của nhiều phần dữ liệu cho ít nhất một khối dữ liệu con trong số nhiều khối dữ liệu con bao gồm bước: dịch theo thời gian mỗi phần dữ liệu trong số nhiều phần dữ liệu theo hướng thời gian xử lý theo thứ tự vòng tròn.

4. Phương pháp theo điểm 1, trong đó việc tạo chuỗi lại thứ tự của nhiều phần dữ liệu cho ít nhất một khối dữ liệu con trong số nhiều khối dữ liệu con bao gồm bước: sắp xếp nhiều phần dữ liệu theo một trong số thứ tự mật độ tăng hoặc thứ tự mật độ giảm.

5. Phương pháp theo điểm 1, phương pháp này còn bao gồm các bước:

thay đổi thứ tự của dữ liệu được xử lý được liên kết với nhiều khối dữ liệu con dựa trên thứ tự gốc của nhiều phần dữ liệu trong mỗi khối dữ liệu con trong số nhiều khối dữ liệu con; và

cung cấp dữ liệu được xử lý cho bộ yêu cầu xử lý dữ liệu.

6. Phương pháp theo điểm 1, phương pháp này còn bao gồm bước: thay đổi thứ tự của nhiều phần dữ liệu trong mỗi khối dữ liệu con trong số nhiều khối dữ liệu con.

7. Phương pháp theo điểm 1, trong đó việc xác định nhiều phần dữ liệu trong mỗi khối dữ liệu con trong số nhiều khối dữ liệu con được thực hiện bởi bộ điều khiển truy cập bộ nhớ trực tiếp (Direct Memory Access - DMA) truyền thông dữ liệu với nhiều khối xử lý.

8. Phương pháp theo điểm 7, trong đó việc thay đổi thứ tự của nhiều phần dữ liệu trong ít nhất một khối dữ liệu con trong số nhiều khối dữ liệu con được thực hiện bởi bộ điều khiển DMA.

9. Phương pháp theo điểm 1, trong đó việc thay đổi nhiều phần dữ liệu trong ít nhất một khối dữ liệu con trong số nhiều khối dữ liệu con bao gồm bước đọc, bởi nhiều khối xử lý, nhiều phần dữ liệu theo thứ tự định trước.

10. Phương pháp theo điểm 1, trong đó nhiều khối xử lý bao gồm ít nhất một khối xử lý tín hiệu neuron.

11. Hệ thống xử lý để thực hiện xử lý dữ liệu song song, hệ thống bao gồm:

bộ nhớ bao gồm các lệnh thực thi được bởi máy tính;

một hoặc nhiều bộ xử lý được tạo cấu hình để thực thi các lệnh thực thi được bởi máy tính và khiến cho hệ thống xử lý:

nhận dữ liệu để xử lý song song từ bộ yêu cầu xử lý dữ liệu;

tạo nhiều khối dữ liệu con dựa trên dữ liệu;

xác định nhiều phần dữ liệu trong mỗi khối dữ liệu con trong số nhiều khối dữ liệu con;

thay đổi thứ tự của nhiều phần dữ liệu trong ít nhất một khối dữ liệu con trong số nhiều khối dữ liệu con;

cung cấp nhiều khối dữ liệu con, gồm ít nhất một khối dữ liệu con bao gồm thứ tự thay đổi của nhiều phần dữ liệu, cho nhiều khối xử lý để xử lý song song; và

nhận dữ liệu được xử lý được liên kết với nhiều khối dữ liệu con từ nhiều khối xử lý.

12. Hệ thống xử lý theo điểm 11, trong đó để xác định nhiều phần dữ liệu trong mỗi khối dữ liệu con trong số nhiều khối dữ liệu con, một hoặc nhiều bộ xử lý còn được tạo cấu hình để:

xác định nhiều phạm vi mật độ, trong đó mỗi phạm vi mật độ của nhiều phạm vi mật độ khác biệt với từng phạm vi mật độ khác của nhiều phạm vi mật độ; và

xác định nhiều phần dữ liệu dựa trên nhiều phạm vi mật độ.

13. Hệ thống xử lý theo điểm 11, trong đó để thay đổi thứ tự của nhiều phần dữ liệu cho ít nhất một khối dữ liệu con trong số nhiều khối dữ liệu con, một hoặc nhiều bộ xử lý còn được tạo cấu hình để dịch theo thời gian mỗi phần dữ liệu trong số nhiều phần dữ liệu theo hướng thời gian xử lý theo thứ tự vòng tròn.

14. Hệ thống xử lý theo điểm 11, trong đó để thay đổi thứ tự của nhiều phần dữ liệu cho ít nhất một khối dữ liệu con trong số nhiều khối dữ liệu con, một hoặc nhiều bộ xử lý còn được tạo cấu hình để sắp xếp nhiều phần dữ liệu theo một trong số thứ tự mật độ tăng hoặc thứ tự mật độ giảm.

15. Hệ thống xử lý theo điểm 11, trong đó một hoặc nhiều bộ xử lý còn được tạo cấu hình để:

thay đổi thứ tự của dữ liệu được xử lý được liên kết với nhiều khối dữ liệu con dựa trên thứ tự gốc của nhiều phần dữ liệu trong mỗi khối dữ liệu con trong số nhiều khối dữ liệu con;

cung cấp dữ liệu được xử lý cho bộ yêu cầu xử lý dữ liệu.

16. Hệ thống xử lý theo điểm 11, trong đó một hoặc nhiều bộ xử lý còn được tạo cấu hình để thay đổi thứ tự của nhiều phần dữ liệu trong mỗi khối dữ liệu con trong số nhiều khối dữ liệu con.

17. Hệ thống xử lý theo điểm 11, hệ thống này còn bao gồm: bộ điều khiển truy cập bộ nhớ trực tiếp (DMA) truyền thông dữ liệu với nhiều khối xử lý được tạo cấu hình để xác định nhiều phần dữ liệu trong mỗi khối dữ liệu con trong số nhiều khối dữ liệu con được thực hiện.

18. Hệ thống xử lý theo điểm 17, trong đó bộ điều khiển DMA còn được tạo cấu hình để thay đổi thứ tự của nhiều phần dữ liệu trong ít nhất một khối dữ liệu con trong số nhiều khối dữ liệu con.

19. Hệ thống xử lý theo điểm 11, trong đó để thay đổi thứ tự của nhiều phần dữ liệu trong ít nhất một khối dữ liệu con trong số nhiều khối dữ liệu con, một hoặc nhiều bộ xử lý được tạo cấu hình để đọc nhiều phần dữ liệu theo thứ tự định trước.

20. Hệ thống xử lý theo điểm 11, trong đó nhiều khối xử lý bao gồm ít nhất một khối xử lý tín hiệu nơron.

21. Phương pháp thực hiện xử lý dữ liệu song song, phương pháp này bao gồm các bước:

nhận dữ liệu để xử lý song song từ bộ yêu cầu xử lý dữ liệu;

xác định nhiều phạm vi mật độ, trong đó mỗi phạm vi mật độ của nhiều phạm vi mật độ khác biệt với từng phạm vi mật độ khác của nhiều phạm vi mật độ;

xác định nhiều phần dữ liệu của dữ liệu dựa trên các phạm vi mật độ;

tạo nhiều khối dữ liệu con có các phần dữ liệu được tạo chuỗi lại;

cung cấp nhiều khối dữ liệu con cho nhiều khối xử lý để xử lý song song; và

nhận dữ liệu được xử lý được liên kết với nhiều khối dữ liệu con từ nhiều khối xử lý.

22. Phương pháp theo điểm 21, trong đó bước xác định nhiều phần dữ liệu của dữ liệu dựa trên các phạm vi mật độ bao gồm các bước:

xác định mật độ cho mỗi phần dữ liệu tương ứng trong số nhiều phần dữ liệu;
và

gán mỗi phần dữ liệu tương ứng trong số nhiều phần dữ liệu cho một bin mật độ trong số nhiều bin mật độ được dựa trên mật độ được xác định cho phần dữ liệu tương ứng,

trong đó mỗi bin mật độ được liên kết với một phạm vi mật độ trong số nhiều phạm vi mật độ.

23. Phương pháp theo điểm 22, trong đó việc tạo nhiều khối dữ liệu con có các phần dữ liệu được tạo chuỗi lại bao gồm:

thêm các phần dữ liệu vào nhiều khối dữ liệu con theo nhiều chuỗi bin,

trong đó mỗi chuỗi bin trong nhiều chuỗi bin bao gồm chuỗi duy nhất của nhiều bin mật độ.

24. Phương pháp theo điểm 23, phương pháp này còn bao gồm các bước:

gán mỗi chuỗi bin trong nhiều chuỗi bin cho một hoặc nhiều khối xử lý trong số nhiều khối xử lý,

trong đó mỗi khối dữ liệu con tương ứng trong số nhiều khối dữ liệu con được cung cấp cho một khối xử lý trong số nhiều khối xử lý dựa trên chuỗi bin được gán của khối dữ liệu con tương ứng.

25. Phương pháp theo điểm 23, phương pháp này còn bao gồm bước: xác định nhiều chuỗi bin theo:

một trong thứ tự mật độ tăng hoặc thứ tự mật độ giảm; và

dịch thời gian thứ tự xử lý.

26. Phương pháp theo điểm 21, phương pháp này còn bao gồm bước: cung cấp dữ liệu được xử lý cho bộ yêu cầu xử lý dữ liệu.

27. Phương pháp theo điểm 21, trong đó:

dữ liệu để xử lý song song là dữ liệu đầu vào mô hình học máy cho mô hình mạng nơron tích chập,

mỗi phần dữ liệu trong số nhiều phần dữ liệu bao gồm bản đồ đặc trưng, và phương pháp này còn bao gồm các bước:

tạo chuỗi lại nhân tích chập cho mỗi khối dữ liệu con tương ứng trong số nhiều khối dữ liệu con dựa trên chuỗi các phần dữ liệu trong khối dữ liệu con tương ứng; và áp dụng nhân tích chập được tạo chuỗi lại cho khối dữ liệu con tương ứng.

28. Phương pháp theo điểm 21, trong đó bước xác định nhiều phần dữ liệu của dữ liệu dựa trên các phạm vi mật độ được thực hiện bởi bộ điều khiển truy cập bộ nhớ trực tiếp (DMA) truyền thông dữ liệu với nhiều khối xử lý.

