



(12) **BẢN MÔ TẢ GIẢI PHÁP HỮU ÍCH THUỘC BẰNG ĐỘC QUYỀN
GIẢI PHÁP HỮU ÍCH**

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

(11)



2-0004432

(51) **G01C 21/00**
2023.01

(13) **Y**

(21) 2-2023-00796

(22) 30/10/2020

(67) 1-2020-06299

(45) 27/10/2025 451

(43) 25/06/2021 399A

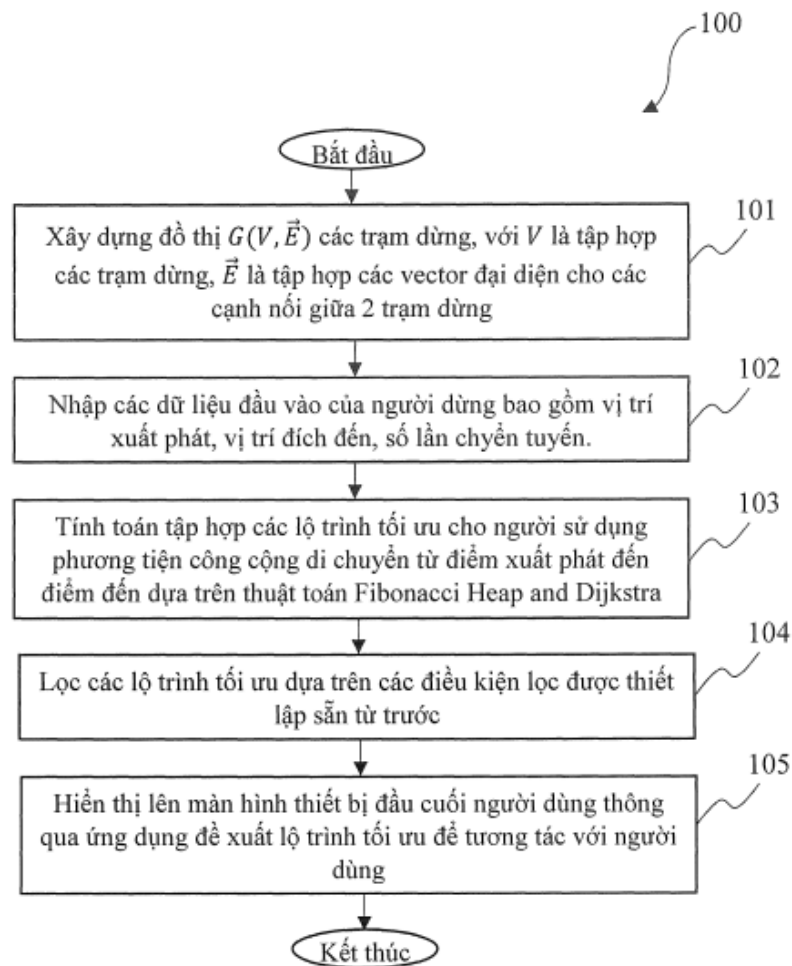
(73) Công ty Cổ phần công nghệ Phenikaa Maas (VN)

Số 373-375 Nguyễn Trọng Tuyển, phường 2, quận Tân Bình, thành phố Hồ Chí Minh

(72) Lê Yên Thanh (VN).

(54) **HỆ THỐNG ĐỀ XUẤT LỘ TRÌNH TỐI ƯU CHO NGƯỜI SỬ DỤNG PHƯƠNG
TIỆN CÔNG CỘNG**

(57) Giải pháp hữu ích tiết lộ hệ thống đề xuất lộ trình tối ưu cho người dùng phương tiện công cộng di chuyển giữa hai điểm bất kì trên bản đồ giao thông đô thị. Phương pháp bao gồm tạo một đồ thị các trạm dừng $G(V, \vec{E})$ trên bản đồ giao thông đô thị; nhận thông tin về đầu vào của người dùng từ ứng dụng đề xuất lộ trình tối ưu, bao gồm vị trí điểm xuất phát, vị trí điểm đến, số lần chuyển tuyến tối đa để đi từ điểm xuất phát đến điểm đến; tìm tập hợp các đường đi tối ưu giữa điểm xuất phát và điểm đến dựa trên sự kết hợp của hai thuật toán Fibonacci Heap và Dijkstra; lọc bỏ các cách đi tối ưu không đặt yêu cầu của nhà quản trị; và hiển thị các lộ trình tối ưu lên màn hình thiết bị đầu cuối người dùng đang sử dụng ứng dụng đề xuất lộ trình tối ưu Web/App. Hệ thống vận hành đề xuất đường đi tối ưu bao gồm thiết bị đầu cuối người dùng, ứng dụng đề xuất lộ trình tối ưu Web/App, cơ sở dữ liệu lớn (Big Data) và hệ thống các máy chủ. Phương pháp và hệ thống đề xuất lộ trình đi tối ưu cho nhiều kết quả hướng dẫn đường đi giúp cho người dùng có nhiều sự lựa chọn và linh hoạt về thời gian.



HÌNH 1

Lĩnh vực kỹ thuật được đề cập

Giải pháp hữu ích liên quan đến hỗ trợ tự động trong việc tìm kiếm các tuyến đường trong các khu vực đô thị trên bản đồ giao thông, liên quan đến đi bộ, phương tiện công cộng và các phương thức giao thông khác. Cụ thể hơn, giải pháp hữu ích đề cập đến hệ thống đề xuất lộ trình tối ưu cho người sử dụng phương tiện công cộng di chuyển giữa hai điểm bất kì trên bản đồ giao thông đô thị.

Tình trạng kỹ thuật của giải pháp hữu ích

Với sự phát triển nhanh chóng của công nghệ định vị vị trí, hệ thống định vị toàn cầu và dịch vụ bản đồ đã thu hút được sự chú ý và bản đồ được kết hợp hiệu quả với công nghệ định vị vị trí đem lại nhiều triển vọng để ứng dụng rộng rãi trong nhiều lĩnh vực. Kèm theo đó, nhiều ứng dụng (App) chỉ đường đang được phát triển trên khắp thế giới để phục vụ việc đi lại của con người. Nhiều quốc gia rất chú trọng việc sử dụng phương tiện công cộng trong các khu đô thị hay thành phố, vì sử dụng phương tiện công cộng giải quyết được nhiều vấn đề trong xã hội như giảm ùn tắc giao thông do giảm thiểu được số lượng phương tiện cá nhân kéo theo giảm nhiên liệu tiêu thụ và giảm lượng khí CO₂ thải ra môi trường. Để thuận tiện cho việc sử dụng phương tiện công cộng, nhiều ứng dụng hướng dẫn cho người dùng những lộ trình tốt nhất di chuyển giữa hai điểm bất kì trên bản đồ giao thông đô thị bằng phương tiện công cộng.

Để tối ưu hóa việc hướng dẫn cho người dùng những lộ trình tốt nhất di chuyển giữa hai điểm bất kì trên bản đồ giao thông đô thị bằng phương tiện công cộng, các phương pháp và thuật toán tìm đường đi ngắn nhất được chú trọng nghiên cứu và áp dụng. Các thuật toán được sử dụng khác phổ biến để tìm đường đi ngắn nhất có thể kể đến như thuật toán Dijkstra, thuật toán đường đi K-shortest, thuật toán A*,

Hiện tại ở Việt Nam, nhiều ứng dụng hướng dẫn lộ trình tối ưu bằng phương tiện công cộng chỉ mới hoàn thành việc hướng dẫn di chuyển bằng phương tiện công cộng, chưa có tính năng xem thời gian thực tế xe đến trạm, chưa có tính năng đề xuất tuyến xe dựa trên tốc độ di chuyển được theo dõi bằng GPS, chưa có tính năng thông báo người dùng khi đến trạm dừng, v.v. Hơn nữa, Google Map cũng có tính năng hướng dẫn di chuyển bằng phương tiện công cộng nhưng cũng chưa hoàn thiện tất cả các tính năng vừa nêu ở trên.

Trong đơn xin cấp bằng sáng chế mang số US7957871B1 của Hoa Kỳ, các tác giả đã tiết lộ phương pháp, hệ thống và công cụ để cung cấp bản đồ và chỉ đường, bao gồm chỉ đường đi bộ và phương tiện công cộng trong môi trường đô thị nơi người dùng không tự lái xe ô tô và/hoặc chỉ đường lái xe. Sáng chế cung cấp các tiêu chí di chuyển khác nhau bao gồm các tùy chọn về đoạn đường đi bộ, các tên đường cụ thể, một số loại phương tiện giao thông công cộng nhất định, số lần chuyển giữa các phương tiện giao thông công cộng và các ưu tiên khác, được xem xét tính toán trong tuyến đường di chuyển cũng như thông tin liên kết khác như điều kiện hiện tại và các thông tin khác. Sáng chế còn cung cấp việc ghi nhận và lưu trữ các phản hồi của người dùng về các khía cạnh định tính khác nhau của các hướng dẫn chỉ đường để sử dụng trong các quyết định định tuyến tiếp theo.

Trong đơn xin cấp bằng sáng chế mang số CN104572811B của Trung Quốc, các tác giả đã cung cấp phương pháp đề xuất lộ trình đi xe buýt, và các thiết bị vận hành. Phương pháp đề xuất lộ trình đi xe buýt được khuyến nghị bao gồm: nhận tuyến đường đề xuất phương tiện công cộng để thực hiện điều chỉnh; xác định các điểm nối của tuyến đường đề xuất cần điều chỉnh; và hợp nhất các tuyến đường đề xuất cần điều chỉnh theo các điểm nối. Ngoài ra, sáng chế CN104572811B còn cung cấp nhiều lựa chọn tuyến xe buýt, giúp người dùng tránh bị chiếm dụng quá nhiều thời gian do đợi xe buýt, để thực hiện một cách hiệu quả lộ trình đề nghị vận chuyển công cộng hiệu quả hơn trong việc sử dụng xe buýt qua các tuyến đường.

Trong đơn xin cấp bằng sáng chế mang số CN103364002B của Trung Quốc, các tác giả tiết lộ một hệ thống và phương pháp lập kế hoạch tuyến đường kết hợp điều kiện đường xá theo thời gian thực và thông tin vận hành giao thông công cộng. Phương pháp này bao gồm: xác định địa điểm khởi hành và điểm đến, thiết lập thông tin thời gian và tải lên máy chủ; máy chủ nhận địa điểm khởi hành, điểm đến và thông tin thời gian, lập kế hoạch tuyến đường kết hợp với thông tin vận hành theo thời gian thực của giao thông công cộng và điều kiện đường xá. Hệ thống của sáng chế CN103364002B bao gồm máy chủ, thiết bị đầu cuối và máy khách.

Do đó, điều cần thiết là cần có một ứng dụng kết hợp và cải tiến tất cả các tính năng kể trên bao gồm hướng dẫn di chuyển bằng phương tiện công cộng, xem thời gian thực xe đến trạm, đề xuất tuyến xe dựa trên tốc độ di chuyển được theo dõi bằng GPS, thông báo người dùng khi đến trạm.

Điều cần thiết nữa là cần có một phương pháp tìm đường đi tối ưu giữa điểm xuất phát và điểm đến cải thiện được tốc độ xử lý, đề xuất được nhiều tuyến xe để cung cấp nhiều lựa chọn cho người dùng.

Giải pháp hữu ích cung cấp các giải pháp đạt được các mục tiêu trên.

Bản chất kỹ thuật của giải pháp hữu ích

Giải pháp hữu ích cung cấp một hệ thống đề xuất lộ trình tối ưu cho người sử dụng phương tiện công cộng bao gồm thiết bị đầu cuối người dùng, ứng dụng đề xuất lộ trình tối ưu, cơ sở dữ liệu lớn (Big Data) và hệ thống máy chủ. Thiết bị đầu cuối người dùng được người dùng sử dụng để cài đặt và sử dụng ứng dụng đề xuất lộ trình tối ưu. Ứng dụng đề xuất lộ trình tối ưu giao tiếp với người dùng, thu thập yêu cầu tìm đường đi của người dùng bao gồm thông tin điểm xuất phát, điểm đến, số lần chuyển tuyến tối đa và và thực hiện các chức năng bao gồm đề xuất lộ trình tối ưu, tra cứu thông tin từng tuyến xe, xem thời gian thực xe đến trạm, và xem các thông báo từ hệ thống. Cơ sở dữ liệu lớn (Big Data) nhận các thông tin và yêu cầu tìm đường đi của người dùng từ ứng dụng đề xuất lộ trình tối ưu, xử lý thông tin và chạy thuật toán tìm đường đi tối ưu để gợi ý các lộ trình di chuyển tối ưu cho người dùng sử dụng phương tiện công cộng. Hệ thống máy chủ chứa số lượng lớn các máy chủ thực hiện chức năng tính toán và lưu trữ hình thành một trung tâm xử lý dữ liệu, là nơi lưu trữ các thông tin về lộ trình của phương tiện công cộng, lưu trữ thông tin các trạm dừng trên bản đồ giao thông đô thị giúp cho cơ sở dữ liệu lớn (Big Data) tính toán đề xuất lộ trình tối ưu khi sử dụng phương tiện công cộng; kiểm soát và phân chia lưu lượng tài nguyên đến các máy chủ hợp lý và bảo vệ dữ liệu tránh các cuộc tấn công từ bên ngoài hệ thống.

Theo một phương án thực hiện của giải pháp hữu ích, ứng dụng đề xuất tìm đường đi tối ưu Web/app bao gồm mô-đun đề xuất lộ trình tối ưu, mô-đun tra cứu thông tin phương tiện công cộng, mô-đun dự báo xe đến trạm theo thời gian thực, mô-đun thông báo, trong đó:

mô-đun đề xuất lộ trình tối ưu cho phép người dùng nhập thông tin về vị trí điểm xuất phát và vị trí điểm đến và trả về nhiều cách đi tối ưu thông qua nhiều tuyến xe khác nhau, hiển thị bao gồm cả thông tin về giá tiền, thời gian đi, quãng đường đi bộ, đi xe, số lần chuyển tuyến tối đa;

mô-đun tra cứu thông tin phương tiện công cộng cho phép tra cứu thông tin chi tiết của từng tuyến xe buýt, xe buýt sông, tàu điện; các thông tin bao gồm biểu đồ giờ xuất bến, danh sách các trạm dừng, thông tin mô tả chi tiết của tuyến xe (giá vé, lộ trình di chuyển, loại xe, độ dài, doanh nghiệp quản lý, ...), các đánh giá về tuyến xe;

mô-đun dự báo xe đến trạm theo thời gian thực cho phép xem được chi tiết thông tin các xe đang đi đến trạm đó; các thông tin xe bao gồm: biển số, khoảng cách, tốc độ, thời gian đến trạm, loại xe, tuyến xe;

mô-đun thông báo cho phép xem các thông báo được gửi từ hệ thống bao gồm các thông tin liên quan đến vé xe, cập nhật các thông tin hoạt động của các tuyến xe, các khuyến mãi và cập nhật từ hệ thống.

Giải pháp hữu ích cung cấp thêm phương pháp đề xuất đường đi tối ưu cho người sử dụng phương tiện công cộng bao gồm:

i) tạo một đồ thị $G(V, \vec{E})$ các trạm dừng của phương tiện công cộng trên bản đồ giao thông đô thị, với V là tập hợp các trạm dừng và $\vec{E}$ là tập hợp các vector cạnh nối giữa hai trạm dừng;

ii) nhận dữ liệu đầu vào bao gồm điểm xuất phát, điểm đến, số lần chuyển tuyến để thực hiện đề xuất lộ trình tối ưu để bắt phương tiện công cộng đi từ điểm xuất phát đến điểm đến;

iii) tìm tập hợp các đề xuất lộ trình tối ưu từ điểm xuất phát đến điểm đến dựa trên sự kết hợp hai thuật toán Fibonacci Heap và Dijkstra;

iv) lọc các đề xuất lộ trình tối ưu trong danh sách các đề xuất lộ trình tối ưu tìm được ở bước (iii);

v) hiển thị lên màn hình ứng dụng các đề xuất lộ trình để di chuyển từ điểm xuất phát đến điểm đến;

trong đó, thuật toán Dijkstra sử dụng một hàng đợi ưu tiên để theo dõi các đỉnh đã được truy cập và khoảng cách hiện tại của chúng từ nguồn; các đỉnh nằm trong hàng đợi ưu tiên này, nơi thường thực hiện các tối ưu hóa cho thuật toán Dijkstra; thuật toán Fibonacci Heap để triển khai hàng đợi ưu tiên.

Theo đó, mục đích thứ nhất của giải pháp hữu ích là cung cấp hệ thống vận hành tìm đường đi tối ưu cho phương tiện công cộng cho người sử dụng, hỗ trợ đi lại cho người sử dụng.

Mục đích thứ hai của giải pháp hữu ích là cung cấp một ứng dụng đề xuất đường đi tối ưu kết hợp nhiều tính năng bao gồm hướng dẫn đi xe công cộng, xem thời gian thực xe đến trạm, đề xuất tuyến xe dựa trên tốc độ di chuyển được theo dõi trên GPS, thông báo người dùng khi đến trạm.