29. Phương pháp theo điểm 28, trong đó bước tạo nhiều khối dữ liệu con có các phần dữ liệu được tạo chuỗi lại được thực hiện bởi bộ điều khiển DMA.

30. Phương pháp theo điểm 21, trong đó nhiều khối xử lý bao gồm ít nhất một khối xử lý tín hiệu nơron.

31. Hệ thống xử lý để thực hiện xử lý dữ liệu song song, hệ thống này bao gồm:

bộ nhớ bao gồm các lệnh thực thi được bởi máy tính;

một hoặc nhiều bộ xử lý được tạo cấu hình để thực thi các lệnh thực thi được bởi máy tính và khiến cho hệ thống xử lý:

nhận dữ liệu để xử lý song song từ bộ yêu cầu xử lý dữ liệu;

xác định nhiều phạm vi mật độ, trong đó mỗi phạm vi mật độ của nhiều phạm vi mật độ khác biệt với từng phạm vi mật độ khác của nhiều phạm vi mật độ;

xác định nhiều phần dữ liệu của dữ liệu dựa trên các phạm vi mật độ;

tạo nhiều khối dữ liệu con có các phần dữ liệu được tạo chuỗi lại;

cung cấp nhiều khối dữ liệu con cho nhiều khối xử lý để xử lý song song; và

nhận dữ liệu được xử lý được liên kết với nhiều khối dữ liệu con từ nhiều khối xử lý.

32. Hệ thống xử lý theo điểm 31, trong đó để xác định nhiều phần dữ liệu của dữ liệu dựa trên các phạm vi mật độ, một hoặc nhiều bộ xử lý còn được tạo cấu hình để:

xác định mật độ cho mỗi phần dữ liệu tương ứng trong số nhiều phần dữ liệu; và

gán mỗi phần dữ liệu tương ứng trong số nhiều phần dữ liệu cho một bin mật độ trong số nhiều bin mật độ được dựa trên mật độ được xác định cho phần dữ liệu tương ứng,

trong đó mỗi bin mật độ được liên kết với một phạm vi mật độ trong số nhiều phạm vi mật độ.

33. Hệ thống xử lý theo điểm 32, trong đó để tạo nhiều khối dữ liệu con có các phần dữ liệu được tạo chuỗi lại, một hoặc nhiều bộ xử lý còn được tạo cấu hình để:

thêm các phần dữ liệu vào nhiều khối dữ liệu con theo nhiều chuỗi bin,

trong đó mỗi chuỗi bin trong nhiều chuỗi bin bao gồm chuỗi duy nhất của nhiều bin mật độ.

34. Hệ thống xử lý theo điểm 33, trong đó một hoặc nhiều bộ xử lý còn được tạo cấu hình để:

gán mỗi chuỗi bin trong nhiều chuỗi bin cho một hoặc nhiều khối xử lý trong số nhiều khối xử lý,

trong đó mỗi khối dữ liệu con tương ứng trong số nhiều khối dữ liệu con được cung cấp cho một khối xử lý trong số nhiều khối xử lý dựa trên chuỗi bin được gán của khối dữ liệu con tương ứng.

35. Hệ thống xử lý theo điểm 33, trong đó một hoặc nhiều bộ xử lý còn được tạo cấu hình để xác định nhiều chuỗi bin theo:

một trong thứ tự mật độ tăng hoặc thứ tự mật độ giảm; và

dịch theo thời gian thứ tự xử lý.

36. Hệ thống xử lý theo điểm 31, trong đó một hoặc nhiều bộ xử lý còn được tạo cấu hình để cung cấp dữ liệu được xử lý cho bộ yêu cầu xử lý dữ liệu.

37. Hệ thống xử lý theo điểm 31, trong đó:

dữ liệu để xử lý song song là dữ liệu đầu vào mô hình học máy cho mô hình mạng nơron tích chập,

mỗi phân dữ liệu trong số nhiều phân dữ liệu bao gồm bản đồ đặc trưng, và một hoặc nhiều bộ xử lý còn được tạo cấu hình để:

tạo chuỗi lại nhân tích chập cho mỗi khối dữ liệu con tương ứng trong số nhiều khối dữ liệu con dựa trên chuỗi các phân dữ liệu trong khối dữ liệu con tương ứng; và

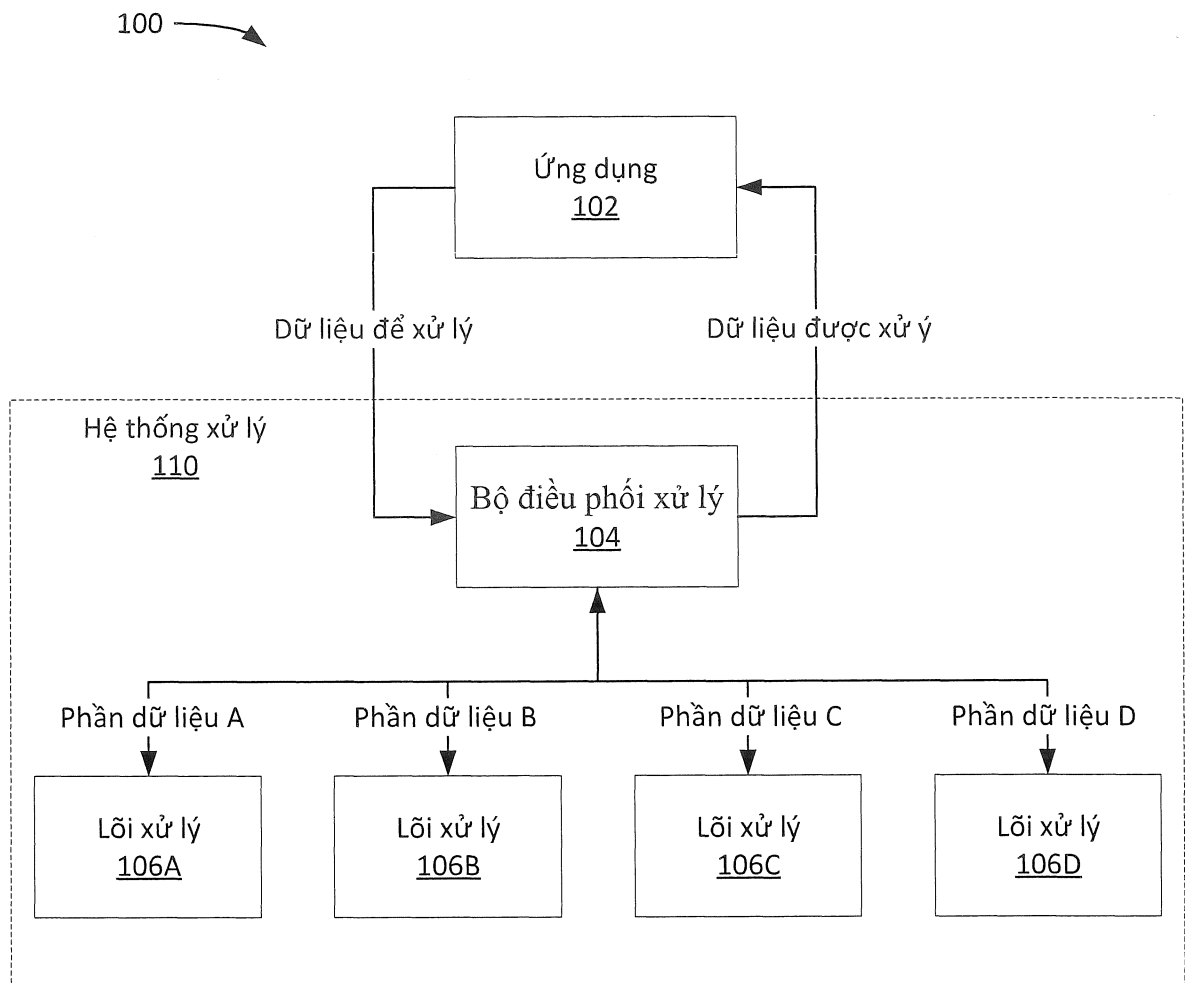
áp dụng nhân tích chập được tạo chuỗi lại cho khối dữ liệu con tương ứng.

38. Hệ thống xử lý theo điểm 31, còn bao gồm bộ điều khiển truy cập bộ nhớ trực tiếp (DMA) truyền thông dữ liệu với nhiều khối xử lý và được tạo cấu hình để xác định nhiều phân dữ liệu của dữ liệu dựa trên các phạm vi mật độ.

39. Hệ thống xử lý theo điểm 38, trong đó bộ điều khiển DMA còn được tạo cấu hình để tạo nhiều khối dữ liệu con có các phân dữ liệu được tạo chuỗi lại.

40. Hệ thống xử lý theo điểm 31, trong đó nhiều khối xử lý bao gồm ít nhất một khối xử lý tín hiệu nơron.

1/17

**FIG. 1**

2/17

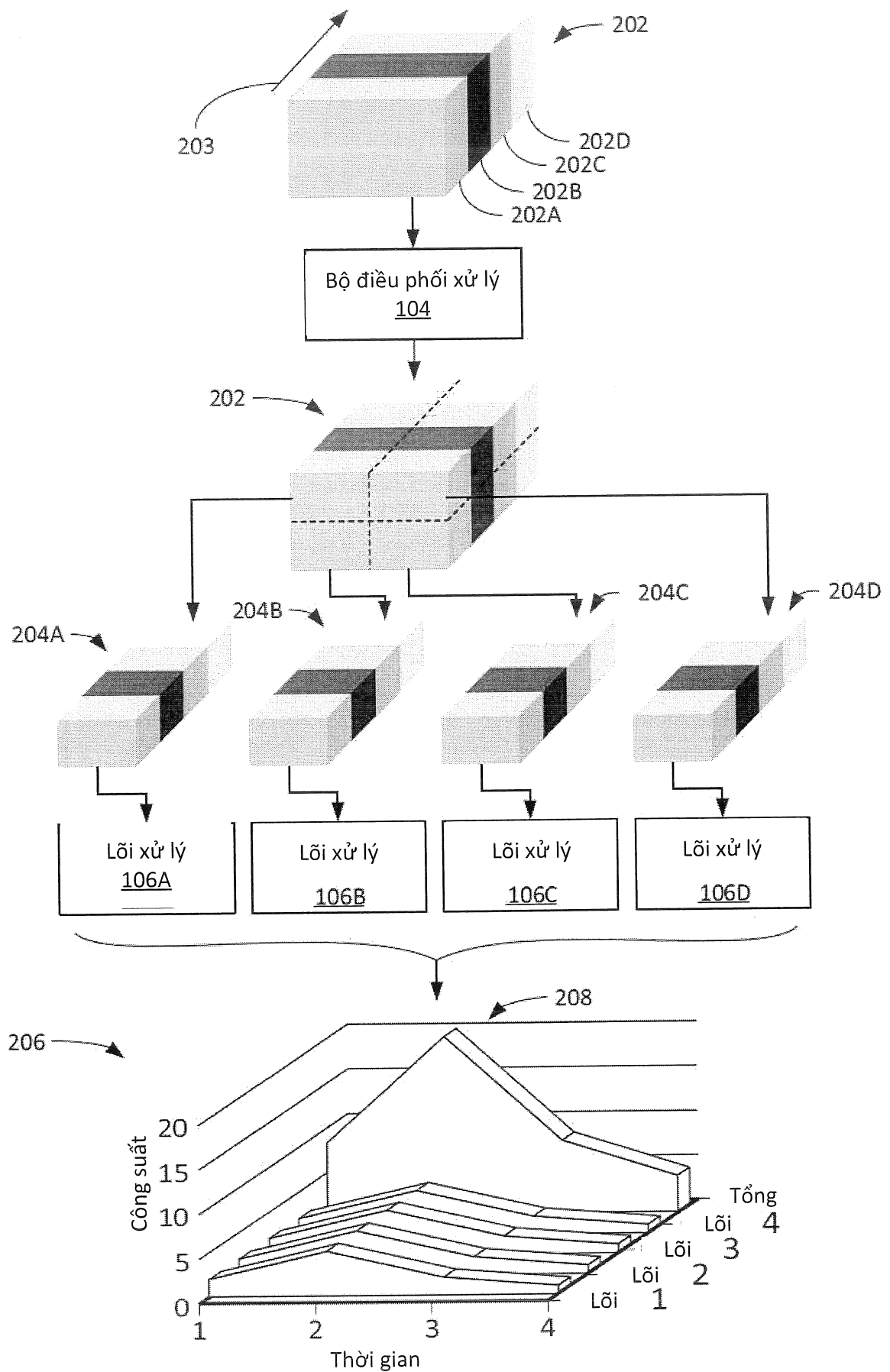


FIG. 2

3/17

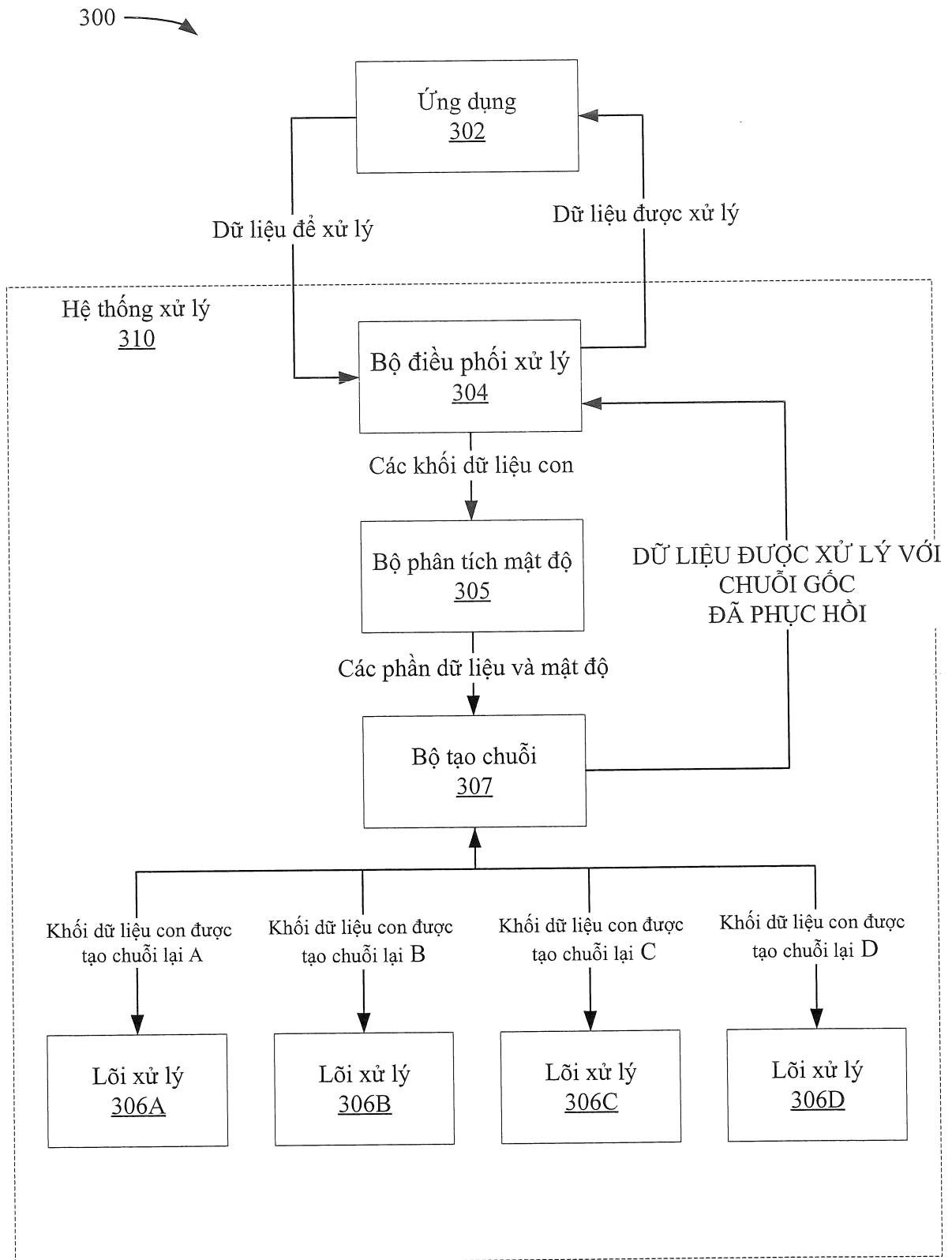


FIG. 3

4/17

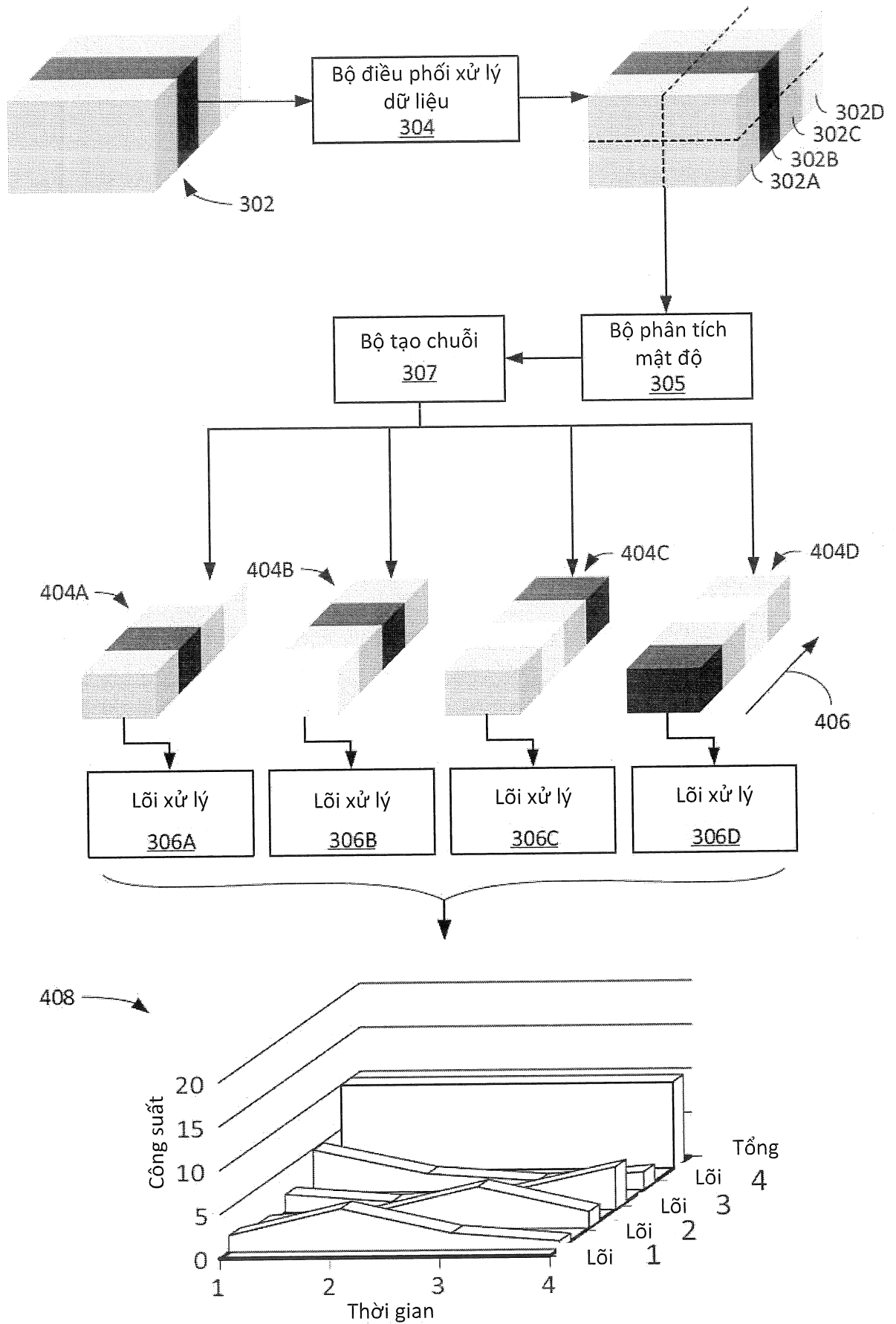
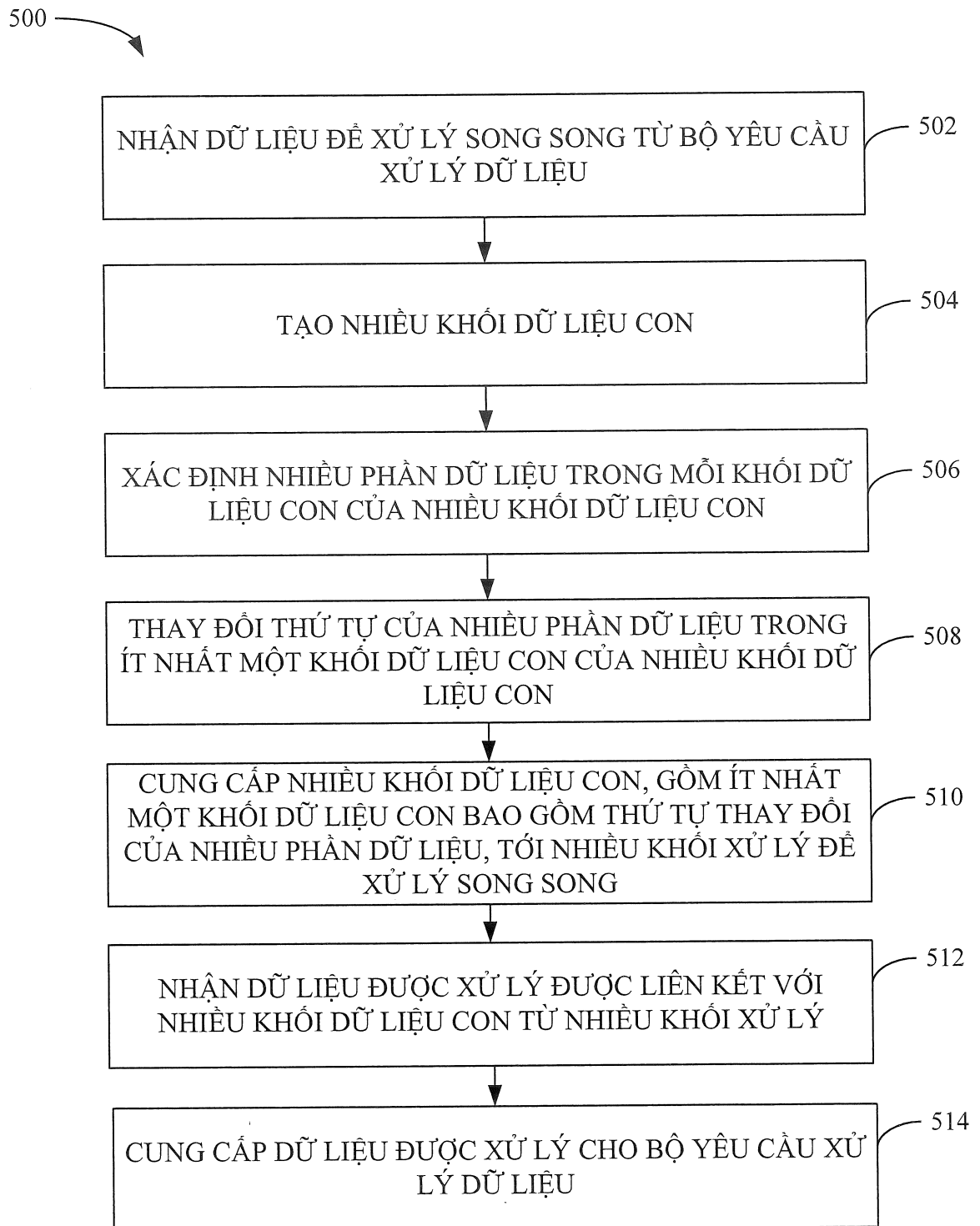