Mục đích thứ ba của giải pháp hữu ích là cung cấp một thuật toán đề xuất đường đi tối ưu giữa điểm xuất phát và điểm đến cải thiện được tốc độ xử lý, đề xuất được nhiều tuyến xe để cung cấp nhiều lựa chọn cho người dùng.

Những ưu điểm này và ưu điểm khác của giải pháp hữu ích sẽ không còn nghi ngờ gì nữa đối với những người trình độ kỹ năng và kỹ thuật hiểu biết thông thường sau khi đọc phần mô tả chi tiết về các phương án được ưu tiên, và được minh họa trong các hình vẽ khác nhau dưới đây.

Mô tả vắn tắt các hình vẽ

Các bản vẽ đi kèm được kết hợp và tạo thành một phần của giải pháp hữu ích này nhằm minh họa các điểm yêu cầu bảo hộ độc lập và phụ thuộc của giải pháp hữu ích, cũng như phục vụ cho việc giải thích các nguyên lý hoạt động của toàn bộ giải pháp kỹ thuật, trong đó:

Hình 1 là lưu đồ minh họa phương pháp đề xuất lộ trình tối ưu cho người dùng phương tiện công cộng di chuyển từ điểm xuất phát đến điểm đến theo một phương án của giải pháp hữu ích.

Hình 2 là lưu đồ minh họa phương pháp xây dựng đồ thị các trạm dừng của phương tiện công cộng trên bản đồ giao thông đô thị theo một phương án của giải pháp hữu ích.

Hình 3 là sơ đồ minh họa một ví dụ về kết quả thực hiện phương pháp xây dựng đồ thị các trạm dừng của phương tiện công cộng trên bản đồ giao thông đô thị theo một phương án của giải pháp hữu ích.

Hình 4 là lưu đồ minh họa thuật toán đề xuất lộ trình tối ưu cho người dùng phương tiện công cộng di chuyển từ điểm xuất phát đến điểm đến trên bản đồ giao thông đô thị theo một phương án của giải pháp hữu ích.

Hình 5 là sơ đồ minh họa ví dụ thực hiện thuật toán đề xuất lộ trình tối ưu cho người dùng phương tiện công cộng di chuyển từ điểm xuất phát đến điểm đến trên bản đồ giao thông đô thị theo một phương án mẫu mực của giải pháp hữu ích.

Hình 6 là lưu đồ minh họa phương pháp lọc các lộ trình tối ưu dựa trên danh sách các lộ trình tối ưu đã được đề xuất theo một phương án của giải pháp hữu ích.

Hình 7 là sơ đồ minh họa một hệ thống phân tích dữ liệu thực hiện chạy thuật toán đề xuất lộ trình tối ưu để tính toán đề xuất lộ trình tối ưu cho người sử dụng phương tiện công cộng theo một phương án của giải pháp hữu ích.

Hình 8 là sơ đồ minh họa cấu tạo của ứng dụng đề xuất lộ trình tối ưu Web/App cho người dùng phương tiện công cộng theo một phương án của giải pháp hữu ích.

Hình 9 là sơ đồ minh họa cấu trúc tổng quan của một hệ thống phần cứng đề xuất lộ trình tối ưu cho người sử dụng phương tiện công cộng theo một phương án của giải pháp hữu ích.

Hình 10A – Hình 10D là hình minh họa ví dụ đề xuất lộ trình tối ưu cho người dùng xe buýt được thực hiện bởi phương pháp đề xuất lộ trình tối ưu cho người dùng phương tiện công cộng theo một phương án của giải pháp hữu ích.

Hình 11A là hình minh họa một ví dụ về tính năng xem xe đến trạm theo thời gian thực của ứng dụng đề xuất lộ trình tối ưu Web/App theo một phương án của giải pháp hữu ích.

Hình 11B – Hình 11D là hình minh họa một ví dụ về tính năng tra cứu thông tin tuyến xe của ứng dụng đề xuất lộ trình tối ưu Web/App theo một phương án của giải pháp hữu ích.

Mô tả chi tiết giải pháp hữu ích

Mặc dù giải pháp hữu ích sẽ được mô tả cùng với các phương án ưu tiên, nhưng sẽ được hiểu rằng giải pháp hữu ích không có ý định giới hạn đối với các phương án này. Giải pháp hữu ích nhằm mục đích đảm bảo các lựa chọn thay thế, cải tiến và định lượng, có thể bao gồm trong tinh thần và phạm vi của giải pháp hữu ích được xác định bởi các yêu cầu bảo hộ được thêm vào. Hơn nữa, trong phần mô tả chi tiết sau đây của giải pháp hữu ích, nhiều chi tiết cụ thể được đặt ra để cung cấp sự hiểu biết thấu đáo về giải pháp hữu ích. Tuy nhiên, sẽ là hiển nhiên đối với người có trình độ hiểu biết thông thường có thể thực hiện được mà không cần thể hiện các chi tiết cụ thể trong giải pháp hữu ích này. Trong các trường hợp khác, các phương pháp đã biết, quy trình, thành phần và phạm vi ứng dụng không được mô tả chi tiết để không làm mờ đi các khía cạnh không cần thiết của giải pháp hữu ích.

Cần phải hiểu thêm rằng giải pháp hữu ích không giới hạn ở phương pháp, hợp chất, vật liệu, kỹ thuật sản xuất, sử dụng và ứng dụng cụ thể, được mô tả trong tài liệu này, vì những điều này có thể khác nhau. Cũng cần hiểu rằng thuật ngữ được sử dụng ở đây được sử dụng cho mục đích chỉ mô tả các phương án cụ thể và không nhằm mục đích giới hạn phạm vi của giải pháp hữu ích. Cần lưu ý rằng các thuật ngữ được sử dụng ở đây và trong các yêu cầu bảo hộ, các hình thức số ít, “một” và “một số ít” bao gồm các tham chiếu số nhiều trừ khi bối cảnh chỉ ra rõ ràng. Do đó, ví dụ, một tài liệu tham khảo về một phần tử, một phần tử tham chiếu đến một hoặc nhiều phần tử và bao gồm các phần tử tương đương được biết đến với những người có kỹ năng thông thường trong lĩnh vực kỹ thuật tương đương. Tương tự, đối với một ví dụ khác, một tham chiếu đến một bước đi hay một cách có nghĩa là một cách tham chiếu đến một hoặc nhiều bước hoặc phương tiện và có thể bao gồm các bước phụ và phương tiện phụ thuộc. Tất cả các liên từ được sử dụng phải được hiểu theo nghĩa bao hàm nhất có thể. Do đó, từ ngữ “ưu tiên” hay “nên” được hiểu là có định nghĩa về một lô-gic hoặc một lô-gic chứ không phải là một từ hay lô-gic độc quyền, trừ khi bối cảnh chỉ ra rõ ràng và cần thiết. Các cấu trúc được mô tả trong tài liệu này cũng được hiểu là để chỉ các chức năng tương đương của các cấu trúc đó. Ngôn ngữ có thể được hiểu để diễn đạt gần đúng nên được hiểu như vậy trừ khi bối cảnh khác chỉ rõ.

Trừ khi được định nghĩa khác, tất cả các thuật ngữ khoa học và kỹ thuật được sử dụng ở đây đều có cùng ý nghĩa như thường được hiểu bởi một trong những kỹ năng thông thường trong kỹ thuật mà giải pháp hữu ích này đề cập. Các phương pháp, kỹ thuật, thiết bị và vật liệu phổ biến được mô tả, mặc dù mọi phương pháp, kỹ thuật, thiết bị hoặc vật liệu tương tự hoặc tương đương với các phương pháp được mô tả trong tài liệu này có thể được sử dụng trong thực tế hoặc thử nghiệm của giải pháp hữu ích. Các cấu trúc được mô tả trong tài liệu này cũng được hiểu là để chỉ các chức năng tương đương của các cấu trúc đó. Giải pháp hữu ích hiện tại sẽ được mô tả chi tiết với tham chiếu đến các phương án của nó như được minh họa trong các bản vẽ kèm theo.

Các tài liệu tham khảo về phương án của một trong những phương án, hiện thực, ví dụ, có thể chỉ ra rằng các phương án của giải pháp hữu ích được mô tả có thể bao gồm một tính năng, cấu trúc hoặc đặc điểm cụ thể, nhưng không phải mọi phương án nhất thiết phải bao gồm các tính năng, cấu trúc hoặc đặc tính cụ thể. Hơn nữa, việc sử dụng lặp đi lặp lại cụm từ trong một phương án, trực tiếp hoặc trong một phương án mẫu mực, không nhất thiết phải đề cập đến cùng một phương án, mặc dù chúng có thể.

Các tiêu đề được cung cấp ở đây là để thuận tiện trong việc mô tả sáng giải pháp hữu ích và không được coi là hạn chế tiết lộ dưới bất kỳ hình thức nào.

Các thiết bị hoặc mô-đun hệ thống ít nhất là giao tiếp chung với nhau không cần phải liên lạc với nhau, trừ khi có quy định rõ ràng khác. Ngoài ra, các thiết bị hoặc mô-đun hệ thống ít nhất là giao tiếp chung với nhau có thể giao tiếp trực tiếp hoặc gián tiếp thông qua một hoặc nhiều trung gian.

Mô tả về một phương án với một số thành phần trong giao tiếp với nhau không có nghĩa là tất cả các thành phần như vậy là bắt buộc. Trái lại, một loạt các thành phần tùy chọn được mô tả để minh họa cho nhiều phương án có thể có của giải pháp hữu ích.

Các khía cạnh của giải pháp hữu ích được mô tả dưới đây với tham chiếu đến sơ đồ minh họa và/hoặc sơ đồ khối của phương pháp, bộ máy (hệ thống) và các sản phẩm chương trình máy tính theo các phương án của giải pháp hữu ích. Điều này sẽ được hiểu rằng mỗi khối của hình minh họa sơ đồ và/hoặc lưu đồ khối, và sự kết hợp của các khối trong hình minh họa sơ đồ và/hoặc lưu đồ khối, có thể được thực hiện bằng lệnh chương trình máy tính. Các lệnh chương trình máy tính này có thể được cung cấp cho bộ xử lý của máy tính đa năng, máy tính có mục đích đặc biệt hoặc thiết bị xử lý dữ liệu lập trình khác để xử lý, như lệnh thực hiện thông qua bộ xử lý của máy tính hoặc thiết bị xử lý dữ liệu lập trình khác, tạo phương tiện để thực hiện các chức năng/hành vi được chỉ định trong sơ đồ khối và/hoặc khối hoặc lưu đồ khối.

Sơ đồ khối/lưu đồ khối trong các hình minh họa kiến trúc, chức năng và hoạt động của việc triển khai có thể của các hệ thống, phương pháp và các sản phẩm chương trình máy tính theo các phương án khác nhau. Về vấn đề này, mỗi khối trong sơ đồ khối hoặc sơ đồ khối có thể đại diện cho một mô-đun, phân đoạn hoặc một phần mã, bao gồm một hoặc nhiều lệnh thực thi để thực hiện (các) hàm lô-gic đã chỉ định. Cũng cần lưu ý rằng, trong một số triển khai thay thế, các chức năng được ghi chú trong khối có thể xảy ra ngoài trật tự được ghi trong các hình. Ví dụ, trên thực tế, hai khối được hiển thị có thể được thực thi đồng thời hoặc các khối đôi khi có thể được thực hiện theo thứ tự ngược lại, tùy thuộc vào chức năng liên quan. Cũng cần lưu ý rằng mỗi khối của sơ đồ khối và/hoặc lưu đồ khối, và sự kết hợp các khối trong sơ đồ khối và/hoặc lưu đồ khối, có thể được thực hiện bởi các hệ thống dựa trên phần cứng có mục đích đặc biệt thực hiện các chức năng hoặc hành vi được chỉ định, hoặc kết hợp các lệnh đặc biệt về phần cứng máy tính.

Hơn nữa, mặc dù các bước trong quy trình, các bước trong phương pháp, thuật toán hoặc tương tự có thể được mô tả theo thứ tự tuần tự, các quy trình, phương pháp và thuật toán đó có thể được cấu hình để hoạt động theo thứ tự thay thế. Nói cách khác, bất kỳ chuỗi hoặc thứ tự các bước có thể được mô tả không nhất thiết chỉ ra một yêu cầu rằng các bước được thực hiện theo thứ tự đó. Các bước của quy trình được mô tả trong tài liệu này có thể được thực hiện theo bất kỳ thứ tự thực tế nào. Hơn nữa, một số bước có thể được thực hiện đồng thời.

Giải pháp hữu ích hiện tại sẽ được mô tả chi tiết với tham chiếu đến các phương án của nó như được minh họa trong các bản vẽ kèm theo sau đây.

Bây giờ đề cập đến Hình 1, là lưu đồ minh họa phương pháp đề xuất lộ trình tối ưu cho người dùng phương tiện công cộng từ điểm xuất phát đến điểm đến 100 (“phương pháp 100”) theo một phương án của giải pháp hữu ích. Cần hiểu rằng lộ trình tối ưu có nghĩa là khoảng đường đi bộ và đi xe công cộng từ điểm xuất phát đến điểm đến trong khoảng thời

gian cho phép, đoạn đường đi bộ của người dùng không quá lớn và số lần chuyển tuyến không được vượt quá 3.

Theo các phương án của giải pháp hữu ích, phương pháp 100 bao gồm thêm một ứng dụng đề xuất lộ trình tối ưu Web/App được thực thi trên một thiết bị đầu cuối người dùng để giao tiếp với người dùng, cơ sở dữ liệu lớn (Big Data) để chạy thuật toán tìm đề xuất lộ trình tối ưu cho người sử dụng phương tiện công cộng di chuyển giữa điểm xuất phát và điểm đến trên bản đồ giao thông đô thị. Chi tiết về ứng dụng đề xuất lộ trình tối ưu Web/App, cơ sở dữ liệu lớn và thuật toán tìm đề xuất lộ trình tối ưu sẽ được đề cập chi tiết từ Hình 4 – Hình 11 và sẽ thảo luận sau.

Theo các phương án của giải pháp hữu ích, phương tiện công cộng bao gồm xe buýt, tàu điện, tàu ngầm, taxi, thuyền.

Tiếp tục với Hình 1, phương pháp 100 bắt đầu từ bước 101, tạo một đồ thị $G(V, \vec{E})$ các trạm dừng của phương tiện công cộng trên bản đồ giao thông đô thị bao gồm:

- a) xây dựng một tập hợp V các trạm dừng đại diện bởi các điểm trên bản đồ giao thông đô thị;
- b) xây dựng một tập hợp $\vec{E}$ các vector là các cạnh nối có hướng giữa các trạm dừng thuộc cùng một tuyến xe hoặc các trạm dừng lân cận khác tuyến xe nhưng có khoảng cách nằm trong phạm vi cho phép; hai thuật toán Heuristic và KDTree được sử dụng để giải quyết vấn đề kết nối các trạm dừng;
- c) xây dựng trọng số của các cạnh nối giữa hai trạm dừng theo công thức tính $w = R * Ratio$, với w là trọng số cạnh nối giữa hai trạm dừng, R là khoảng cách thực theo đường thẳng giữa 2 trạm dừng, $Ratio$ là tỉ lệ khoảng cách, được suy ra dựa trên vận tốc trung bình của tuyến và/hoặc đánh giá chất lượng xe buýt của người dùng và/hoặc tần suất xuất bến của các tuyến xe vào thời điểm hiện tại;

Ở bước 102, nhận dữ liệu đầu vào từ người dùng bao gồm điểm xuất phát, điểm đến, số lần chuyển tuyến tối đa để thực hiện đề xuất đường đi tối ưu và hướng dẫn đi từ điểm xuất phát đến điểm đến.

Ở bước 103, tính toán đề xuất tập hợp các lộ trình tối ưu cho người sử dụng phương tiện công cộng từ điểm xuất phát đến điểm đến bằng thuật toán tìm đường đi tối ưu 400 với sự kết hợp hai thuật toán Fibonacci Heap và Dijkstra. Thuật toán tìm đường đi tối 400 sẽ được mô tả chi tiết sau ở Hình 4. Theo các khía cạnh khác nhau của giải pháp hữu ích, lộ trình gọi là tối ưu nhất khi nó thỏa mãn các điều kiện sau:

- e) tổng trọng số các cạnh nối từ điểm xuất phát đến điểm đích là nhỏ nhất;
- f) tổng chiều dài đoạn đường đi bộ trong suốt chuyến đi nằm trong phạm vi cho phép;
- g) không được vượt quá ba lần chuyển tuyến để đi từ điểm xuất phát tới điểm đến.

Tiếp tục bước 103, thuật toán Fibonacci Heap là một cấu trúc dữ liệu cho các hoạt động hàng đợi ưu tiên, bao gồm một tập hợp các cây được sắp xếp theo thứ tự của Heap. Nó có thời gian chạy được khấu hao tốt hơn so với nhiều cấu trúc dữ liệu hàng đợi ưu tiên khác bao gồm cả Heap nhị phân và Heap nhị thức. Sử dụng Fibonacci Heap cho hàng đợi ưu tiên cải thiện thời gian chạy tiệm cận của các thuật toán quan trọng, chẳng hạn như thuật

toán Dijkstra để tính toán đường đi ngắn nhất giữa hai nút trong biểu đồ, so với cùng một thuật toán sử dụng cấu trúc dữ liệu hàng đợi ưu tiên chậm hơn khác.

Tiếp tục bước 103, Thuật toán Dijkstra là một thuật toán để tìm đường đi ngắn nhất giữa các nút trong một biểu đồ, ví dụ, có thể đại diện cho mạng lưới đường. Thuật toán tồn tại trong nhiều biến thể. Thuật toán ban đầu của Dijkstra đã tìm thấy đường đi ngắn nhất giữa hai nút nhất định, nhưng một biến thể phổ biến hơn sửa một nút duy nhất làm nút "nguồn" và tìm đường đi ngắn nhất từ nguồn đến tất cả các nút khác trong biểu đồ, tạo ra một cây đường đi ngắn nhất. Hơn nữa, giải pháp thuật toán nổi tiếng nhất cho vấn đề tìm đường đi ngắn nhất này là thuật toán Dijkstra, sử dụng một hàng đợi ưu tiên để theo dõi các đỉnh đã được truy cập và khoảng cách hiện tại của chúng từ nguồn. Nó nằm trong hàng đợi ưu tiên này, nơi thường thực hiện các tối ưu hóa cho thuật toán Dijkstra. Một cách để triển khai hàng đợi ưu tiên này là như một Fibonacci Heap. Các hình ảnh trực quan tương tác ở đây phục vụ cho việc hiển thị cách hoạt động của các hoạt động trên Fibonacci Heap thực sự hoạt động, cả dưới dạng một Heap độc lập và khi được sử dụng trong thuật toán Dijkstra như một hàng đợi ưu tiên.

Ở bước 104, lọc các đề xuất lộ trình tối ưu trong danh sách các lộ trình tối ưu được đề xuất ở bước 103 dựa trên các điều kiện lọc được thiết lập sẵn từ trước. Các điều kiện lọc bao gồm:

- h) loại bỏ các lộ trình tối ưu khác nhau có cùng tuyến xe, chỉ giữ lại 1 lộ trình tối ưu nhất;
- i) bỏ qua các đề xuất tuyến đi có tổng quãng đường đi bộ nhiều hơn 2km;
- k) từ cách đi tối ưu thứ 3 trở đi và của nhóm đi từ 2 tuyến trở lên: nếu quãng đường đi bộ lớn hơn 10% quãng đường đi bộ của cách đi tối ưu nhất trong nhóm 1 lần chuyển tuyến thì được bỏ qua;
- l) từ cách đi tối ưu thứ 3 trở đi trong các nhóm các cách đi tối ưu, nếu quãng đường đi bộ lớn hơn gấp đôi quãng đường đi bộ của cách đi tối ưu nhất trong nhóm đó thì được bỏ qua;

Ở bước 105, hiển thị lên màn hình thiết bị đầu cuối người dùng thông qua ứng dụng đề xuất lộ trình tối ưu Web/App các đề xuất và hướng dẫn di chuyển bằng cả đường đi bộ và đi xe công cộng để di chuyển từ điểm xuất phát đến điểm.

Bây giờ đề cập đến Hình 2, là lưu đồ khối minh họa phương pháp xây dựng đồ thị các trạm dừng trên bản đồ giao thông đô thị 200 ("phương pháp 200") theo một phương án của giải pháp hữu ích. Phương pháp 200 diễn giải chi tiết các bước thực hiện để xây dựng một đồ thị $G(V, E)$ các trạm dừng ở bước 101 của phương pháp 100. Phương pháp 200 được bắt đầu từ bước 201, hình thành tập hợp các trạm dừng V bằng cách sắp xếp các trạm dừng v tăng dần theo vĩ độ trên bản đồ giao thông đô thị. Mỗi phần tử v được xác định bằng $v(x, y, z)$, với x là khoảng cách giữa 2 trạm dừng đang được xem xét đến, y là số tuyến xe để đi đến trạm dừng v với cách đi tối ưu thứ z . Ta có, $d[v] = d[x][y][z]$ là khoảng cách ngắn nhất giữa trạm dừng có cạnh nối với trạm dừng v , đi y tuyến khác nhau để đến v với cách đi tối ưu thứ z .

Ở bước 202, hình thành các cạnh nối liên kết giữa các trạm dừng thuộc cùng một tuyến xe. Đồng thời, các cạnh nối có hướng chỉ hướng di chuyển của tuyến xe di chuyển từ trạm dừng bắt đầu đến trạm dừng kết thúc. Cần lưu ý rằng một trạm dừng có thể có nhiều tuyến xe đi qua.

Ở bước 203, hình thành các cạnh nối liên kết giữa các trạm dừng lân cận của các tuyến xe khác nhau có khoảng cách nằm trong phạm vi cho phép (VD: 200m, 300m, 400m, ...). Theo một phương án thực hiện của giải pháp hữu ích, phạm vi cho phép giữa các trạm dừng lân cận được người lập trình quy định trong khi lập trình các thuật toán. Các cạnh nối này cũng có hướng thỏa mãn cả chiều đi và chiều về giữa hai trạm dừng.

Theo các phương án của giải pháp hữu ích, hai thuật toán Heuristic và KDTree được sử dụng để tạo thành các cạnh nối giữa hai trạm dừng. Tập hợp các cạnh nối được tạo thành ở bước 202 và 203 tạo thành tập hợp $\vec{E}$ các vector cạnh nối.

Theo các phương án của giải pháp hữu ích, thuật toán Heuristic, dùng cho mật độ các tuyến phân bố đều, không tập trung. Ưu điểm của thuật toán Heuristic là nó làm cho ứng dụng đề xuất lộ trình tối ưu Web/App chạy nhanh, cài đặt dễ, ít tốn bộ nhớ; khuyết điểm là ứng dụng đề xuất lộ trình tối ưu Web/App sẽ chạy chậm trong một số trường hợp mật độ các trạm phân bố không đồng đều. Sắp xếp các trạm tăng dần theo vĩ độ, sau đó quét các trạm theo nhánh cận để kết nối các trạm dừng gần nhau trong bán kính nhất định (VD: 200m, 300m, 400m, ..).

Theo các phương án của giải pháp hữu ích, thuật toán KDTree, dùng cho mật độ các trạm phân bố không đồng đều. Ưu điểm của thuật toán KDTree là nó làm cho ứng dụng đề xuất lộ trình tối ưu Web/App chạy nhanh; khuyết điểm là khó cài đặt, tốn bộ nhớ. Với mỗi trạm được thêm vào KDTree (Ở đây dùng biến thể là cấu trúc 2D Tree) có thể tìm kiếm nhanh các trạm dừng xung quanh trạm đó trong bán kính nhất định (VD: 200m, 300m, 400m,).

Ở bước 204, tính toán và gán trọng số w cho các cạnh nối tạo thành ở bước 202 và bước 203. Trọng số w của các cạnh nối được tính bằng công thức $w = R * Ratio$, với R là tỉ trọng khoảng cách và $Ratio$ là khoảng cách thực theo đường thẳng giữa hai trạm nối.

Tiếp tục bước 204, tỉ trọng khoảng cách R được tính toán dựa trên các yếu tố bao gồm:

l) vận tốc trung bình của tuyến tại mỗi thời điểm trong ngày cũng như các ngày khác nhau trong tuần thông qua dữ liệu thu thập được từ GPS, qua đó có thể biết được vận tốc trung bình của một tuyến so với các tuyến khác vào một thời điểm trong ngày;

m) đánh giá chất lượng xe buýt của người dùng thu thập được từ ứng dụng đề xuất đường đi tối ưu Web/App;

n) tần suất xuất bến của các tuyến xe vào thời điểm hiện tại, ví dụ tuyến xe có tần suất xuất bến 5 phút/lần sẽ được ưu tiên cao hơn tuyến xe có tần suất xuất bến 15 phút/lần;

Theo một phương án của giải pháp hữu ích, sau khi kết thúc bước 204, một đồ thị $G(V, \vec{E})$ được hình thành với V là tập hợp các điểm đại diện cho các trạm dừng, và $\vec{E}$ là tập hợp các vector có hướng đại diện cho các cạnh nối giữa các trạm dừng.

Bây giờ đề cập đến Hình 3, là sơ đồ minh họa một đồ thị $G(V, \vec{E})$ 300 được xây dựng dựa trên phương pháp 200 theo một phương án của giải pháp hữu ích. Đồ thị $G(V, \vec{E})$ 300 bao gồm một số lượng lớn các trạm dừng 310 và một số lượng lớn các cạnh nối 302. Các trạm dừng 310 được sắp theo chiều tăng dần theo vĩ độ. Mỗi trạm dừng 310 có một hoặc nhiều tuyến xe đi qua, ví dụ như trạm dừng 310-E có 2 tuyến xe đi qua là tuyến xe số một 301 và tuyến xe số hai 302. Các cạnh nối 320 có dấu mũi tên chỉ hướng di chuyển của phương tiện công cộng. Các cạnh nối 320 được hình thành giữa 2 trạm dừng 310 thuộc cùng một tuyến xe, ví dụ như cạnh nối giữa trạm dừng 310-A và trạm dừng 310-B nằm trên cùng tuyến xe

số một 301. Trên mỗi cạnh nối 320, các trọng số w được dùng để xác định mức độ ưu tiên của các cạnh nối 320 cũng như các tuyến đường. Phương pháp tính trọng số w được thực hiện như bước 204 của phương pháp 200.

Ví dụ, các trạm dừng 310-A $\rightarrow$ 310-B $\rightarrow$ 310-C $\rightarrow$ 310-E $\rightarrow$ 310-K $\rightarrow$ 310-I nằm trên cùng tuyến xe số một 301 hình thành các cạnh nối có mũi tên chỉ hướng đi từ trạm dừng 310-A đến trạm dừng 310-I của tuyến; các trạm dừng 310-N $\rightarrow$ 310-G $\rightarrow$ 310-H $\rightarrow$ 310-E $\rightarrow$ 310-F nằm trên cùng tuyến xe số hai 302 hình thành các cạnh nối 320 có mũi tên chỉ hướng đi từ trạm dừng 310-N đến trạm dừng 310-F của tuyến; các trạm dừng 310-O $\rightarrow$ 310-H $\rightarrow$ 310-I $\rightarrow$ 310-L $\rightarrow$ 310-M $\rightarrow$ 310-S nằm trên cùng tuyến xe số ba 303 hình thành các cạnh nối 320 có mũi tên chỉ hướng đi từ trạm dừng 310-O đến trạm dừng 310-S của tuyến.

Cần lưu ý rằng số lượng các trạm dừng 310, các cạnh nối 320 và trọng số w của cạnh nối 320 theo Hình 3 chỉ nhằm minh họa cho khả năng đọc và hiểu, điều này không nhằm hạn chế bản chất kỹ thuật của giải pháp hữu ích, nhất là giải pháp hữu ích ở dạng tiết lộ. Trên thực tế, giải pháp hữu ích có thể thực hiện với số lượng trạm dừng 310 là không giới hạn và các cạnh nối 320 là phi tuyến với các trọng số w là số tùy biến.

Bây giờ đề cập đến Hình 4, là lưu đồ minh họa thuật toán tìm đường đi tối ưu đề xuất lộ trình tối ưu cho người dùng phương tiện công cộng di chuyển từ điểm xuất phát đến điểm đến 400 (“thuật toán 400”) theo một phương án của giải pháp hữu ích. Thuật toán 400 được bắt đầu từ bước 401, nhận dữ liệu đầu vào của người dùng bao gồm vị trí điểm xuất phát, vị trí điểm đến và số lần chuyển tuyến tối đa từ ứng dụng đề xuất lộ trình tối ưu.

Ở bước 402, khởi tạo hàm $Hash(V)$ ghi lại tên tuyến xe đi qua các trạm dừng 310 đã và đang xét duyệt làm điều kiện đầu vào của phương pháp lọc các lộ trình tối ưu 600 để loại bỏ các lộ trình đi tối ưu khác nhau có cùng tuyến xe. Các lộ trình tối ưu khác nhau có cùng tên tuyến xe, có nghĩa là các cách đi tối ưu này có hướng dẫn đi khác nhau nhưng các tuyến xe phải đi là như nhau. Thuật toán chỉ giữ lại một lộ trình tối ưu nhất trong các lộ trình tối ưu khác nhau có cùng tuyến xe.

Ở bước 403, xét các trạm dừng thuộc V có khoảng cách nằm trong phạm vi cho phép với điểm xuất phát. Với mỗi trạm dừng v có khoảng cách nằm trong phạm vi cho phép với điểm xuất phát, cập nhật giá trị $d[v] = d[x][1][1]$ là khoảng cách ngắn nhất từ điểm xuất phát đến trạm dừng v , $y = 1$ và $z = 1$ là các khởi tạo ban đầu. Các điểm $v(x, 1, 1)$ được thêm vào cấu trúc Fibonacci Heap tương ứng với các giá trị $d[v]$. Cần lưu ý thêm rằng, các trạm dừng thuộc V còn lại chưa được xét đến đều có giá trị $d = \infty$.

Ở bước 404, đánh dấu các trạm dừng có khoảng cách nằm trong phạm vi cho phép với điểm đến để xác định trạm dừng cần tới để có thể đi đến đích.

Ở bước 405, nhập giá trị tham số Z , giá trị tham số Z được hiểu là số lộ trình tối ưu tối đa được đề xuất sau khi kết thúc thuật toán.

Ở bước 406, xét trạm dừng $(x, y, z) \in V$ có giá trị nhỏ nhất hiện tại trong cấu trúc Fibonacci Heap. Giả thuyết trạm dừng v được chọn là nút xét duyệt.

Ở bước 407, xem xét cấu trúc Fibonacci Heap, nếu cấu trúc Fibonacci Heap rỗng thì thực hiện phương pháp lọc các lộ trình tối ưu B ở bước 600. Ngược lại thì thực hiện bước 408.

Ở bước 408, xét xem trạm dừng v có phải là trạm dừng được đánh dấu. Nếu trạm dừng v là trạm được đánh dấu, chuyển trạng thái trạm dừng v sang trạng thái đã được xét duyệt, được thực hiện ở bước 409 và sau đó tiếp tục thực hiện bước 406. Ngược lại thì thực hiện tiếp tục bước 410.

Ở bước 409, sau khi chuyển trạng thái trạm dừng v sang trạng thái đã xét duyệt thì quay lại thực hiện bước 406; đó là lấy một trạm dừng có giá trị nhỏ nhất ở trạng thái đang xét duyệt trong cấu trúc Fibonacci Heap để tiếp tục xét duyệt.

Ở bước 410, xét các trạm dừng $u \in V$ ($u \neq v$) lân cận có cạnh nối với trạm dừng v .

Ở bước 411, nếu không có trạm dừng nào hợp lệ thì thực hiện bước 409 và tiếp tục quay lại thực hiện bước 406. Ngược lại, có trạm dừng u hợp lệ thì thực hiện tính giá trị $d[u]$, y_u và z_u cho trạm dừng u ở các bước 413 và 414 tiếp theo. Cần hiểu rằng, trạm dừng hợp lệ là trạm dừng có cạnh nối với trạm dừng v đang xét duyệt, và cạnh nối có mũi tên hướng từ trạm dừng v đang xét duyệt đến trạm dừng u hợp lệ.

Ở bước 413, tính giá trị số lần chuyển tuyến y_u cần có để đi đến trạm dừng u . Nếu trạm dừng u nằm trên cùng một tuyến xe với trạm dừng v thì $y_u = y_v$, nếu u nằm khác tuyến xe với v thì $y_u = y_v + 1$.

Ở bước 414, tính toán giá trị $d[u] = d[v] + w$, với w là trọng số cạnh nối giữa u và v . Sau đó, tìm các giá trị z cho trạm dừng u dựa vào kết quả $d[u]$. Mỗi trạm dừng u có thể có nhiều giá trị $d[u]$ tương ứng với nhiều cách đi để đến trạm dừng u thông qua các trạm dừng lân cận với trạm dừng u , giá trị $d[u]$ nhỏ nhất tương ứng với cách đi tối ưu thứ nhất $z = 1$, tương tự với $z = 2, z = 3, z = 4, \dots$. Các tham số z là dữ liệu đầu vào để tính toán các lộ trình tối ưu khác nhau đi từ điểm xuất phát đến điểm đến.

Ở bước 415, cập nhật giá trị $d[u]$ nhỏ nhất vào cấu trúc Fibonacci Heap và tiếp tục thực hiện bước 409 chuyển trạng thái trạm dừng v sang trạng thái đã xét duyệt và tiếp tục thực hiện bước 406 lấy trạm dừng có giá trị nhỏ nhất trong cấu trúc Fibonacci Heap ($\neq$ trạm dừng v) để tiếp tục xét duyệt. Lúc này, trạm dừng u đóng vai trò là điểm xét duyệt, và các điểm đã xét duyệt sẽ không được xét đến trong cấu trúc Fibonacci Heap.