FIG. 4

5/17

**FIG. 5**

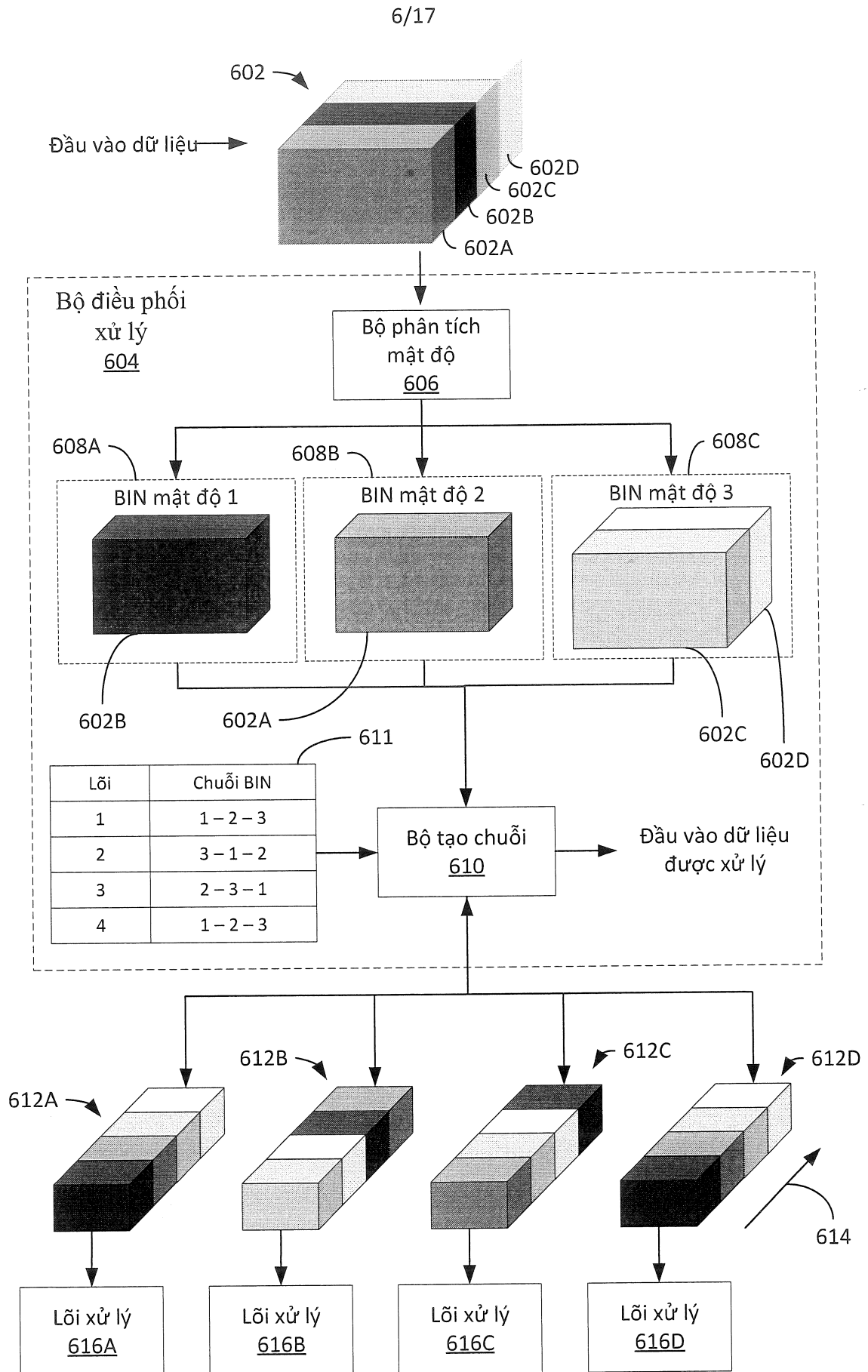
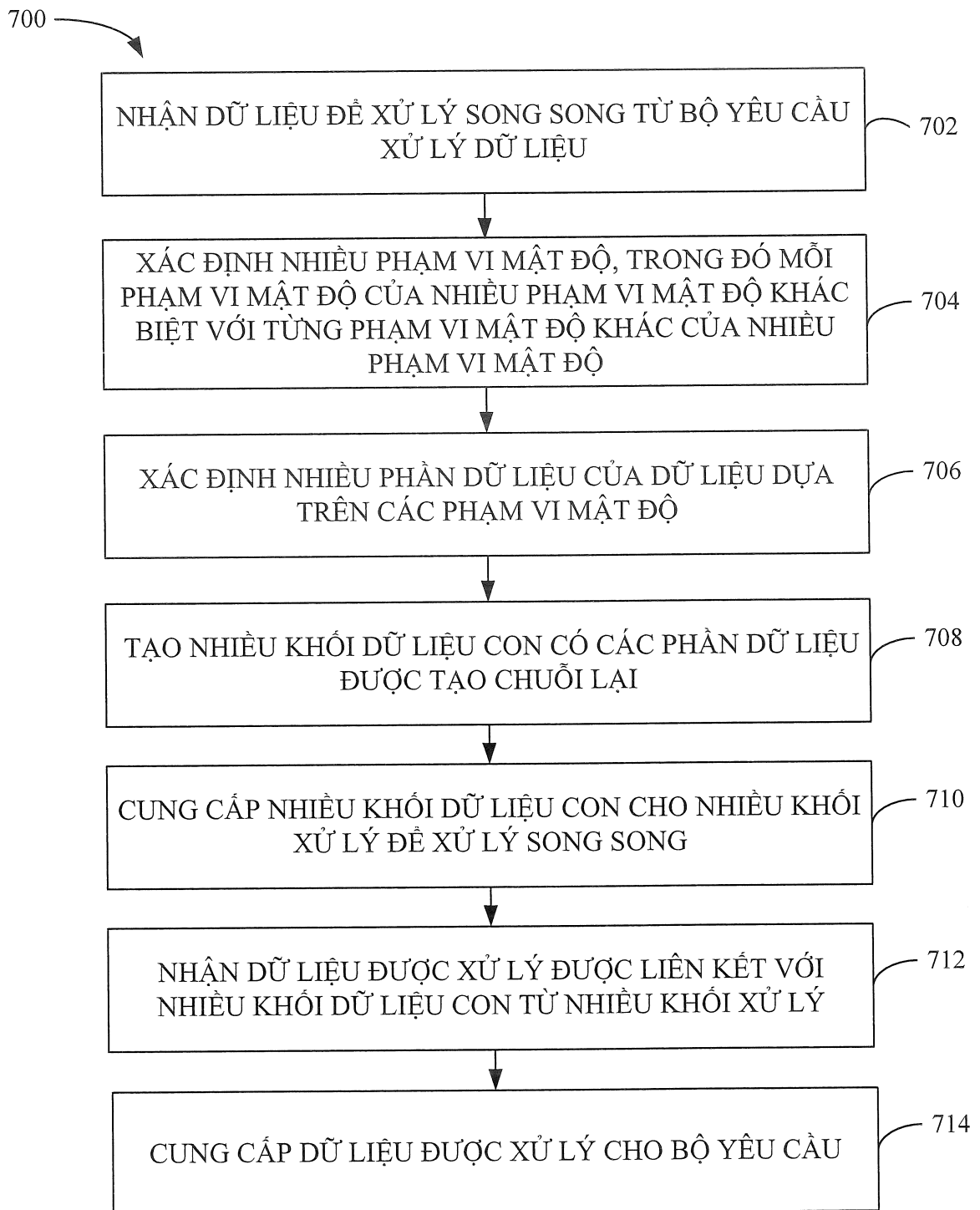


FIG. 6

7/17

**FIG. 7**

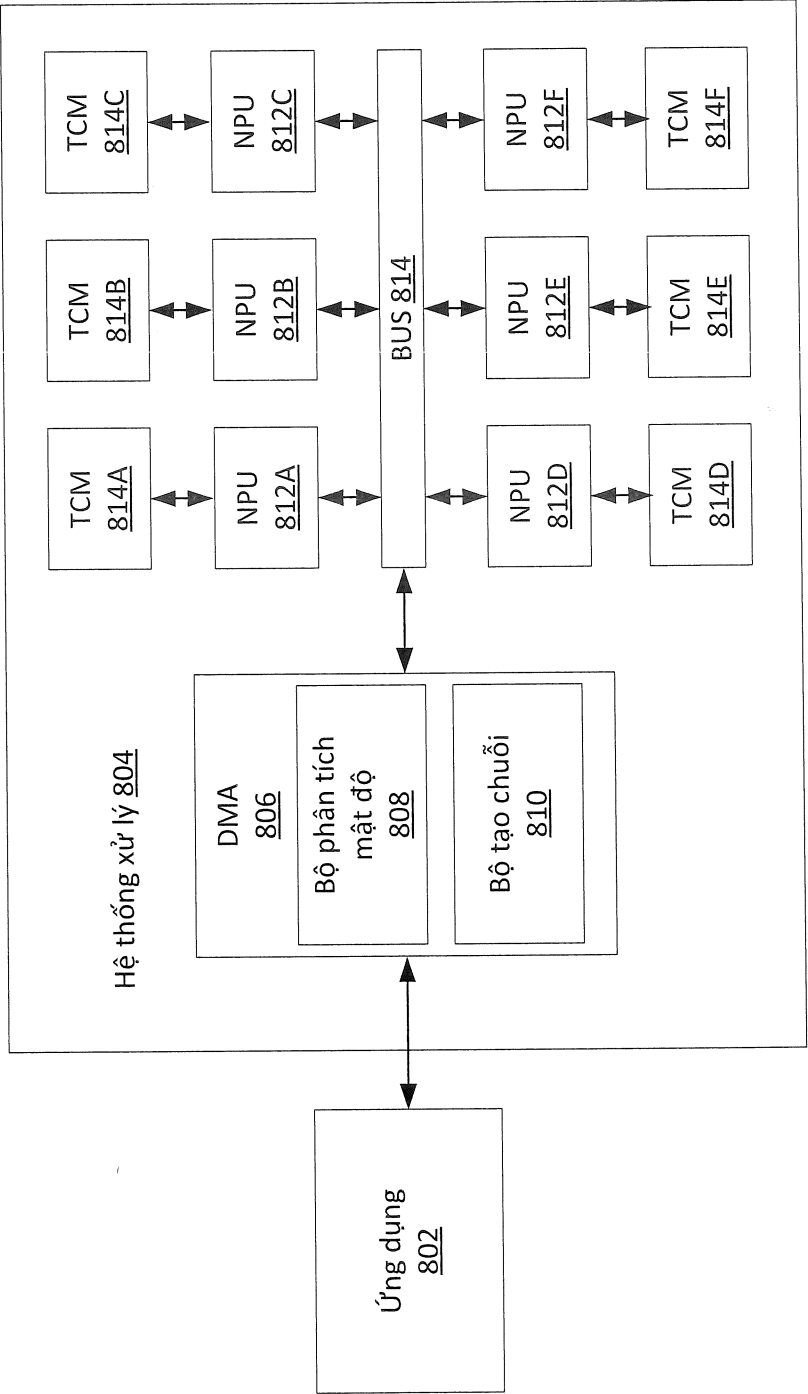
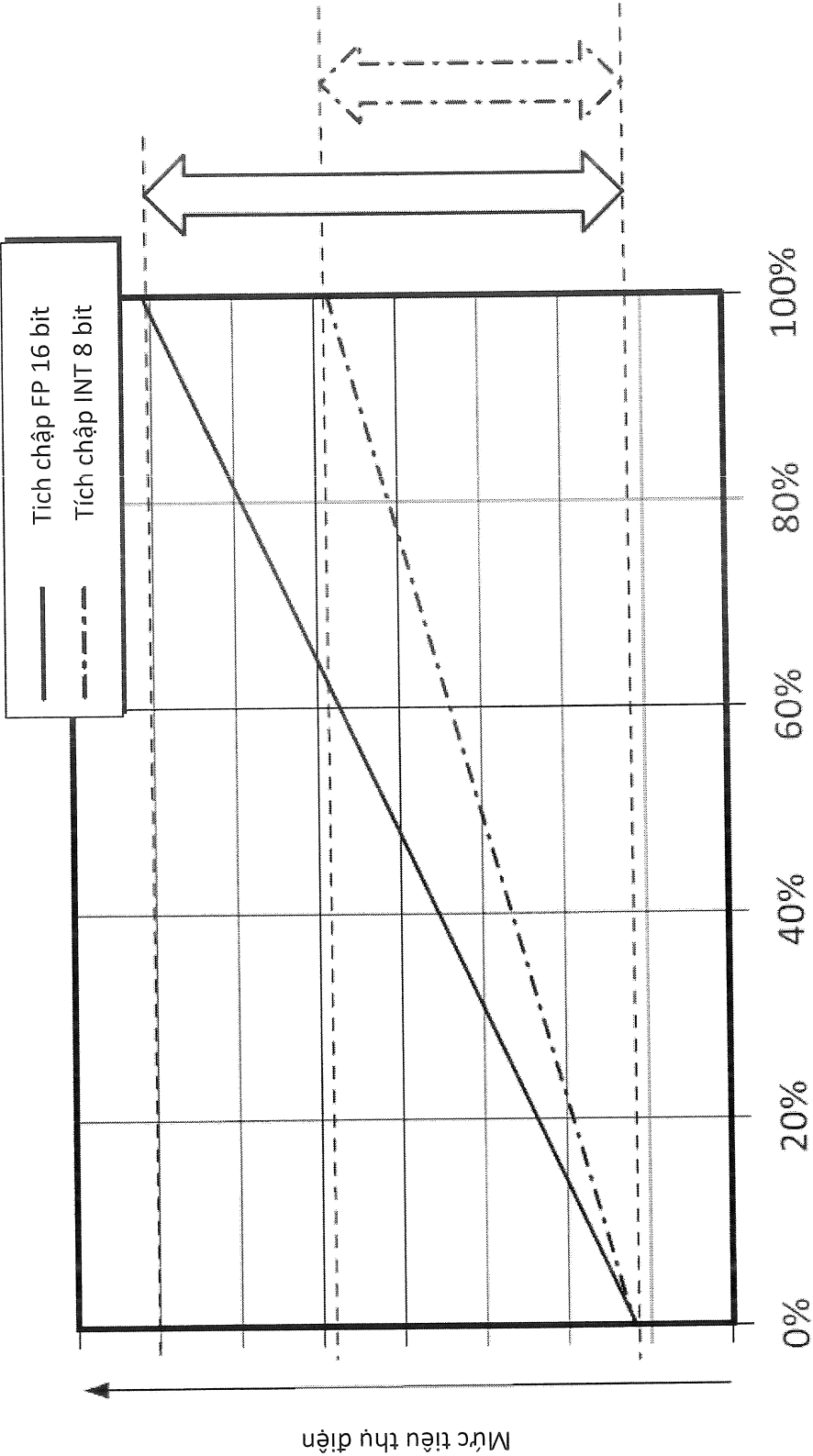