Bây giờ đề cập đến Hình 5, là sơ đồ minh họa ví dụ thực hiện tìm đường đi tối ưu bằng thuật 400 theo một phương án mẫu mực của giải pháp hữu ích. Thuật toán 400 được minh họa bởi ví dụ ở Hình 5 như sau:

Bước 1, một điểm xuất phát 510-P và điểm đến 520-Q được thêm vào đồ thị $G(V, E)$ và xác định các trạm dừng 310 có khoảng cách trong phạm vi cho phép với điểm 510-P và 520-Q. Tham chiếu Hình 5, điểm có khoảng cách trong phạm vi cho phép với điểm 510-P là 310-A và 310-N; điểm có khoảng cách trong phạm vi cho phép với điểm 510-Q là 310-M và 310-S. Các trọng số w được tính cho các cạnh nối 310 từ điểm 510-P $\rightarrow$ 310-A, 510-P $\rightarrow$ 310-N, 520-Q $\rightarrow$ 310-M, 520-Q $\rightarrow$ 310-S theo công thức $w = R * Ratio$ lần lượt là 5, 6, 4, 10.

Bước 2, khởi tạo một hàm $Hash(V)$ ghi lại các tuyến xe 301-303 đi qua các trạm dừng 310 đã và đang xét duyệt. Hàm $Hash(V)$ được dùng làm điều khiển đầu vào nhằm loại bỏ các cách đi tối ưu khác nhau có cùng số tuyến xe đi từ điểm xuất phát 510-P đến điểm đến 520-Q, các cách đi tối ưu này có cùng số tuyến xe nhưng khác nhau về hướng dẫn đi.

Bước 3, xác định các trạm dừng 310 có khoảng cách gần nhất với điểm 310-P, có trạm dừng 310-A và 310-N. Trạm dừng 310-A có $d_A = 5$, trạm dừng 310-N có $d_N = 6$, các trạm dừng 310 khác chưa được xét đến có $d = \infty$ (theo quy ước của thuật toán). Trạm dừng 310-A (5,1,1) và trạm dừng 310-N (6,1,1) được thêm vào cấu trúc Fibonacci Heap. Trạm dừng 310-A có giá trị nhỏ nhất trong cấu trúc Fibonacci Heap, nên 310-A sẽ được chọn để làm điểm bắt đầu xét duyệt. Xét các trạm dừng 310 lân cận có cạnh nối 320 với trạm dừng 310-A, hiện tại chỉ có trạm dừng 310-B hợp lệ. Với trạm dừng 310-B, ta có $d_B = d_A + w = 15$, trạm dừng 310-B được xác định là 310-B (15,1,1) cùng với d_B được cập nhật vào cấu trúc Fibonacci Heap.

Bước 4, chọn trạm dừng 310-N (6,1,1) có giá trị nhỏ nhất trong cấu trúc Fibonacci Heap. Xét các trạm dừng 310 lân cận có cạnh nối 320 với trạm dừng 310-N, có trạm dừng 310-G hợp lệ. Với trạm dừng 310-G, ta có $d_G = d_N + w = 14$, trạm dừng 310-G được xác định là 310-G (14,1,1) cùng với d_G được cập nhật vào cấu trúc Fibonacci Heap.

Bước 5, chọn trạm dừng 310-G (14,1,1) có giá trị nhỏ nhất trong cấu trúc Fibonacci Heap. Xét các trạm dừng 310 lân cận có cạnh nối 320 với 310-G, có trạm dừng 310-H hợp lệ. Với trạm dừng 310-H, ta có $d_H = d_G + w = 24$, trạm dừng 310-H được xác định là 310-H (24,1,1) cùng với d_H được cập nhật vào cấu trúc Fibonacci Heap.

Bước 6, chọn trạm dừng 310-B (15,1,1) có giá trị nhỏ nhất trong cấu trúc Fibonacci Heap. Xét các trạm dừng 310 lân cận có cạnh nối 320 với trạm dừng 310-B, có trạm dừng 310-C hợp lệ. Với trạm dừng 310-C, ta có $d_C = d_B + w = 2$, trạm dừng 310-C được xác định là C (29,1,1) cùng với d_C được cập nhật vào cấu trúc Fibonacci Heap.

Bước 7, chọn trạm dừng 310-H (24,1,1) có giá trị nhỏ nhất trong cấu trúc Fibonacci Heap. Xét các trạm dừng 310 lân cận có cạnh nối 320 với trạm dừng 310-H, có hai trạm dừng 310-E, 310-I hợp lệ. Với trạm dừng 310-E, ta có $d_E = d_H + w = 41$, trạm dừng 310-E được xác định là E (41,1,1) cùng với d_E được cập nhật vào cấu trúc Fibonacci Heap. Với trạm dừng 310-I, ta có $d_I = d_H + w = 37$; trạm dừng 310-H và trạm dừng 310-I thuộc 2 tuyến khác nhau nên $y_I = y_H + 1 = 2$; trạm dừng 310-I được xác định là 310-I (37,2,1) cùng với d_I được cập nhật vào cấu trúc Fibonacci Heap.

Bước 8, chọn trạm dừng 310-C (29,1,1) có giá trị nhỏ nhất trong cấu trúc Fibonacci Heap. Xét các trạm dừng 310 lân cận có cạnh nối 320 với trạm dừng 310-C, có trạm dừng 310-E hợp lệ. Với trạm dừng 310-E, ta có $d_E = d_C + w = 42$, trạm dừng 310-E được xác định là 310-E (42,1,1). Trạm dừng 310-E có 2 cách đi tối ưu để đi đến tương đương với hai giá trị $d_E = 41$ và $d_E = 42$ nên trạm dừng 310-E có hai giá trị là (41,1,1) và (42,1,1). Giá trị 310-E (42,1,1) cùng với $d_E = 42$ không được cập nhật vào cấu trúc Fibonacci Heap.

Bước 9, chọn trạm dừng 310-I (37,2,1) có giá trị nhỏ nhất trong cấu trúc Fibonacci Heap. Xét các trạm dừng 310 lân cận có cạnh nối 320 với trạm dừng 310-I, có trạm dừng 310-L hợp lệ. Với trạm dừng 310-L, ta có $d_L = d_I + w = 58$, trạm dừng 310-L được xác định là L (58,2,1) cùng với d_L được cập nhật vào cấu trúc Fibonacci Heap.

Bước 10, chọn trạm dừng 310-E (41,1,1) có giá trị nhỏ nhất trong cấu trúc Fibonacci Heap. Xét các trạm dừng 310 lân cận có cạnh nối 320 với trạm dừng 310-E, có hai trạm dừng 310-F và 310-K hợp lệ. Với trạm dừng 310-F, ta có $d_F = d_E + w = 63$, trạm dừng 310-F được xác định là 310-F (63,1,1) cùng với d_F được cập nhật vào cấu trúc Fibonacci Heap. Với trạm dừng 310-K, ta có $d_K = d_E + w = 59$, trạm dừng 310-K được xác định là K (59,1,1) cùng với d_K được cập nhật vào cấu trúc Fibonacci Heap.

Bước 11, chọn trạm dừng 310-L (58,2,1) có giá trị nhỏ nhất trong cấu trúc Fibonacci Heap. Xét các trạm dừng 310 lân cận có cạnh nối 320 với trạm dừng 310-L, có trạm dừng 310-M hợp lệ. Với trạm dừng 310-M, ta có $d_M = d_L + w = 72$ và trạm dừng 310-M được xác định là 310-M (72,2,1) cùng với d_M được cập nhật vào cấu trúc Fibonacci Heap.

Bước 12, chọn trạm dừng 310-K (59,1,1) có giá trị nhỏ nhất trong Fibonacci Heap. Xét các trạm dừng 310 lân cận có cạnh nối 320 với trạm dừng 310-K, có trạm dừng 310-I hợp lệ. Với trạm dừng 310-I, ta có $d_I = d_K + w = 69$ và trạm dừng 310-I được xác định là 310-I (69,1,2). Trạm dừng 310-I có 2 cách đi tối ưu để đi đến tương đương với hai giá trị $d_I = 37$ và $d_I = 69$ nên trạm dừng 310-I có hai giá trị là (37,1,1) và (69,1,1). Giá trị 310-I (69,1,1) cùng với $d_I = 69$ không được cập nhật vào cấu trúc Fibonacci Heap.

Bước 13, chọn trạm dừng 310-F (63,1,1) có giá trị nhỏ nhất trong cấu trúc Fibonacci Heap. Xét trạm dừng 310 lân cận có cạnh nối 320 với trạm dừng 310-F, không có điểm nào hợp lệ, chuyển trạng thái điểm 310-F sang trạng thái đã xét duyệt

Bước 14, chọn trạm dừng 310-M (72,1,1) có giá trị nhỏ nhất trong cấu trúc Fibonacci Heap. Trạm dừng 310-M là điểm được đánh dấu vì có khoảng cách nằm trong phạm vi cho phép với điểm 310-Q, do đó chuyển trạng thái điểm 310-M sang trạng thái đã xét duyệt và tiếp tục lấy điểm nhỏ nhất trong cấu trúc Fibonacci Heap để tiếp tục tính toán.

Bước 15, không còn điểm nào đang xét duyệt trong cấu trúc Fibonacci Heap nên các lộ trình tối ưu từ điểm 310-P đến điểm 310-Q được xác định dựa trên các trạm dừng 310 đã xét duyệt và các giá trị cách đi tối ưu z bao gồm:

a) lộ trình tối ưu thứ nhất: $510-P \rightarrow 310-A \rightarrow 310-B \rightarrow 310-C \rightarrow 310-E \rightarrow 310-K \rightarrow 310-I \rightarrow 310-L \rightarrow 310-M \rightarrow 520-Q$, và hướng dẫn đi chi tiết bao gồm: từ 510-P đi bộ đến trạm dừng 310-A bắt tuyến xe số một 301 đi đến trạm dừng 310-I xuống xe và bắt tiếp tục tuyến xe số hai 302 để đi đến trạm dừng 310-M; xuống xe tại trạm dừng 310-M và đi bộ đến điểm đến 520-Q.

b) lộ trình tối ưu thứ hai: $510-P \rightarrow 310-N \rightarrow 310-G \rightarrow 310-H \rightarrow 310-I \rightarrow 310-L \rightarrow 310-M \rightarrow 520-Q$, hướng dẫn đi chi tiết bao gồm: từ 510-P đi bộ đến trạm dừng 310-N bắt tuyến xe số hai 302 đi đến trạm dừng 310-H xuống xe và bắt tiếp tục tuyến xe số ba 303 để đi đến trạm dừng 310-M; xuống xe tại trạm dừng 310-M và đi bộ đến điểm đến 520-Q.

c) lộ trình tối ưu thứ ba: $510-P \rightarrow 310-N \rightarrow 310-G \rightarrow 310-H \rightarrow 310-E \rightarrow 310-K \rightarrow 310-I \rightarrow 310-L \rightarrow 310-M \rightarrow 520-Q$, hướng dẫn đi chi tiết bao gồm: từ 510-P đi bộ đến trạm dừng 310-N bắt tuyến xe số hai 302 đi đến trạm dừng 310-E xuống xe và bắt tiếp tục tuyến xe số một 301 để đi đến trạm dừng 310-I xuống xe và bắt tuyến xe số ba 303 đi từ trạm dừng 310-I đến trạm dừng 310-M; xuống xe tại trạm dừng 310-M và đi bộ đến điểm đến 520-Q.

Đề cập đến Hình 6, là lưu đồ minh họa phương pháp lọc các lộ trình tối ưu B 600 (“phương pháp 600”) theo một phương án của giải pháp hữu ích. Phương pháp 600 bắt đầu từ bước 601, sắp xếp và hệ thống lại các số tuyến xe phải đi để đi từ điểm xuất phát đến điểm đến dựa trên các trạm dừng đã được xét duyệt trong đồ thị $G(V, \vec{E})$ và dựa trên các cách đi tối ưu z .

Ở bước 602, tính tổng quãng đường cả đi xe và đi bộ của các cách đi tối ưu thứ z khác nhau từ điểm xuất phát đến điểm đích.

Ở bước 603, xem xét loại bỏ các lộ trình tối ưu có cùng số tuyến đi, chỉ giữ lại một lộ trình tối ưu duy nhất. Ví dụ, cách đi tối ưu thứ 3 có tuyến 10 và 20, cách đi tối ưu thứ 5 cũng có tuyến 10 và 20 nhưng khác nhau về cách chuyển trạm dừng và số quãng đường đi bộ xa hơn cách thứ 3 do đó cách đi tối ưu thứ năm bị loại.

Ở bước 604, phân nhóm các lộ trình tối ưu từ điểm xuất phát đến điểm đích dựa vào số lần chuyển tuyến để đi đến đích, ví dụ nhóm 1 lần chuyển tuyến có 3 cách đi tối ưu, nhóm 2 lần chuyển tuyến có 4 cách đi tối ưu, nhóm 3 lần chuyển tuyến có 3 cách đi tối ưu. Để đảm bảo chi phí đi xe thì 3 lần chuyển tuyến là tối đa để đi từ điểm xuất phát đến điểm đích.

Ở bước 605, xem xét quãng đường đi bộ ở các cách đi tối ưu thứ 3 trở lên của nhóm 2 lần chuyển tuyến và loại bỏ cách đi tối ưu có quãng đường đi bộ lớn hơn 10% quãng đường đi bộ của các đi tối ưu thứ nhất.

Ở bước 606, xem xét quãng đường đi bộ của các cách đi tối ưu thứ 3 trở lên của các nhóm và loại bỏ cách đi tối ưu có quãng đường đi bộ lớn hơn hai lần quãng đường đi bộ của cách đi tối ưu nhất trong cùng nhóm.

Ở bước 607, thực hiện truy vết ngược để đưa ra chi tiết các trạm dừng phải đi qua để hiển thị lộ trình chi tiết cho người dùng.

Tham chiếu đến Hình 7, là sơ đồ minh họa một hệ thống phân tích dữ liệu 700 thực hiện tính toán đề xuất lộ trình tối ưu cho người dùng phương tiện công cộng đi từ điểm xuất phát đến điểm đích theo một phương án của giải pháp hữu ích. Hệ thống phân tích dữ liệu 700 có cấu tạo bao gồm một số lượng các người dùng 710 từ 710-1 đến 710-N, ứng dụng đề xuất lộ trình tối ưu Web/App 800, hệ thống quản lý nội bộ 730, cơ sở dữ liệu lớn (Big Data) 740. Ứng dụng đề xuất lộ trình tối ưu Web/App 800, hệ thống quản lý nội bộ 730 và cơ sở dữ liệu lớn 740 được kết nối thông qua mạng 702 bằng các kênh truyền dẫn 701. Mạng 702 theo giải pháp hữu ích bao gồm mạng dựa trên đám mây (Cloud), mạng diện rộng (WAN), mạng đô thị (MAN), mạng cục bộ (LAN). Ứng dụng đề xuất lộ trình tối ưu Web/App 800 thu thập dữ liệu tìm lộ trình xe công cộng từ một số lượng lớn người dùng 710 và truyền đến cơ sở dữ liệu lớn 740 để phân tích và xử lý, đề xuất lộ trình tối ưu cho người dùng phương tiện công cộng. Cơ sở dữ liệu lớn 740 gửi các đề xuất lộ trình tối ưu đến ứng dụng đề xuất lộ trình tối ưu Web/App 800 và hiển thị kết quả đến người dùng 710. Các hoạt động giao tiếp giữa ứng dụng đề xuất lộ trình tối ưu Web/App 800 và cơ sở dữ liệu lớn 740 được giám sát và quản lý bởi hệ thống quản lý nội bộ 730.

Theo một số phương án của giải pháp hữu ích, kênh truyền dẫn 701 có thể là các kết nối vật lý khác nhau như các kênh giao tiếp không dây như Bluetooth, LTE/4G/5G, Wi-Fi, Zigbee, Z-wave, tần số vô tuyến (RF), tầm gần (NFC), LoRa hoặc giao tiếp có dây như Ethernet, RS-232, RS-485, USB hoặc bất kỳ kết hợp nào của chúng.

Tiếp tục với Hình 7, ứng dụng đề xuất lộ trình tối ưu Web/App 800 được cấu hình để gợi ý người dùng các tuyến xe công cộng đi từ vị trí xuất phát đến vị trí đích tối ưu về thời gian, chi phí và khoảng đường đi bộ. Ứng dụng đề xuất lộ trình tối ưu Web/App 800 có hai phiên bản là phiên bản Web và phiên bản ứng dụng di động App giúp thuận tiện cho việc truy cập và sử dụng. Hơn thế nữa, ứng dụng đề xuất lộ trình tối ưu Web/App 800 cho phép người dùng 710 tra cứu thông tin chi tiết của từng tuyến xe, xem thời gian chờ xe theo thời gian thực, theo dõi lộ trình di chuyển của phương tiện công cộng và đọc tên các trạm sắp đến. Các dữ liệu yêu cầu sử dụng các tính năng được ứng dụng đề xuất lộ trình tối ưu Web/App ghi nhận và truyền về cơ sở dữ liệu lớn 740 phân tích và xử lý. Chi tiết về ứng

dụng đề xuất lộ trình tối ưu Web/App 800 sẽ được đề cập chi tiết trong Hình 8 và sẽ thảo luận sau.