FIG. 8

900



Mật độ các giá trị trong hoạt động tích chấp

FIG. 9

10/17

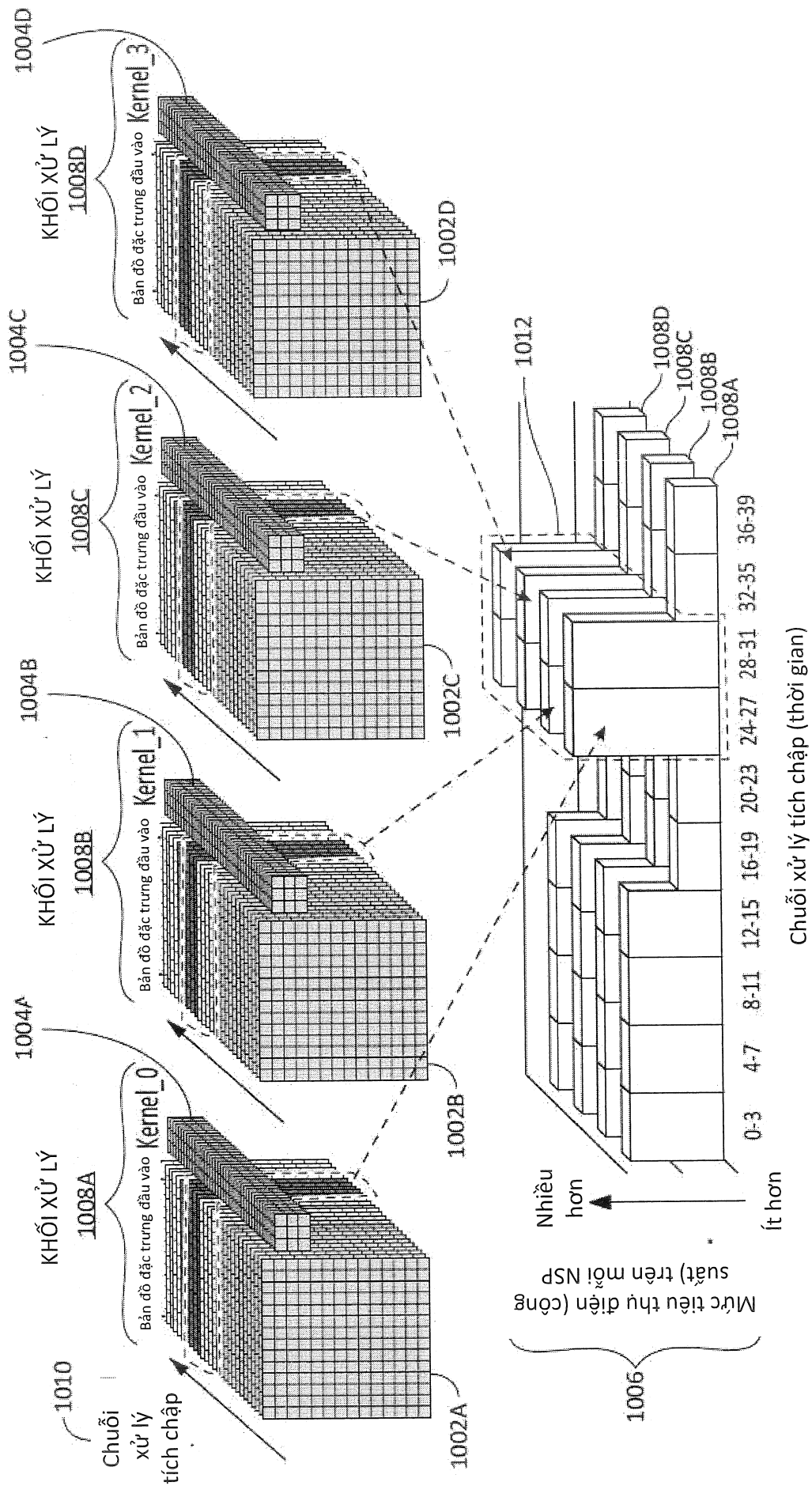


FIG. 10

11/17

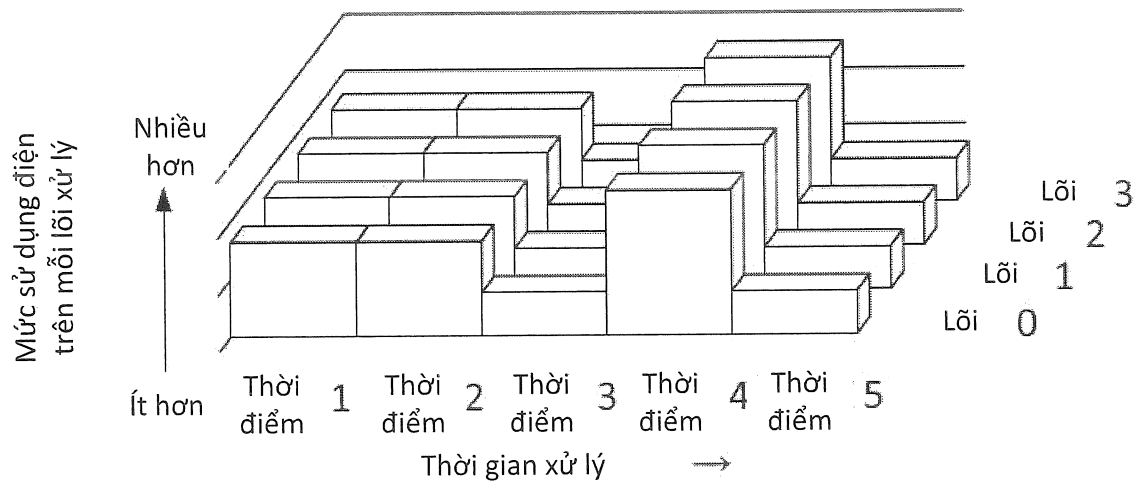


FIG. 11A

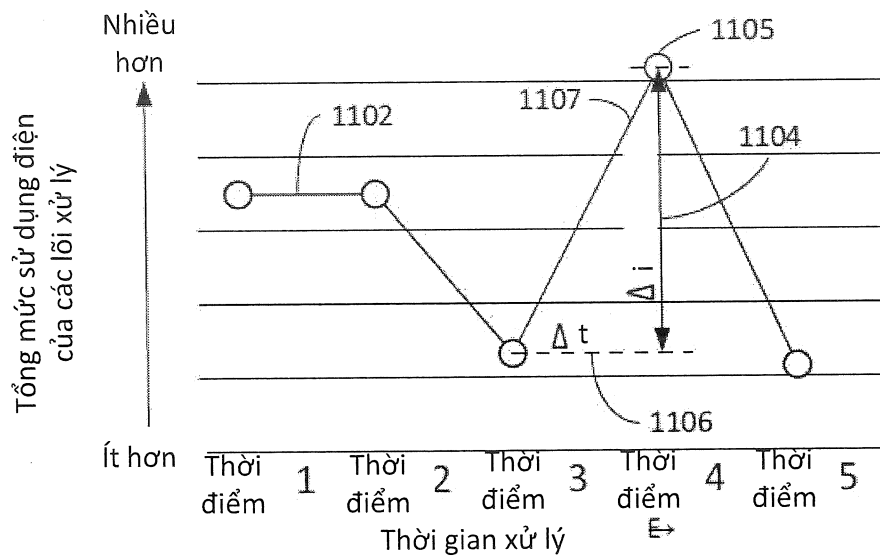


FIG. 11B

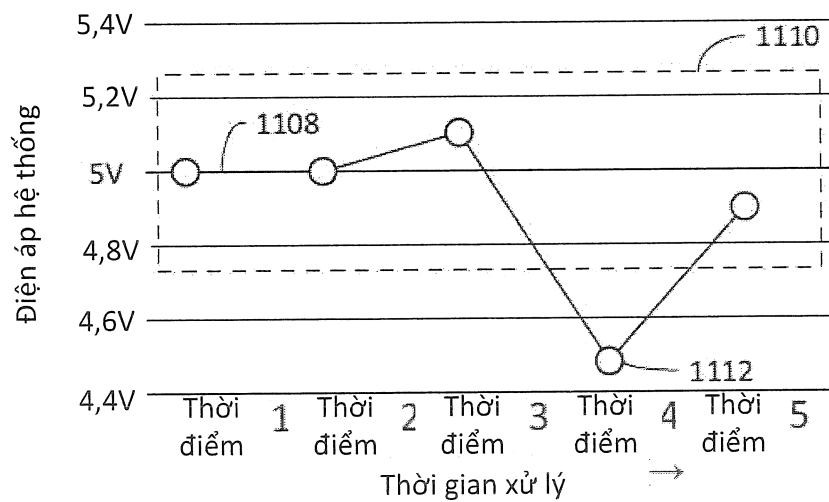
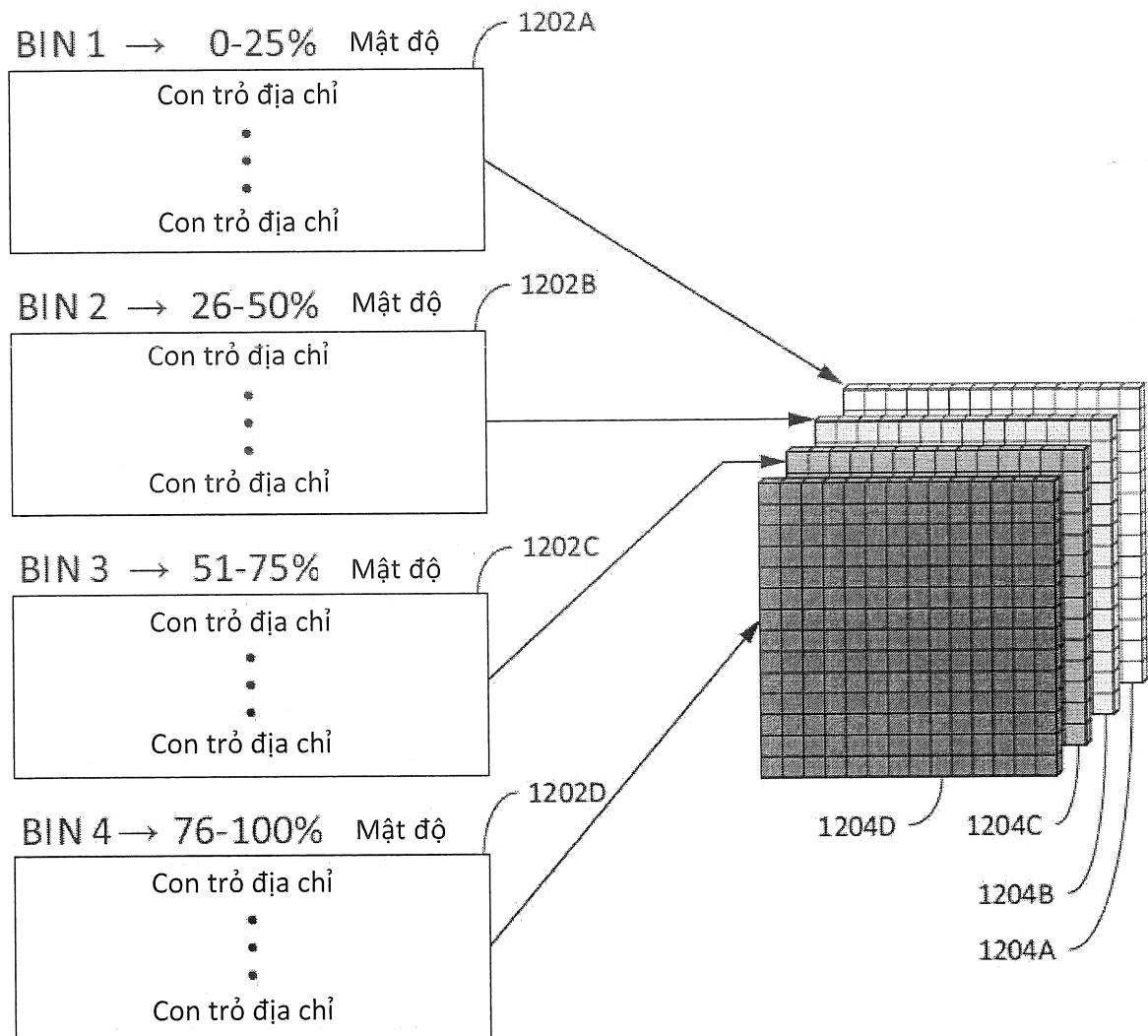
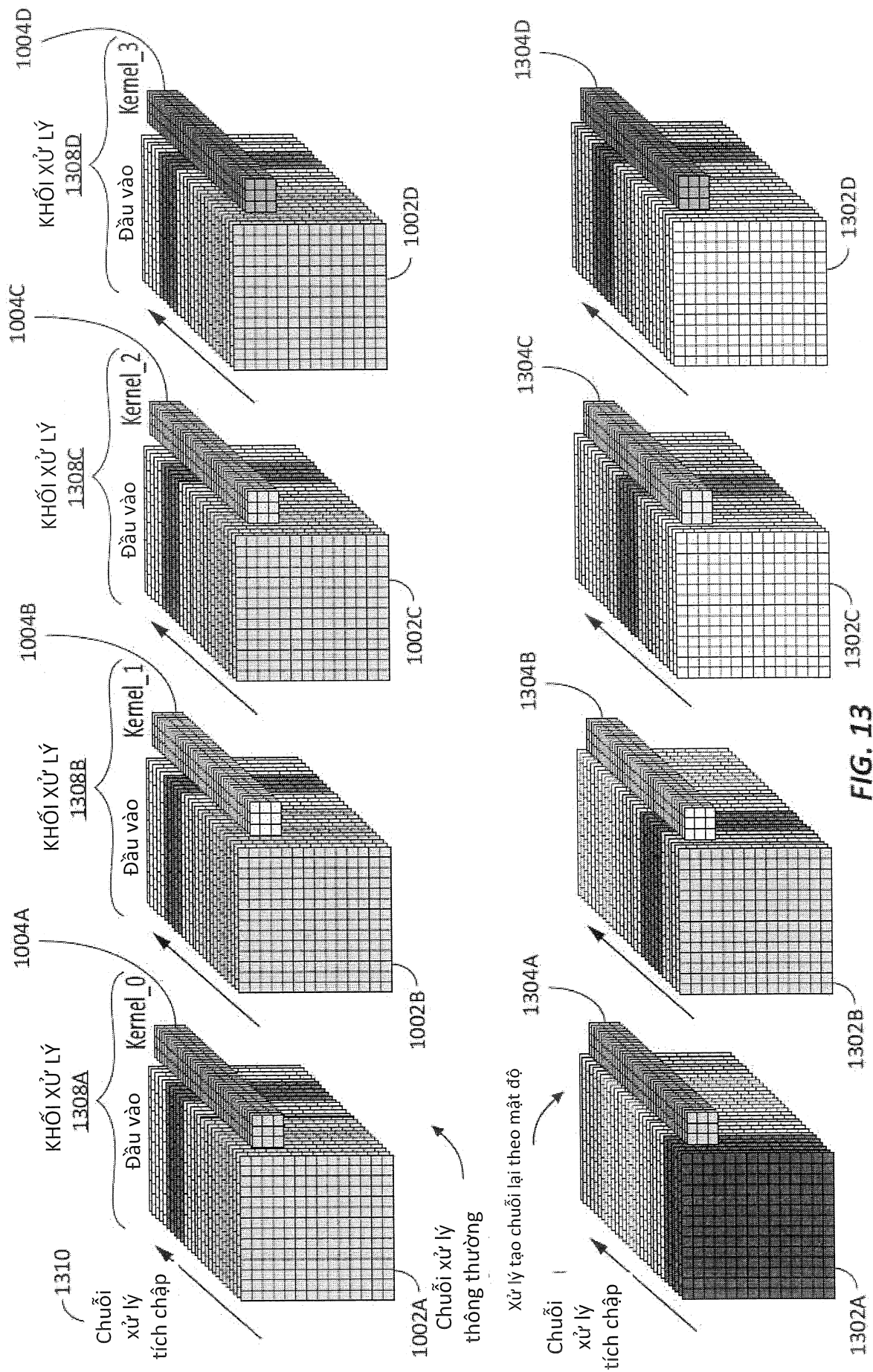


FIG. 11C

12/17

**FIG. 12**

13/17



14/17

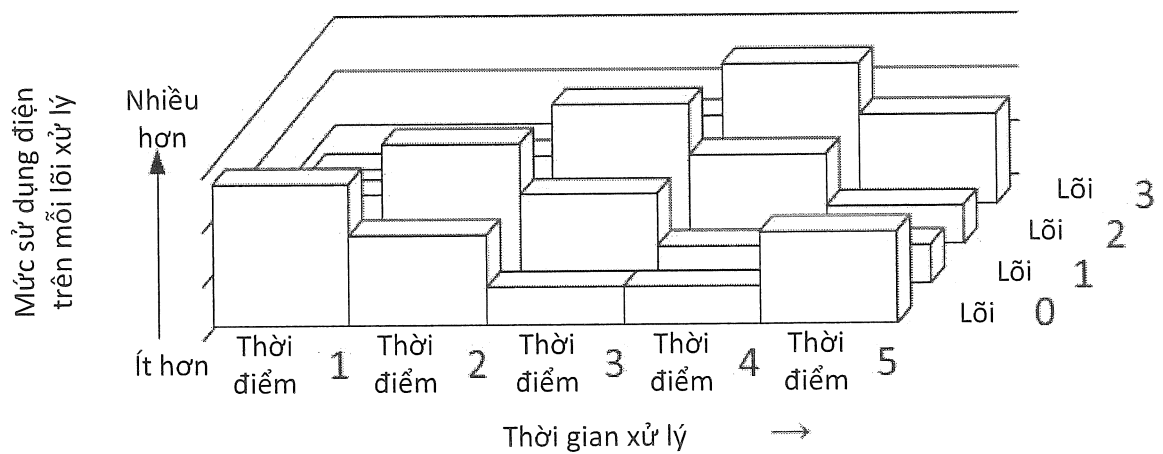


FIG. 14A

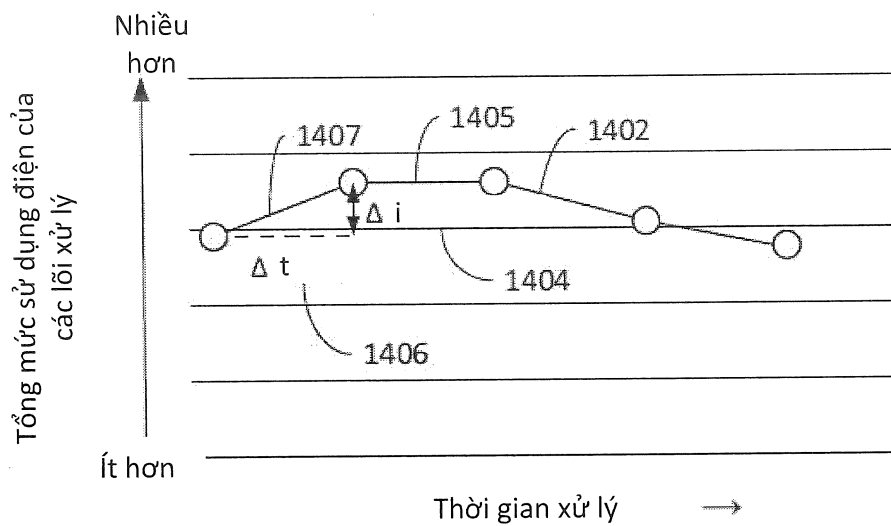


FIG. 14B

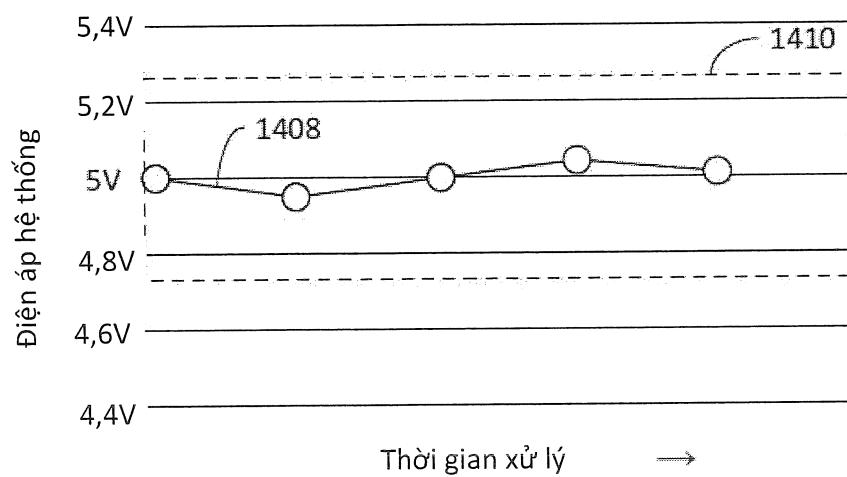