Tiếp tục với Hình 7, cơ sở dữ liệu lớn 740 thu nhận dữ liệu từ ứng dụng đề xuất lộ trình tối ưu Web/App 800, phân tích và xử lý dữ liệu, và trả về các đề xuất lộ trình tối ưu cho người dùng 710 sử dụng phương tiện công cộng để hiển thị lên ứng dụng đề xuất lộ trình tối ưu Web/App 800. Cơ sở dữ liệu lớn 740 có kiến trúc bao gồm hệ thống lưu trữ dữ liệu 741, hệ thống tiền xử lý 742, hệ thống trích xuất dữ liệu 743, hệ thống chuẩn hóa dữ liệu 744, trực quan hóa dữ liệu 745, giải thuật xử lý dữ liệu 746, bộ làm sạch dữ liệu đầu cuối 747, mã hóa và bảo mật 748, hệ thống phân quyền truy cập 749, và hệ thống đầu cuối truy xuất dữ liệu kết quả 750. Các bộ phận bên trong cơ sở dữ liệu lớn 740 được liên kết với nhau thông qua kênh truyền dẫn 701.

Cơ sở dữ liệu lớn 740 liên tục nhận dữ liệu từ ứng dụng đề xuất lộ trình tối ưu Web/App 800. Đầu tiên, dữ liệu được xử lý bởi hệ thống tiền xử lý 742, giải thuật xử lý dữ liệu 746, và bộ làm sạch dữ liệu đầu cuối 747. Sau đó, dữ liệu được lưu trữ bên trong hệ thống lưu trữ 741. Hệ thống trích xuất dữ liệu 743 và hệ thống chuẩn hóa dữ liệu 744 làm giảm bớt sự dư thừa và loại bỏ những sự cố mâu thuẫn về dữ liệu, tiết kiệm được không gian lưu trữ. Trực quan hóa dữ liệu 745 biến các nguồn dữ liệu thành những thông tin được thể hiện một cách trực quan, dễ quan sát, dễ hiểu, để truyền đạt thông điệp đưa ra từ dữ liệu.

Hơn nữa, mã hóa và bảo mật 748, hệ thống phân quyền truy cập 749 cung cấp các bảo mật dữ liệu cần thiết cho hệ thống cơ sở dữ liệu lớn 740. Hệ thống đầu cuối xuất dữ liệu kết quả 750 trả dữ liệu đã được xử lý về ứng dụng đề xuất lộ trình tối ưu Web/App 800.

Tiếp tục với Hình 7, hệ thống quản lý nội bộ 730 quản lý và giám sát hoạt động giao tiếp giữa ứng dụng đề xuất lộ trình tối ưu Web/App 800 và cơ sở dữ liệu lớn 740. Hệ thống quản lý nội bộ 730 truy cập vào hệ thống cơ sở dữ liệu lớn 740 để khắc phục các sự cố, bảo dưỡng và bảo trì hệ thống 700.

Bây giờ đề cập đến Hình 8, là sơ đồ minh họa một ứng dụng đề xuất lộ trình tối ưu Web/App 800 đề xuất lộ trình tối ưu cho người dùng phương tiện công cộng di chuyển từ điểm xuất phát đến điểm đến theo một phương án của giải pháp hữu ích. Ứng dụng đề xuất lộ trình tối ưu Web/App 800 bao gồm mô-đun đề xuất lộ trình tối ưu 810, mô-đun tra cứu thông tin phương tiện công cộng 820, mô-đun dự báo xe đến trạm theo thời gian thực 830, mô-đun thông báo 840.

Mô-đun đề xuất lộ trình tối ưu 810 cho phép người dùng nhập thông tin về vị trí điểm xuất phát và vị trí điểm đến để đề xuất đường đi tối ưu thông qua các tuyến phương tiện công cộng. Mô-đun đề xuất lộ trình tối ưu 810 trả về nhiều cách đi tối ưu thông qua nhiều tuyến xe khác nhau, hiển thị bao gồm cả thông tin về giá tiền, thời gian đi, quãng đường đi bộ, đi xe, số lần chuyển tuyến tối đa. Hơn nữa, mô-đun đề xuất lộ trình tối ưu 810 còn hướng dẫn chi tiết cách đi, các vị trí trạm đón xe, các vị trí trạm xuống xe, chi tiết thời gian và quãng đường của từng cách đi, và có thông tin thời gian thực về thời gian xe đến trạm.

Mô-đun tra cứu thông tin phương tiện công cộng 820 tra cứu thông tin chi tiết của từng tuyến xe buýt, tàu điện, tàu điện ngầm, taxi, thuyền. Các thông tin bao gồm biểu đồ giờ xuất bến, danh sách các trạm dừng, thông tin mô tả chi tiết của tuyến xe (giá vé, lộ trình di chuyển, quãng đường di chuyển, doanh nghiệp quản lý, ...), các đánh giá về tuyến xe.

Mô-đun dự báo xe đến trạm theo thời gian thực 830, bám vào một trạm dừng trên bản đồ có thể xem được chi tiết thông tin các xe đang đi đến trạm đó. Các thông tin xe bao gồm:

biển số, khoảng cách, tốc độ, thời gian đến trạm, loại xe, tuyến xe. Ngoài ra còn có thể xem được chi tiết các tuyến xe nào đi qua trạm đang chọn.

Mô-đun thông báo 840 xem các thông báo được gửi từ hệ thống bao gồm các thông tin liên quan đến vé xe, cập nhật các thông tin hoạt động của các tuyến xe, các khuyến mãi và cập nhật từ hệ thống 700.

Đề cập đến Hình 9, là sơ đồ minh họa cấu trúc tổng quang của một hệ thống phần cứng 900 (“hệ thống 900”) vận hành thuật toán tìm đường đi tối ưu 400, thuật toán lọc các lộ trình tối ưu B 600, ứng dụng đề xuất lộ trình tối ưu 800 theo một phương án của giải pháp hữu ích. Hệ thống phần cứng 900 bao gồm thiết bị đầu cuối người dùng 910, trung tâm xử lý dữ liệu 920, hệ thống quản lý nội bộ 930, hệ thống quản lý thiết bị IoT 940, hệ thống dự phòng 950. Thiết bị đầu cuối người dùng 910 kết nối với trung tâm xử lý dữ liệu 920 thông qua mạng 960. Trung tâm xử lý dữ liệu 920 liên kết với hệ thống quản lý nội bộ 930, hệ thống quản lý thiết bị IoT 940, hệ thống dự phòng 950 thông qua kênh truyền dẫn 901; trong đó, kênh truyền dẫn 901 là kênh truyền dẫn có dây bao gồm cáp RS485, RS232, RJ45, cáp quang học, v.v.

Tiếp tục với Hình 9, thiết bị đầu cuối người dùng 910 bao gồm điện thoại thông minh, máy tính bản, đồng hồ thông minh, PC, laptop, v.v. Thiết bị đầu cuối người dùng 910 được người dùng sử dụng để chạy ứng dụng đề xuất lộ trình tối ưu 800 và thực hiện các chức năng bao gồm đề xuất lộ trình tối ưu, tra cứu thông tin từng tuyến xe, xem thời gian thực xe đến trạm, và xem thông báo từ hệ thống.

Tiếp tục với Hình 9, trung tâm xử lý dữ liệu 920 bao gồm tường lửa 921, cân bằng tải 922, lõi (core value) xử lý 923, cơ sở dữ liệu nội bộ 924, cơ sở dữ liệu ngoài 925, và hệ thống dự phòng 926. Tường lửa 921 bảo vệ hệ thống 900 tránh khỏi các cuộc tấn công mạng để đánh cắp hoặc phá hủy dữ liệu. Cân bằng tải 922 giúp phân bố đều các lưu lượng truy cập đến các máy chủ có cùng chức năng trong hệ thống 900, giúp giảm thiểu tối đa tình trạng máy chủ quá tải và ngưng hoạt động. Lõi xử lý 923 thực hiện điều phối các máy chủ thực hiện tính toán, phân tích dữ liệu. Cơ sở dữ liệu nội bộ 924 lưu trữ các dữ liệu tính toán của hệ thống 900, hỗ trợ cung cấp dữ liệu cho lõi xử lý 923 tính toán. Cơ sở dữ liệu ngoài 925 bao gồm các dữ liệu chứa các thông tin chi tiết của các tuyến xe được bên thứ ba chia sẻ để bổ sung vào cơ sở dữ liệu nội bộ 924. Cơ sở dữ liệu ngoài 925 bao gồm cơ sở dữ liệu của chính phủ 925-1 và cơ sở dữ liệu phòng nghiên cứu và phát triển 925-2.

Theo một phương án của giải pháp hữu ích, trung tâm xử lý dữ liệu 920 được tạo thành từ một số lượng lớn các máy chủ. Các máy chủ có nhiều chức năng khác nhau bao gồm tính toán và lưu trữ. Trung tâm xử lý dữ liệu 920 vận hành cơ sở dữ liệu lớn 740 thực hiện đề xuất lộ trình tối ưu cho người dùng phương tiện công cộng di chuyển giữa hai điểm bất kỳ trên bản đồ giao thông đô thị.

Tiếp tục với Hình 9, hệ thống quản lý nội bộ 930 cho phép tạo các tài khoản quản trị được đăng nhập vào hệ thống để quản lý dữ liệu, giám sát hoạt động của hệ thống, và khắc phục các lỗi hệ thống có thể xảy.

Hệ thống quản lý thiết bị IoT 940 có nhiệm vụ quản lý các thiết bị IoT liên quan đến định vị hành trình, các thiết bị này có thể là các thiết bị định vị hành trình được lắp trên xe, cũng có thể là từ ứng dụng trên thiết bị đầu cuối người dùng 910 của người dùng. Tất cả dữ liệu được gửi về hệ thống quản lý thiết bị IoT 940 để được sàng lọc và xử lý trước khi gửi vào lõi xử lý 923 để phân tích.

Hệ thống dự phòng 950 có chức năng sao lưu dữ liệu và sẽ trở thành lỗi xử lý tạm thời khi lỗi xử lý 923 xảy ra lỗi hoặc cần bảo trì/nâng cấp hệ thống 900.

Hơn nữa, để hỗ trợ cho hoạt động của cơ sở dữ liệu nội bộ 924 và cơ sở dữ liệu ngoài 925, hệ thống phân cứng 900 còn bao gồm hệ thống quản trị cơ sở dữ liệu quan hệ - đối tượng (PostgreSQL) 970 và hệ thống quản trị cơ sở dữ liệu có cấu trúc (RedisDB) 980.

Hệ thống quản trị cơ sở dữ liệu quan hệ - đối tượng (PostgreSQL) 970 là một hệ thống quản trị cơ sở dữ liệu quan hệ miễn phí và nguồn mở (RDBMS) tập trung vào khả năng mở rộng và tuân thủ các tiêu chuẩn kỹ thuật. Nó được thiết kế để xử lý một loạt các khối lượng công việc lớn, từ các máy tính cá nhân đến kho dữ liệu hoặc dịch vụ Web có nhiều người dùng đồng thời, hỗ trợ cho cơ sở dữ liệu nội bộ 924 và cơ sở dữ liệu ngoài 925. PostgreSQL 970 sở hữu một hệ tính năng đa dạng giúp hỗ trợ các nhà phát triển xây dựng app, các nhà quản trị bảo vệ toàn vẹn dữ liệu, và tạo ra một môi trường chịu lỗi fault-tolerant giúp quản lý dữ liệu bất kể tập dữ liệu lớn hay nhỏ.

Hệ thống quản trị cơ sở dữ liệu có cấu trúc (RedisDB) 980 là nơi lưu trữ dữ liệu có cấu trúc, có thể sử dụng như một cơ sở dữ liệu, bộ nhớ cache hay một message broker. Ngoài tính năng lưu trữ key-value trên RAM thì RedisDB 980 còn hỗ trợ tính năng sắp xếp, hàng đợi, sao lưu dữ liệu trên đĩa cứng cho phép bạn có thể phục hồi dữ liệu khi hệ thống gặp sự cố...và có thể nhân bản (chạy nhiều máy chủ redisDB cùng một lúc).

Hệ thống quản trị cơ sở dữ liệu quan hệ - đối tượng (PostgreSQL) 970 và hệ thống quản trị cơ sở dữ liệu có cấu trúc (RedisDB) 980 được ứng dụng phổ biến trong lĩnh vực công nghệ thông tin nên chúng sẽ không được mô tả chi tiết để không làm mờ đi các khía cạnh không cần thiết của giải pháp hữu ích.

Theo các phương án của giải pháp hữu ích, mạng 960 và mạng 702, kênh truyền dẫn 901 và kênh truyền dẫn 701 là như nhau.

Đề cập đến Hình 10, là hình minh họa một ví dụ đề xuất lộ trình tối ưu cho người dùng đi xe buýt được thực hiện bởi phương pháp 100 và ứng dụng đề xuất lộ trình tối ưu 800 theo một phương án của giải pháp hữu ích. Giao diện hiển thị 1010 của Hình 10A đề xuất lộ trình tối ưu cho người dùng đi xe buýt di chuyển từ Công viên Phú Lâm đến Cổng trước Đại Học Quốc Tế và số lần chuyển tuyến là 3. Các lộ trình tối ưu 1011 được hiển thị, mỗi lộ trình tối ưu được hiển thị bao gồm tên tuyến xe cần đi, quãng đường đi bộ, dự báo thời gian xe đến trạm, thời gian đi và chi phí vé.

Tiếp tục với Hình 10B, giao diện hiển thị 1020 xuất hiện khi bấm vào một lộ trình tối ưu. Trong ví dụ, giao diện hiển thị 1020 hiển thị cho lộ trình tối ưu của hai tuyến xe 14 và 08 để di chuyển từ Công Viên Phú Lâm đến Cổng trước Đại Học Quốc Tế. Giao diện hiển thị 1020 hiển thị hướng dẫn di chuyển chi tiết của lộ trình tối ưu của tuyến xe 14 và 08. Hướng dẫn chi tiết được thể hiện cả trên bản đồ và được ghi chi tiết ở giao diện hiển thị 1030 của Hình 10C.

Tiếp tục với Hình 10D, giao diện hiển thị 1040 hiển thị chi tiết các trạm dừng cần phải đi qua để đi từ Công viên Phú Lâm đến Cổng trước Đại học Quốc Tế. Ứng dụng đề xuất lộ trình tối ưu 800 có thể đọc tên trạm dừng khi xe buýt di chuyển gần đến trạm.

Đề cập đến Hình 11A, là hình minh họa một ví dụ về tính năng dự báo xe đến trạm theo thời gian thực của ứng dụng đề xuất lộ trình tối ưu 800 theo một phương án của giải pháp

hữu ích. Giao diện hiển thị 1110 đang hiển thị tốc độ của các tuyến xe số 6 và thời gian các tuyến xe số 6 đến trạm dừng Đại Học Khoa học Tự nhiên.

Tiếp tục Hình 11B – Hình 11C, là hình minh họa một ví dụ về tính năng tra cứu thông tin phương tiện công cộng theo một phương án của giải pháp hữu ích. Giao diện hiển thị 1120 của Hình 11B hiển thị các tuyến xe buýt trên bản đồ giao thông của thành phố Hồ Chí Minh. Giao diện hiển thị 1130 của Hình 11C hiển thị chi tiết các thông tin của tuyến xe số 1 bao gồm tên tuyến, thời gian hoạt động, giá vé, các trạm đi qua, độ dài tuyến xe, giãn cách tuyến xe, số chuyến trong 1 ngày, tổ chức quản lý. Giao diện hiển thị 1140 của Hình 11D hiển thị chi tiết các trạm dừng của tuyến xe số 1.

Những ích lợi (hiệu quả) có thể đạt được

Giải pháp hữu ích cung cấp một ứng dụng đề xuất lộ trình tối ưu Web/App cho người sử dụng phương tiện công cộng đi giữa hai điểm bất kì trên bản đồ giao thông đô thị. Ứng dụng đề xuất lộ trình tối ưu Web/App giúp người dùng có thể đi xe công cộng từ điểm xuất phát đến điểm đến một cách thuận tiện nhất về thời gian, chi phí. Ứng dụng đề xuất lộ trình tối ưu Web/App gợi ý cho người dùng hướng đi bộ đến trạm dừng gần nhất để đi xe công cộng đến điểm đến. Nếu đi nhiều tuyến xe, ứng dụng đề xuất lộ trình tối ưu Web/App sẽ hướng dẫn chi tiết người dùng chuyển tuyến xe dựa trên các trạm dừng. Hơn nữa, ứng dụng đề xuất lộ trình tối ưu Web/App còn cho phép xem thời gian thực khi nào xe đến trạm dừng, đọc tên trạm dừng khi đến trạm và cho phép người dùng đánh giá về tuyến xe mình đã đi. Ngoài ra, người dùng có tra cứu thông tin từng tuyến xe công cộng trên ứng dụng đề xuất lộ trình tối ưu Web/App.

Giải pháp hữu ích cung cấp một phương pháp đề xuất lộ trình cho người dùng sử dụng phương tiện công cộng với hướng dẫn chi tiết nhiều lộ trình để đi từ điểm xuất phát đến điểm đến, giúp cho người dùng có nhiều phương án để lựa chọn tránh bị động về thời gian. Hơn nữa, phương pháp còn loại bỏ các tuyến xe không còn hoạt động ra khỏi thuật toán đề xuất đường đi tối ưu, và có thể ưu tiên đề xuất các tuyến xe ít bị ùn tắc giao thông tiết kiệm thời gian và giúp ích cho người sử dụng xe công cộng.

Các thảo luận/tuyên bố bên trên đã mô tả đầy đủ ít nhất một phương án của giải pháp hữu ích, các phương pháp tương đương hoặc thay thế khác để thực hiện một hệ thống đề xuất lộ trình tối ưu cho người sử dụng phương tiện công cộng theo giải pháp hữu ích sẽ rõ ràng với những người có kỹ năng trung bình trong lĩnh vực tương đương. Các khía cạnh khác nhau của giải pháp hữu ích đã được mô tả ở trên bằng cách minh họa, và các phương án cụ thể đã được tiết lộ không nhằm mục đích giới hạn giải pháp hữu ích đối với các hình thức cụ thể. Việc triển khai cụ thể hệ thống đề xuất lộ trình tối ưu cho người sử dụng phương tiện công cộng có thể khác nhau tùy thuộc vào bối cảnh hoặc ứng dụng cụ thể. Do đó, giải pháp hữu ích bao gồm tất cả các sửa đổi, tương đương và các lựa chọn thay thế nằm trong tinh thần và phạm vi của các yêu cầu bảo hộ sau đây. Cần phải hiểu thêm rằng không phải tất cả các phương án được tiết lộ trong mô tả đã nêu ở trên sẽ nhất thiết phải thỏa mãn hoặc đạt được từng đối tượng.

Yêu cầu các yếu tố và các bước trong tài liệu này có thể đã được đánh số và/hoặc ký tự chỉ như một sự trợ giúp cho khả năng đọc và hiểu. Bất kỳ việc đánh số và ký tự nào như vậy không nhằm mục đích và không nên được thực hiện để chỉ ra thứ tự của các yếu tố và/hoặc các bước trong yêu cầu bảo hộ.

Tóm tắt giải pháp hữu ích được cung cấp sẽ cho phép người đọc xác định bản chất và ý chính của công bố kỹ thuật. Nó được trình bày với sự hiểu biết rằng nó sẽ không được sử dụng để giới hạn hoặc giải thích phạm vi hoặc ý nghĩa của các yêu cầu bảo hộ. Các yêu cầu bảo hộ sau đây đã được đưa vào mô tả chi tiết, với mỗi yêu cầu bảo hộ đứng riêng như một phương án độc lập.

Chú thích hình vẽ

Hình 3, các chỉ dẫn được mô tả như sau:

301	Tuyến xe số một
302	Tuyến xe số hai
301	Tuyến xe số ba
310	Trạm dừng/điểm
310-A	Trạm dừng/điểm A
310-B	Trạm dừng/điểm B
310-C	Trạm dừng/điểm C
310-E	Trạm dừng/điểm E
310-F	Trạm dừng/điểm F
310-H	Trạm dừng/điểm H
310-G	Trạm dừng/điểm G
310-O	Trạm dừng/điểm O
310-K	Trạm dừng/điểm K
310-I	Trạm dừng/điểm I
310-L	Trạm dừng/điểm L
310-M	Trạm dừng/điểm M
310-N	Trạm dừng/điểm N
310-S	Trạm dừng/điểm S
320	Cạnh nối

Hình 5, các chỉ dẫn được mô tả như sau:

510-P	Điểm xuất phát
520-Q	Điểm đến

Hình 7, các chỉ dẫn được mô tả như sau:

701	Kênh truyền dẫn
702	Mạng
710	Người dùng

710-1	Người dùng thứ nhất
710-2	Người dùng thứ hai
710-3	Người dùng thứ ba
710-N	Người dùng thứ N
800	Ứng dụng đề xuất lộ trình tối ưu
730	Hệ thống quản lý nội bộ
740	Cơ sở dữ liệu lớn
741	Hệ thống lưu trữ dữ liệu
742	Hệ thống tiền xử lý
743	Hệ thống trích xuất dữ liệu
744	Hệ thống chuẩn hóa dữ liệu
745	Trực quan hóa dữ liệu
746	Giải thuật xử lý dữ liệu
747	Bộ làm sạch dữ liệu đầu cuối
748	Mã hóa và bảo mật
749	Hệ thống phân quyền truy cập
750	Hệ thống đầu cuối truy xuất dữ liệu kết quả

Hình 8, các chỉ dẫn được mô tả như sau:

810	Mô-đun đề xuất lộ trình tối ưu
820	Mô-đun tra cứu thông tin phương tiện công cộng
830	Mô-đun dự báo xe đến trạm thời gian thực
840	Mô-đun thông báo

Hình 9, các chỉ dẫn được mô tả như sau:

901	Kênh truyền dẫn
910	Thiết bị đầu cuối người dùng
920	Đơn vị xử lý dữ liệu
921	Tường lửa
922	Cân bằng tải
923	Lỗi xử lý
924	Cơ sở dữ liệu
925	Cơ sở dữ liệu ngoài
925-1	Cơ sở dữ liệu chính phủ

925-2	Phòng nghiên cứu phát triển
930	Hệ thống quản lý nội bộ
940	Hệ thống quản lý IoT
950	Hệ thống dự phòng
960	Mạng
970	Hệ thống quản trị cơ sở dữ liệu quan hệ - đối tượng (PostgreSQL)
980	Hệ thống quản trị cơ sở dữ liệu có cấu trúc (RedisDB)

YÊU CẦU BẢO HỘ

1. Hệ thống đề xuất lộ trình tối ưu cho người sử dụng phương tiện công cộng bao gồm ít nhất một thiết bị đầu cuối người dùng, ít nhất một ứng dụng đề xuất lộ trình tối ưu Web/App, cơ sở dữ liệu lớn (Big Data) và hệ thống máy chủ;

trong đó, thiết bị đầu cuối người dùng được cài đặt và chạy ứng dụng đề xuất lộ trình tối ưu Web/App, giao tiếp với nhiều người dùng;

trong đó, ứng dụng đề xuất lộ trình tối ưu Web/App giao tiếp với người dùng, thu thập yêu cầu tìm đường của người dùng bao gồm thông tin điểm xuất phát, điểm đến, số lần chuyển tuyến tối đa và thực hiện các chức năng bao gồm đề xuất lộ trình tối ưu, tra cứu thông tin từng tuyến xe, xem thời gian thực xe đến trạm, và xem thông báo từ hệ thống đề xuất lộ trình tối ưu;

trong đó, cơ sở dữ liệu lớn (Big Data) nhận các thông tin và yêu cầu tìm đường của nhiều người dùng từ ứng dụng đề xuất lộ trình tối ưu, xử lý thông tin và chạy thuật toán tìm đường đi tối ưu đề gợi ý các lộ trình di chuyển tối ưu cho người dùng sử dụng phương tiện công cộng;

trong đó, hệ thống máy chủ chứa các máy chủ thực hiện chức năng tính toán và lưu trữ, hình thành một trung tâm xử lý dữ liệu; trung tâm xử lý dữ liệu vận hành cơ sở dữ liệu lớn (Big Data); là nơi lưu trữ các thông tin về lộ trình của phương tiện công cộng, lưu trữ thông tin các trạm dừng trên bản đồ giao thông đô thị hỗ trợ cho cơ sở dữ liệu lớn (Big Data) tính toán đề xuất lộ trình tối ưu khi người dùng sử dụng phương tiện công cộng; kiểm soát và phân chia lưu lượng tài nguyên đến các máy chủ hợp lý và bảo vệ dữ liệu tránh các cuộc tấn công từ bên ngoài hệ thống đề xuất lộ trình tối ưu;

trong đó trung tâm xử lý dữ liệu bao gồm:

tường lửa bảo vệ hệ thống tránh khỏi các cuộc tấn công mạng để đánh cắp hoặc phá hủy dữ liệu;

cân bằng tải phân bố các lưu lượng truy cập đến các máy chủ có cùng chức năng trong hệ thống, giảm thiểu tối đa tình trạng máy chủ quá tải và ngưng hoạt động;

lỗi xử lý thực hiện điều phối các máy chủ thực hiện tính toán, phân tích dữ liệu;

cơ sở dữ liệu nội bộ lưu trữ các dữ liệu của hệ thống, hỗ trợ cung cấp dữ liệu cho lỗi xử lý tính toán;

cơ sở dữ liệu ngoài bao gồm các dữ liệu chứa các thông tin chi tiết của các tuyến xe được bên thứ ba chia sẻ để bổ sung vào cơ sở dữ liệu nội bộ.

2. Hệ thống đề xuất lộ trình tối ưu cho người sử dụng phương tiện công cộng theo điểm 1, bao gồm thêm:

hệ thống quản lý nội bộ quản lý và giám sát hoạt động giao tiếp giữa ứng dụng đề xuất lộ trình tối ưu Web/App và cơ sở dữ liệu lớn, truy cập vào cơ sở dữ liệu lớn để khắc phục các sự cố, bảo dưỡng và bảo trì hệ thống;

hệ thống dự phòng sao lưu dữ liệu và trở thành lỗi xử lý tạm thời khi lỗi xử lý chính xảy ra lỗi;

hệ thống quản trị cơ sở dữ liệu quan hệ - đối tượng (PostgreSQL) tập trung vào khả năng mở rộng và tuân thủ các tiêu chuẩn kỹ thuật, được thiết kế để xử lý một loạt các khối lượng công việc lớn, từ các máy tính cá nhân đến kho dữ liệu hoặc dịch vụ Web có nhiều người dùng đồng thời, hỗ trợ cho cơ sở dữ liệu nội bộ và cơ sở dữ liệu ngoài;

hệ thống quản trị cơ sở dữ liệu có cấu trúc (RedisDB) bao gồm tính năng lưu trữ những giá trị cốt lõi (key-value) trên RAM, hỗ trợ tính năng sắp xếp, hàng đợi, sao lưu dữ liệu trên đĩa cứng cho phép bạn có thể phục hồi dữ liệu khi hệ thống gặp sự cố và có thể nhân bản.

3. Hệ thống đề xuất lộ trình tối ưu cho người sử dụng phương tiện công cộng theo điểm 1, thiết bị đầu cuối người dùng bao gồm điện thoại thông minh, máy tính bàn, đồng hồ thông minh, máy tính để bàn, máy tính xách tay (laptop).

4. Hệ thống đề xuất lộ trình tối ưu cho người sử dụng phương tiện công cộng theo điểm 1, trong đó ứng dụng đề xuất lộ trình tối ưu Web/App bao gồm mô-đun đề xuất lộ trình tối ưu, mô-đun tra cứu thông tin phương tiện công cộng, mô-đun dự báo xe đến trạm theo thời gian thực và mô-đun thông báo, trong đó:

mô-đun đề xuất lộ trình tối ưu cho phép người dùng nhập thông tin về vị trí điểm xuất phát và vị trí điểm đến và trả về nhiều cách đi tối ưu thông qua nhiều tuyến xe khác nhau, hiển thị bao gồm cả thông tin về giá tiền, thời gian đi, quãng đường đi bộ, đi xe, số lần chuyển tuyến tối đa;

mô-đun tra cứu thông tin phương tiện công cộng cho phép tra cứu thông tin chi tiết của từng tuyến xe buýt, tàu điện, tàu điện ngầm, tắc xi, thuyền; các thông tin bao gồm biểu đồ giờ xuất bến, danh sách các trạm dừng, thông tin mô tả chi tiết của tuyến xe (giá vé, lộ trình di chuyển, loại xe, quãng đường di chuyển, doanh nghiệp quản lý), các đánh giá về tuyến xe;

mô-đun dự báo xe đến trạm theo thời gian thực cho phép xem chi tiết thông tin các xe đang đi đến trạm; các thông tin xe bao gồm: biển số, khoảng cách, tốc độ, thời gian đến trạm, loại xe, tuyến xe;

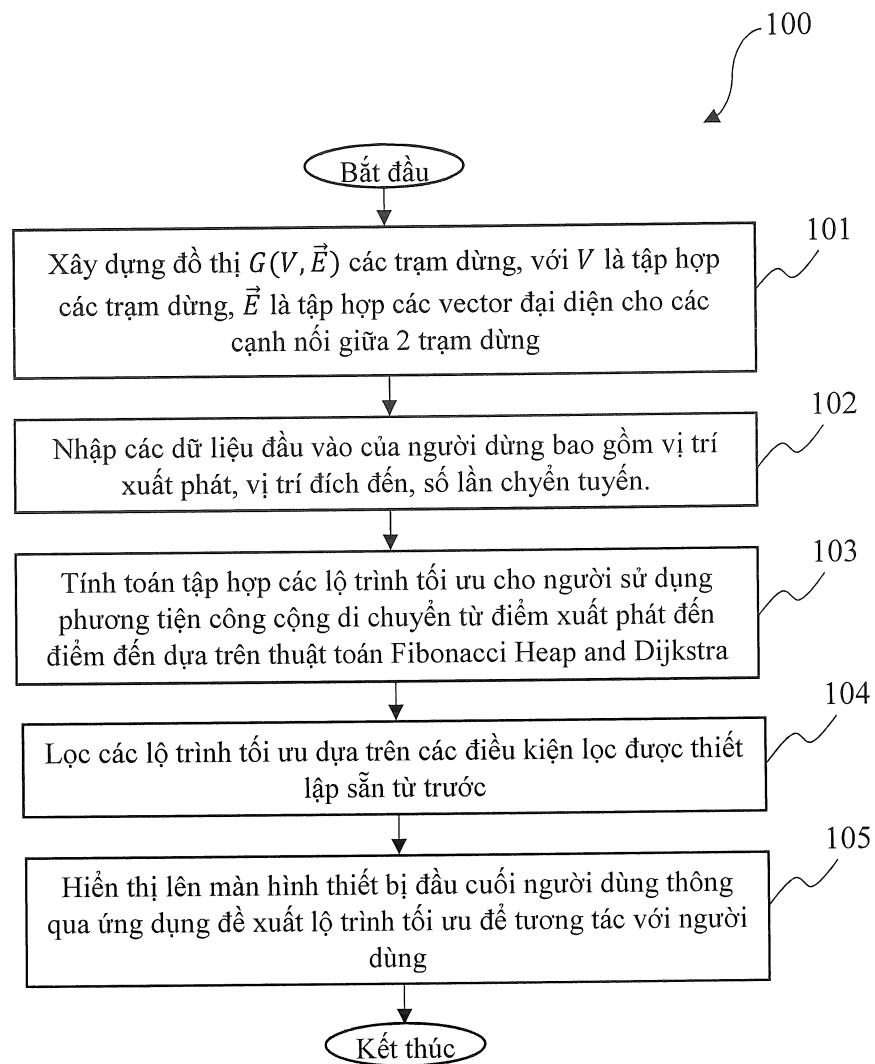
mô-đun thông báo cho phép xem các thông báo được gửi từ hệ thống bao gồm các thông tin liên quan đến vé xe, cập nhật các thông tin hoạt động của các tuyến xe, các khuyến mãi và cập nhật từ hệ thống đề xuất lộ trình tối ưu.