FIG. 14C

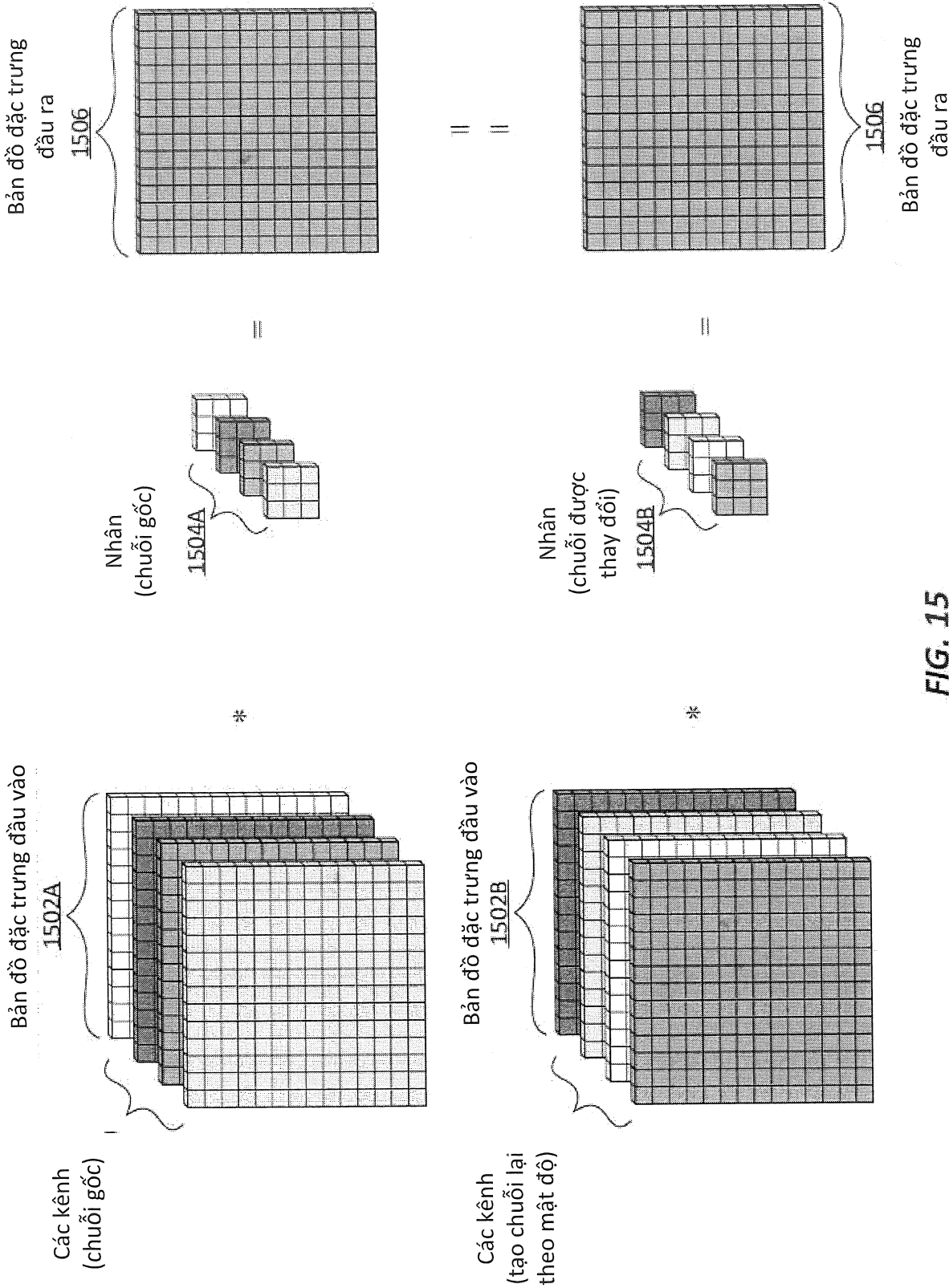


FIG. 15

16/17

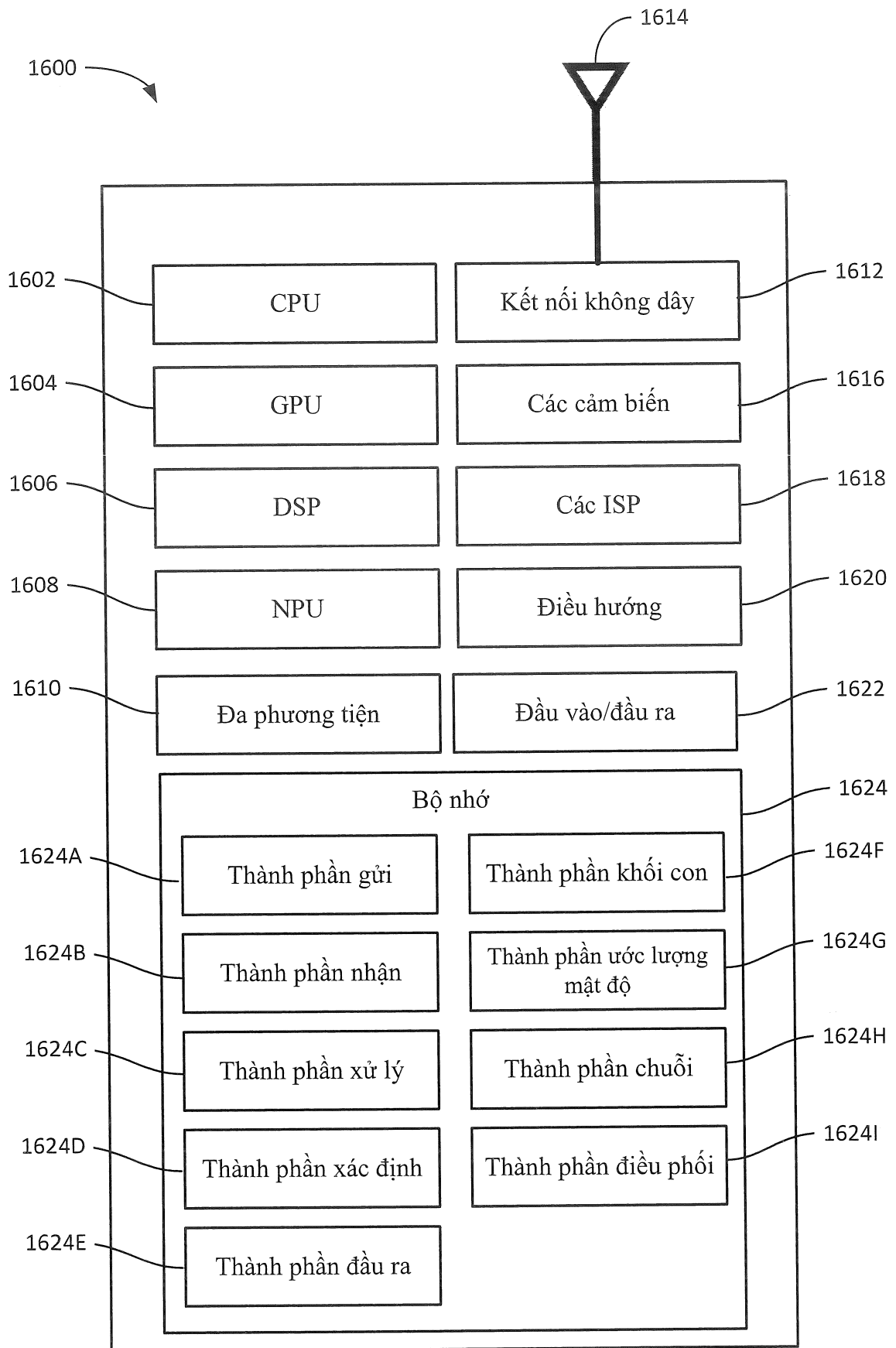


FIG. 16

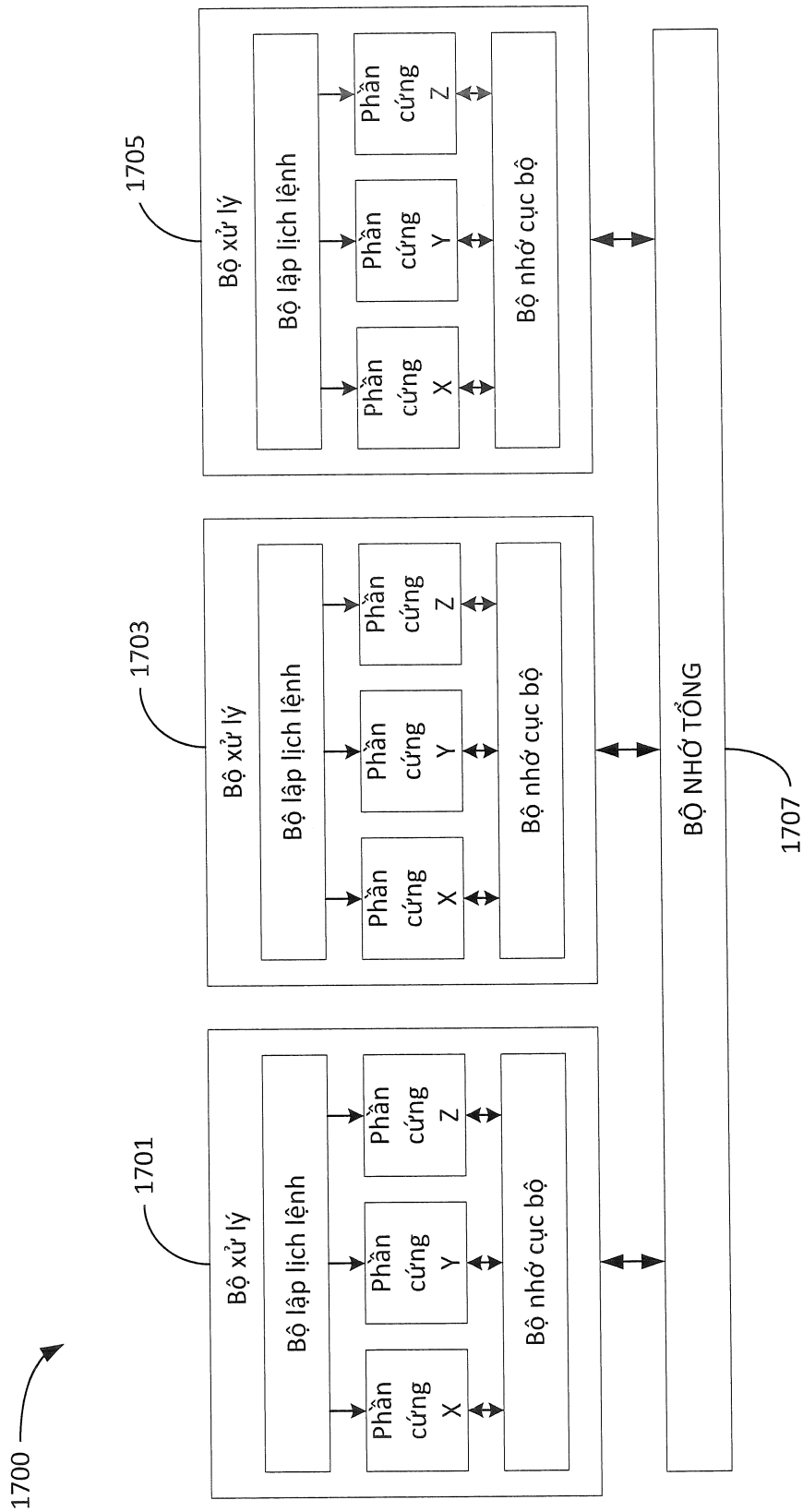


FIG. 17