5. Hệ thống đề xuất lộ trình tối ưu cho người sử dụng phương tiện công cộng theo điểm 1, trong đó cơ sở dữ liệu lớn (Big Data) bao gồm hệ thống lưu trữ dữ liệu, hệ thống tiền xử lý, hệ thống trích xuất và chuẩn hóa dữ liệu, trực quan hóa dữ liệu, giải thuật xử lý dữ liệu, bộ làm sạch dữ liệu đầu cuối, mã hóa và bảo mật, hệ thống phân quyền truy cập, và hệ thống đầu cuối truy xuất dữ liệu kết quả;

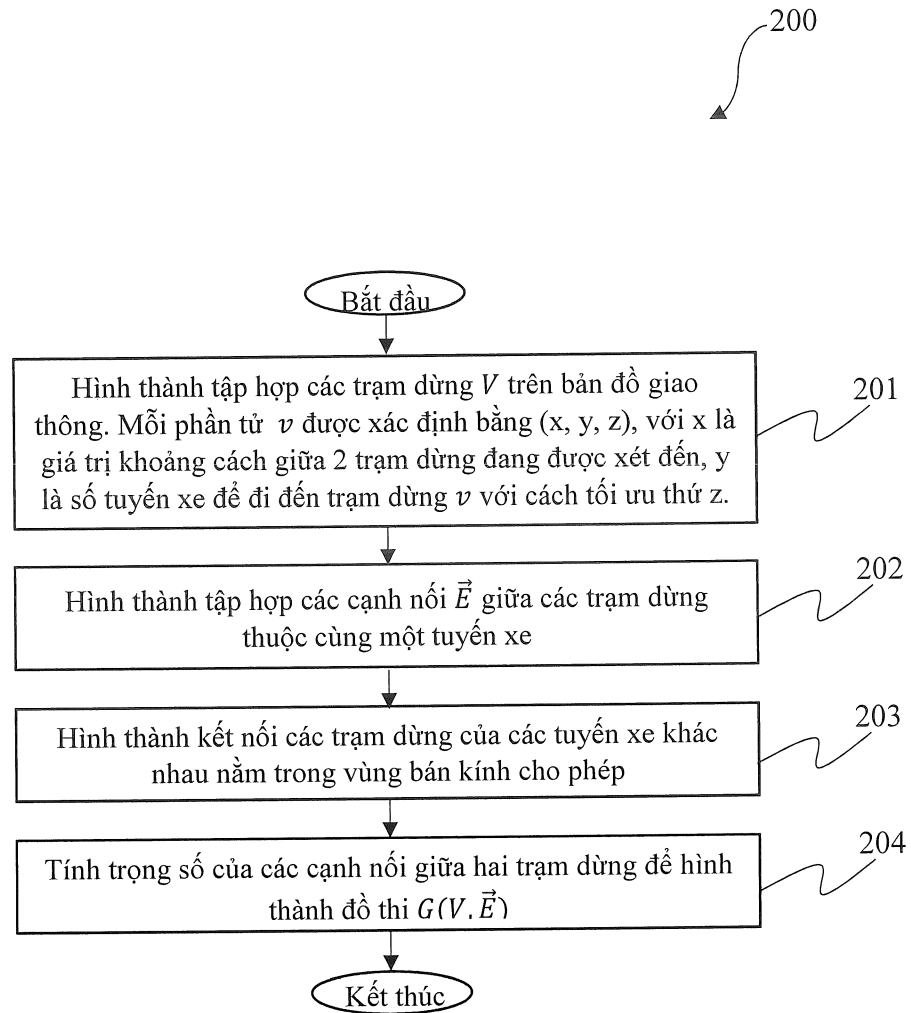
trong đó, cơ sở dữ liệu lớn liên tục nhận dữ liệu từ ứng dụng đề xuất lộ trình tối ưu Web/App; dữ liệu được tiền xử lý bởi hệ thống tiền xử lý, được lưu trữ bên trong hệ thống lưu trữ; dữ liệu được xử lý tiếp tục bởi giải thuật xử lý dữ liệu, dữ liệu tiếp tục được xử lý và làm sạch thông qua bộ làm sạch dữ liệu đầu cuối; hệ thống trích xuất và chuẩn hóa dữ

liệu làm giảm bớt sự dư thừa và loại bỏ những sự cố mâu thuẫn về dữ liệu, tiết kiệm được không gian lưu trữ; trực quan hóa dữ liệu biến các nguồn dữ liệu thành những thông tin được thể hiện một cách trực quan, dễ quan sát, dễ hiểu, để truyền đạt rõ ràng những hiểu biết đầy đủ từ dữ liệu.

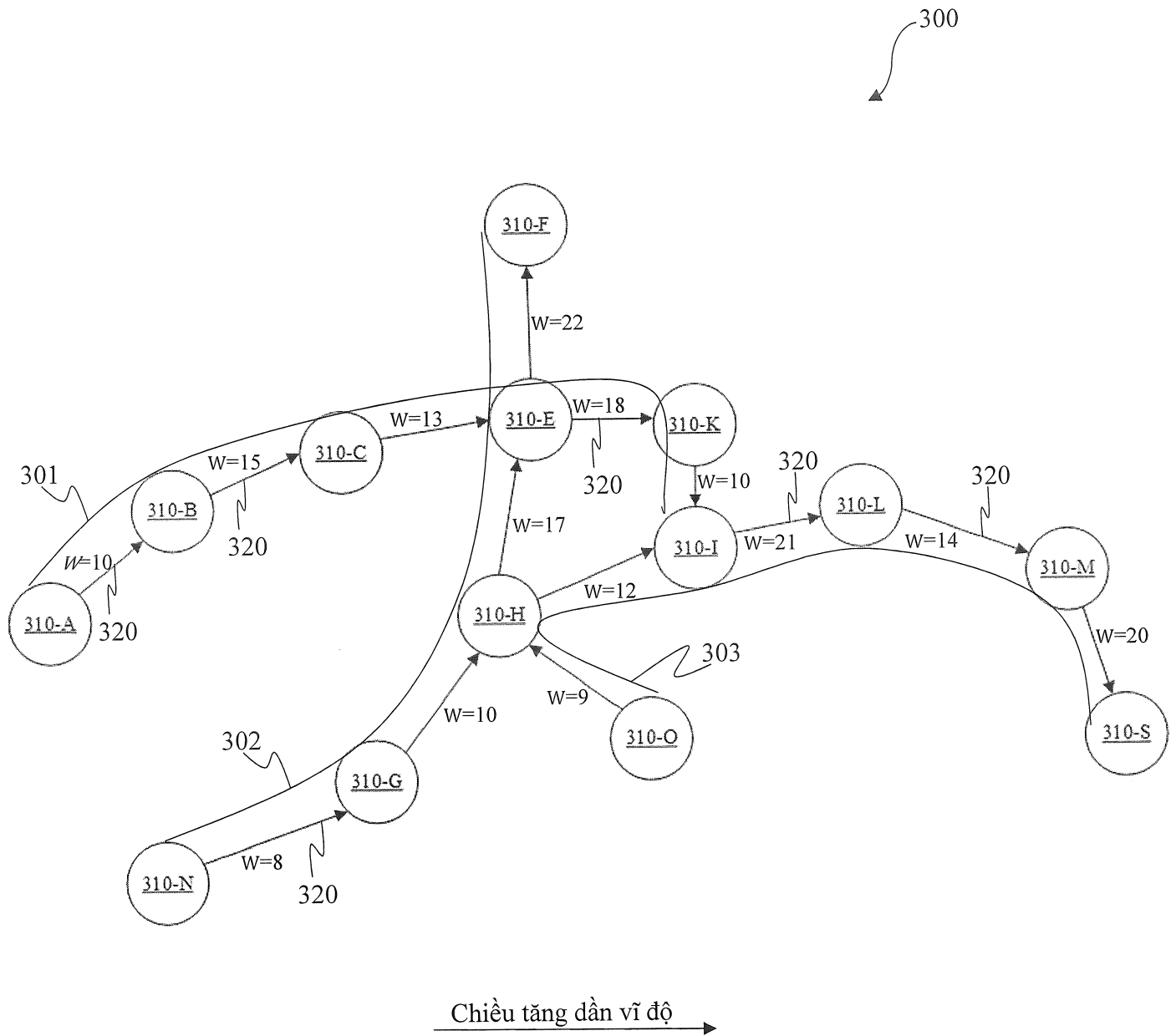
6. Hệ thống đề xuất lộ trình tối ưu cho người sử dụng phương tiện công cộng theo điểm 1, phương tiện công cộng bao gồm xe buýt, tàu điện, tàu điện ngầm, taxi, thuyền.



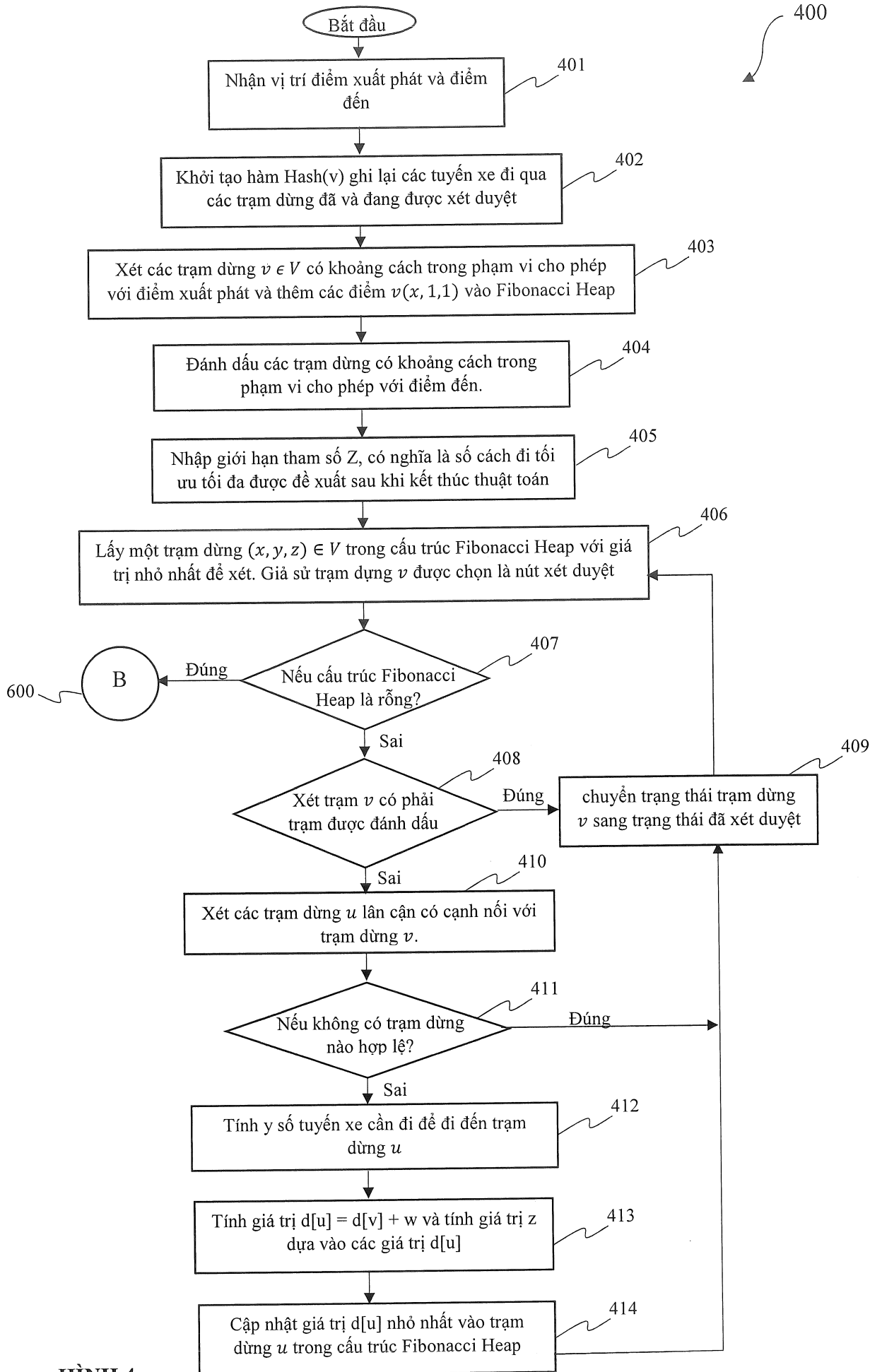
HÌNH 1



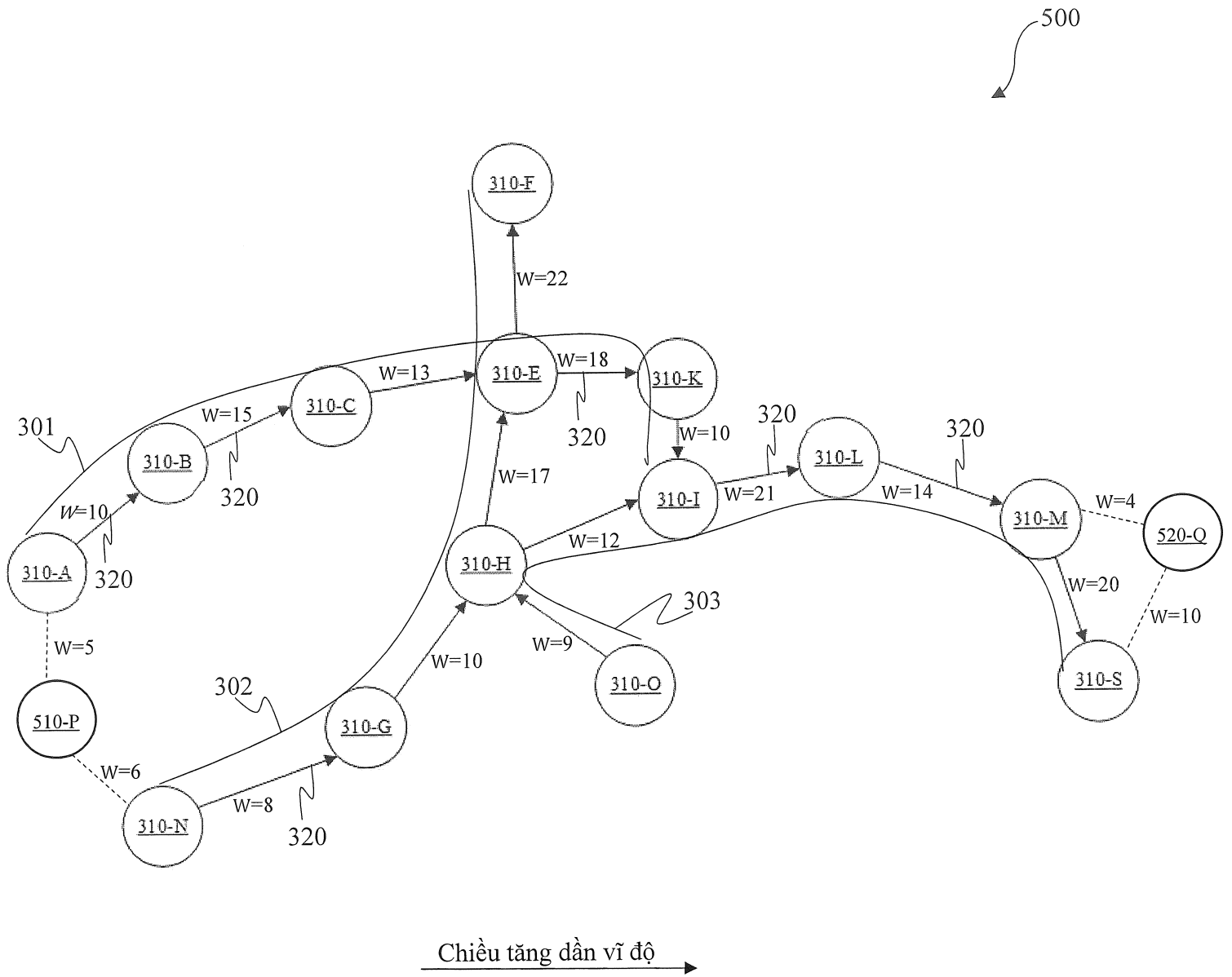
HÌNH 2



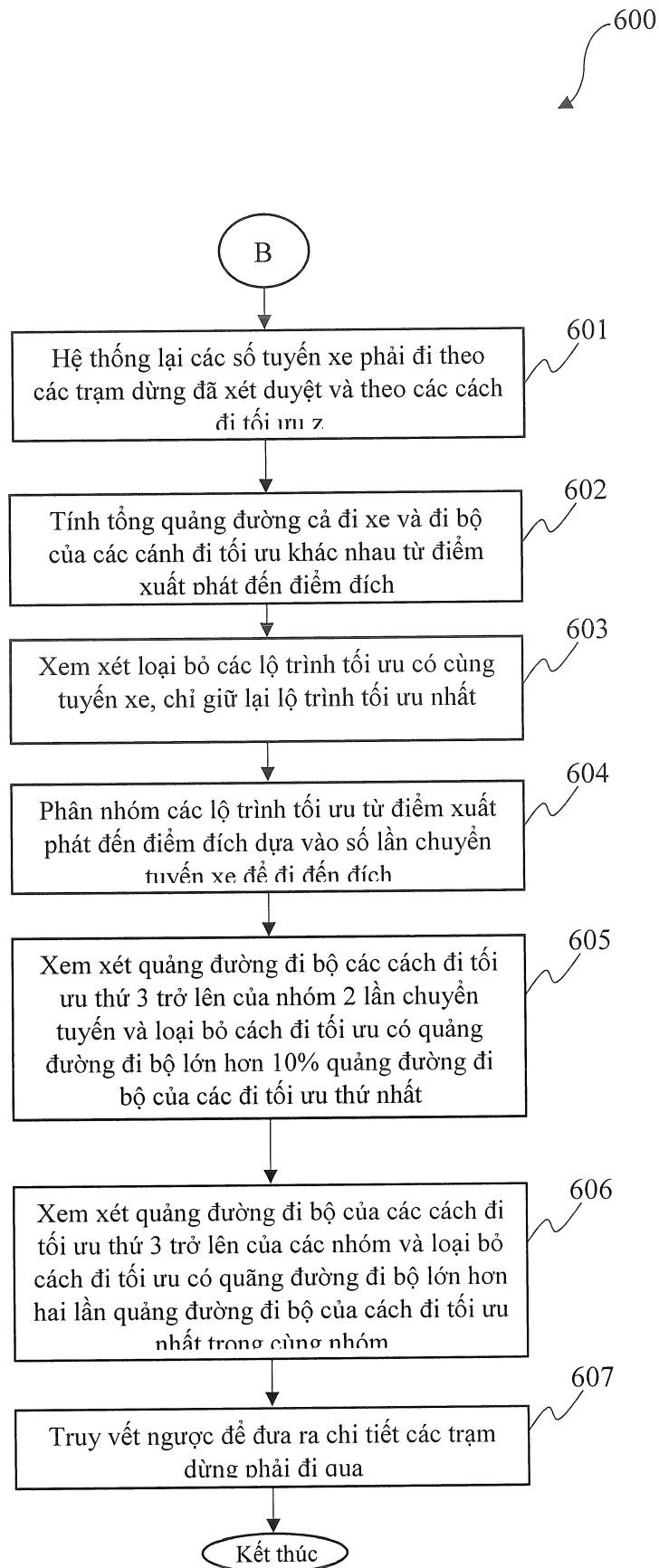
HÌNH 3



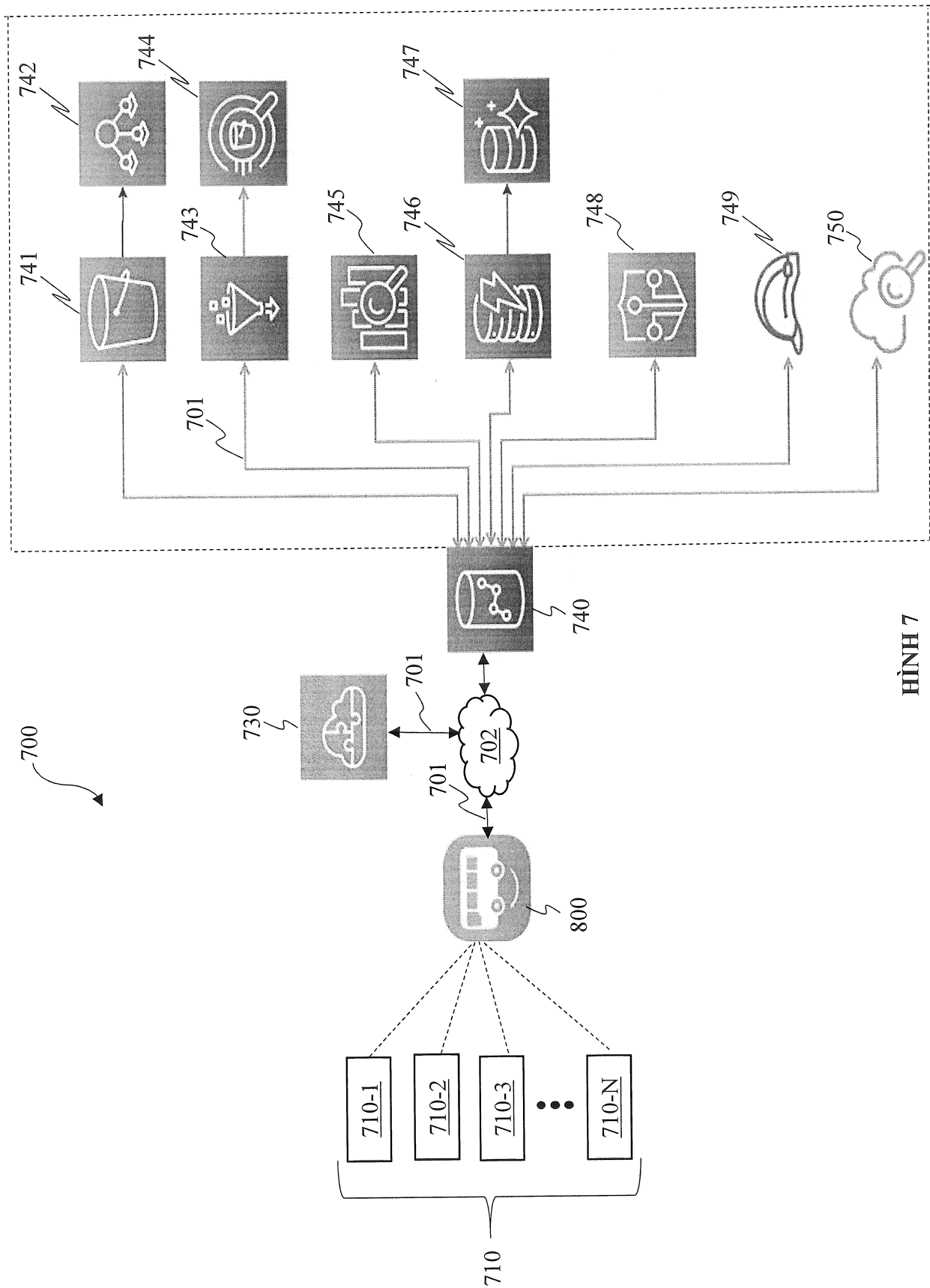
HÌNH 4



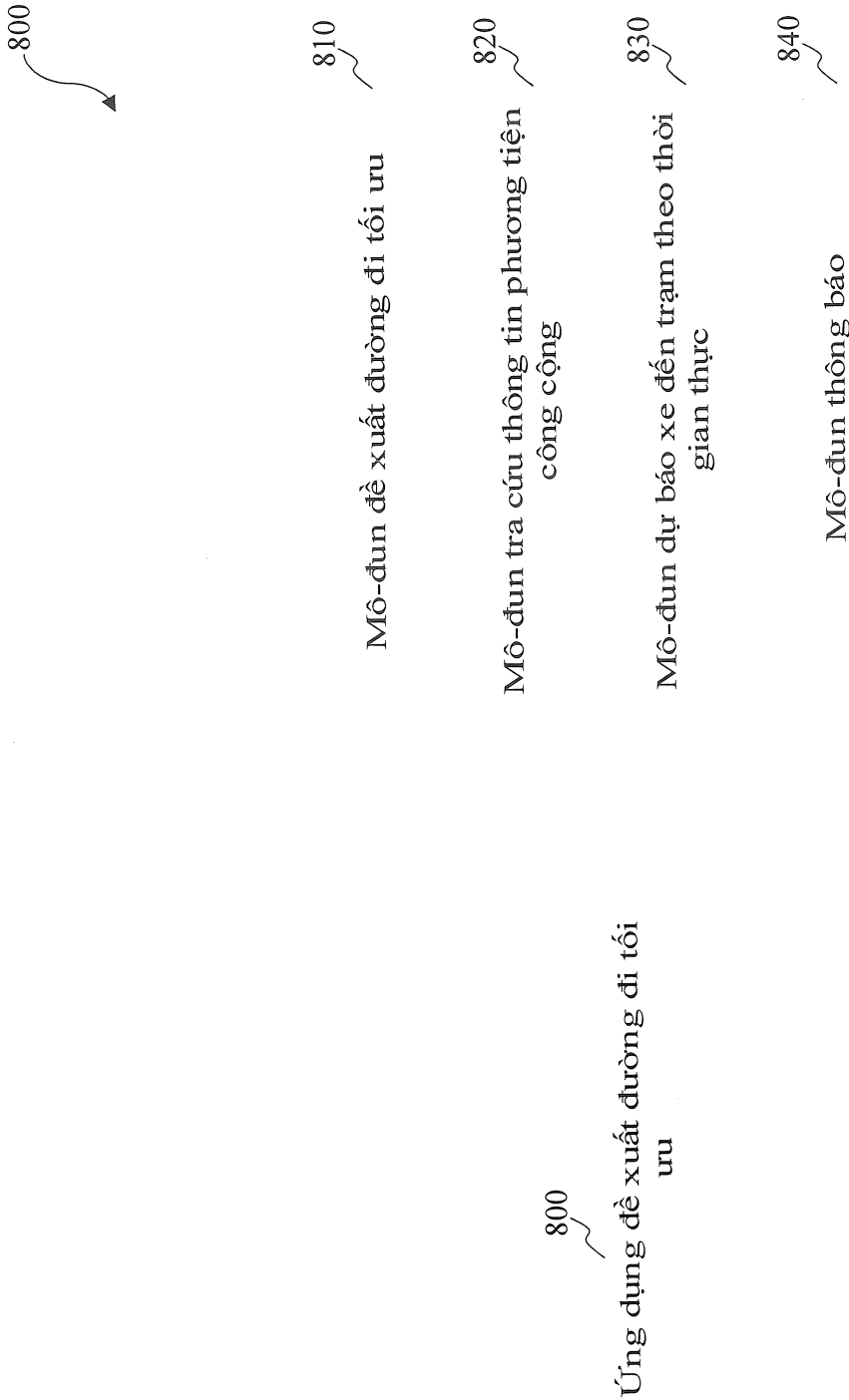
HÌNH 5



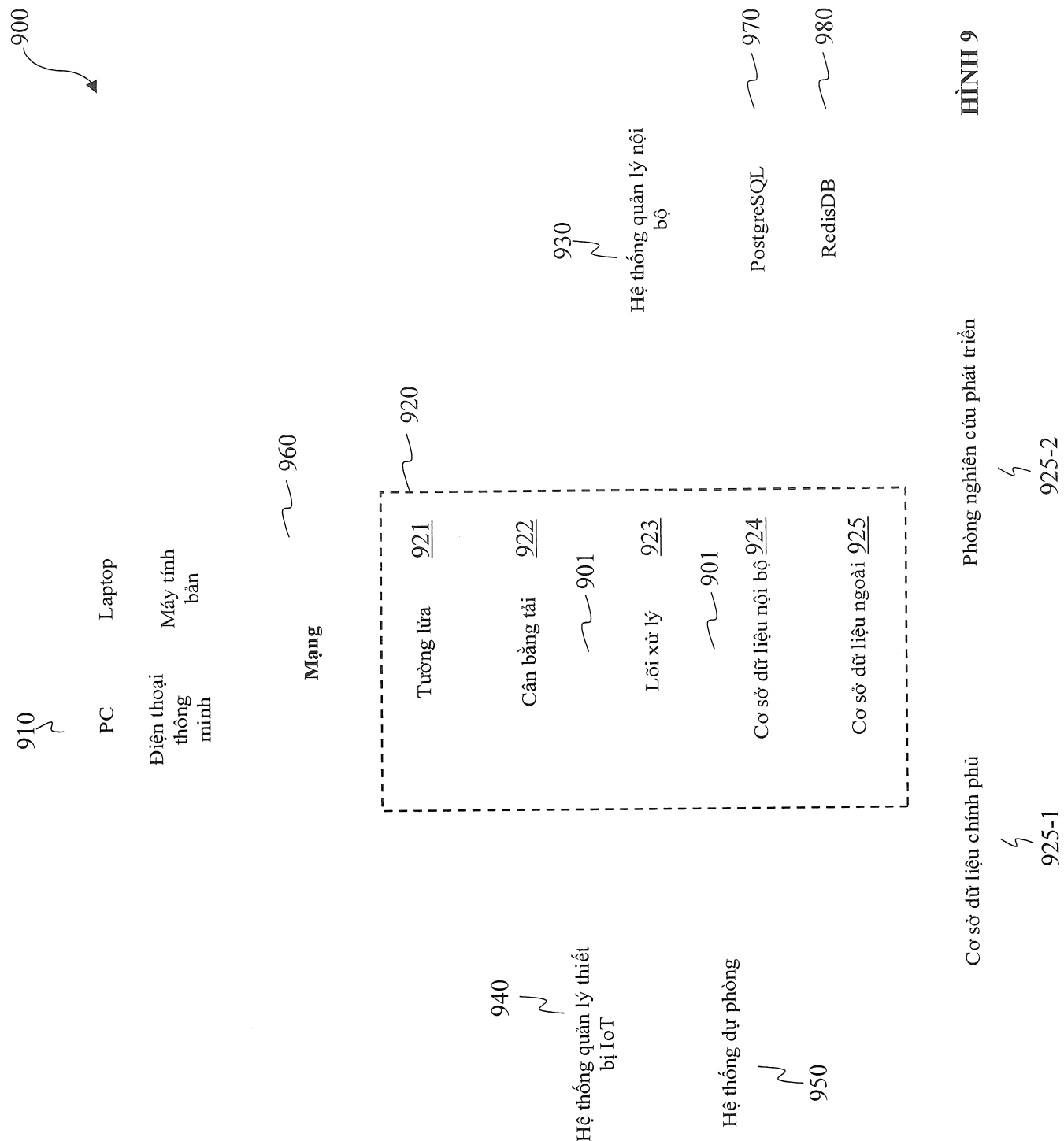
HÌNH 6



HÌNH 7



HÌNH 8



1010

1020

1030

1040

1000

10:54

Đi từ Công Viên Phú Lâm

Đến Công trường Đại học Quốc tế

Chuyến tới đã 3 chuyến

Tùy chỉnh

Xe buýt đường dài

Hồ Chí Minh → Đồng Nai

60,000đ

Các cách đi chuyển phù hợp

1011

14 → 08 162 m 28.5 km 82 phút (13K VND)

Xe tới trong 2 phút. Đón xe tại trạm Công viên Phú Lâm.

13 → 30 250 m 28.5 km 83 phút (13K VND)

Xe tới trong 2 phút. Đón xe tại trạm Công viên Phú Lâm.

1011

14 → 52 235 m 28.4 km 84 phút (12K VND)

Xe tới trong 2 phút. Đón xe tại trạm Công viên Phú Lâm.

14 → 10 162 m 27.6 km 79 phút (20K VND)

Xe tới trong 2 phút. Đón xe tại trạm Công viên Phú Lâm.

14 → 10 162 m 27.6 km 79 phút (20K VND)

Xe tới trong 3, 7, 9 phút

HÌNH 10A

10:55

Đi bộ: 162 m
Xe buýt: 28.5 km

CHI TIẾT CÁCH ĐI

Đón tuyến 14
Xuất phát từ Công viên Phú Lâm
Đón tuyến 14 tại trạm Công viên Phú Lâm
xe tới trong <1 phút

Đi tuyến xe 14
Công viên Phú Lâm → Siêu thị Coop Mart
(6K VND)

Chuyến từ tuyến 14 sang tuyến 08
Xưởng xe tại trạm Sầu thi Coop Mart
Đón tuyến 08 tại trạm Sầu thi Coop Mart
xe tới trong 3, 7, 9 phút

CÁC TRẠM ĐI QUA

PHƯỜNG 14
Lũy
Tân Hòa
Tân Hòa Đông
Bà Hom
PHƯỜNG 13
PHƯỜNG 12
Hậu Giang
Quận 6
Chợ Lớn

Đi bộ: 162 m
Xe buýt: 28.5 km

HÌNH 10B

10:55

Đi bộ: 162 m
Xe buýt: 28.5 km

CHI TIẾT CÁCH ĐI

Đón tuyến 14
Xuất phát từ Công viên Phú Lâm
Đón tuyến 14 tại trạm Công viên Phú Lâm
xe tới trong 1 phút

Đi tuyến xe 14
Công viên Phú Lâm → Siêu thị Coop Mart
(6K VND)

Chuyến từ tuyến 14 sang tuyến 08
Xưởng xe tại trạm Sầu thi Coop Mart
Đón tuyến 08 tại trạm Sầu thi Coop Mart
xe tới trong 3, 6, 10 phút

Đi tuyến xe 08
Siêu thị Coop Mart → Công trường Đại học Quốc tế
(7K VND)

Đi bộ tới điểm đến
Xưởng xe tại trạm Công trường Đại học Quốc tế
Đi bộ đến Công trường Đại học Quốc tế

CÁC TRẠM ĐI QUA

Đón tuyến 14
1 phút
161m

Đi tuyến xe 14
37 phút
13.2km

Chuyến từ tuyến 14 sang tuyến 08
43 phút
15.3km

HÌNH 10C

10:56

Đi bộ: 162 m
Xe buýt: 28.5 km

CHI TIẾT CÁCH ĐI

CÁC TRẠM ĐI QUA

14 Mạc Đình Chi

14 Đinh Tiên Hoàng

14 Thảo Cầm Viên

14 Cầu Thị Nghè

14 Nhà Thờ Thị Nghè

14 Trung tâm Đường Lãi

14 Trường trung học Gia Định

14 Ngã Ba Hàng Xanh

14 Dải Liệt sĩ

14 Siêu thị Coop Mart

08 Siêu thị Coop Mart

08 Cổng ra - Bến xe Miền Đông 3

08 Ngã tư Bình Triệu

08 Đường số 20

08 Chùa Lũ Đầm

08 Cã sấu Hoa cà

08 THCS Ngô Chí Quốc

08 Chùa An Lạc

08 Cầu Gò Nổi

HÌNH 10D

HÌNH 11A

HÌNH 11B

HÌNH 11C

HÌNH 11D