This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

Urban Structure Accessibility Modeling and Visualization for Joint Spatiotemporal Constraints

Farah Kamw, Shamal AL-Dohuki, Ye Zhao, Thomas Eynon, David Sheets, Jing Yang, Xinyue Ye, and Wei Chen

***Abstract*—In modern cities, service providers want to identify the regions that are hard to reach from multiple fire stations, a citizen wants to meet with friends in a restaurant close to everyone, and administrators want to find whether an area far from two bus stations needs a new one. Such tasks involve studying the dynamic accessibility of the urban structures over multiple geospatial and temporal constraints, which is an important topic in geographical sciences and urban transportation. In this paper, we present a new computational model and a visualization system that help domain users to interactively study the jointly constrained accessible regions, street segments, and Points of Interest (POIs). In particular, Urban Structure Accessibility Visualization system is built upon a new Min–Max Joint Set model, where specifically designed set operations not only represent the accessible regions but also compute the minimum and maximum access times to urban structures from the joint constraints. The computation and visualization are supported by a new graph model that accommodates the real-world dynamic traffic situation and the geographical settings of urban street segments and POIs. The visualization system allows the users to conveniently construct and manage accessible regions and visually explore the urban structures inside them.**

***Index Terms*—Urban accessibility, urban trajectories, visual analytics, geo-visualization.**

I. Introduction

URBAN accessibility under complex geospatial and temporal constraints is an important topic in many applications [22] of urban transportation. For example, it is necessary to find:

- Q1: Whether an urban area, when on fire, can be reached by fire engines coming from three different fire stations, given specific traffic condition and time period?
- Q2: For a school in this area, what is the earliest (minimum) time that the first fire truck from any station can arrive? and what is the latest maximum) time that trucks from all three stations can arrive?

Our aim is to quickly find the answers through visualization which would be greatly helpful in transportation study and urban planning. However, these seemingly simple questions are indeed challenging because: (1) they involve *joint* spatial (e.g., hospitals) and temporal (e.g., 5 minutes) constraints with real traffic conditions at specific time periods (e.g., rush hour) (2) they require to find maximum and/or minimum access time to POIs/street segments from multiple seed locations. Moreover, effective geographical visual representations are needed to convey the information. Unfortunately, these questions cannot be easily answered by existing tools. For example, the popular isochrone map only calculates and shows the reachable region based on single location (e.g., [11]). Simply drawing multiple isochrones on the map can show the overlap, but users cannot discern the combinatorial information of inside POIs, especially the maximum or minimum access time from multiple locations. As a consequence, there is an urgent need for an intelligent system that integrates new computational models and visualizations to study the complex accessibility of urban structures. After performing a requirement analysis with several domain experts, we identified:

- First, the system should easily discover accessible urban regions from multiple starting locations at different time periods. These regions can be defined based on *joint conditions* of the spatio-temporal constraints, such as the reachable region from two fire stations *A and B*, *A or B*, *A but not B*.
- Second, users should be able to visually study the minimum and maximum access times to urban structures inside these regions. The system should compute and visualize their minimum and maximum access times by the joint conditions. For example, it should allow users to identify and compare the access times to different POIs from the two fire stations
- Third, an easy-to-use visual interface should allow users to interactively set up and combine spatial-temporal constraints, quickly get visual responses of the accessible regions and structures, and iteratively modify the constraints based on feedbacks to their previous exploration.

In this paper, to fulfill these requirements, we propose an intelligent system, named as USAVis (Urban Structure

Manuscript received August 29, 2018; revised December 13, 2018; accepted December 14, 2018. This work was supported in part by the U.S. NSF under Grant 1535031, Grant 1535081, and Grant 1739491. The work of W. Chen was supported by NSFC under Grant U1609217 and Grant 61772456. The Associate Editor for this paper was W. Lin. *(Corresponding author: Ye Zhao.)*

F. Kamw, S. AL-Dohuki, Y. Zhao, T. Eynon, and D. Sheets are with the Department of Computer Science, Kent State University, Kent, OH 44242 USA (e-mail: zhao@cs.kent.edu).

J. Yang is with the Department of Computer Science, University of North Carolina at Charlotte, Charlotte, NC 28223 USA.

X. Ye is with the Department of Informatics, New Jersey Institute of Technology, Newark, NJ 07102 USA.

W. Chen is with the State Key Lab of CAD&CG, Zhejiang University, Hangzhou 310058, China (e-mail: zhao@cs.kent.edu).

Digital Object Identifier 10.1109/TITS.2018.2888994

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

Accessibility Visualization). It models and visualizes accessible regions (and their POIs/street segments) satisfying joint conditions with given traffic conditions. This system is built up on a new Min-Max Joint Set model, referred as MinMaxJS. This model uses special set of accessible POIs/streets to represent an accessible region (we call it Traffic Region) and compute minimum/maximum access times from multiple seed locations. First, users can define Primary Traffic Regions (PTRs) according to individual spatial-temporal constraints. A PTR covers an isochrone (reachable) region from one seed/source location. The region is modeled as a MinMaxJS set of the street segments and POIs inside it. Each of these elements is given a characteristic value as the access time from the seed location. Second, users can build a Constructive Traffic Region (CTR) by flexibly combining multiple PTRs through joint MinMaxJS operations, including *Union Max*, *Union Min*, *Intersection Max*, *Intersection Min*, and *Difference*. Unlike traditional set operations, MinMaxJS operations are designed to compute the minimum and maximum access times when performing intersections or unions, specifically considering multiple seed locations.

The MinMaxJS computations are enabled by a USAGraph (Urban Structure Accessibility Graph) connecting POIs and street segments through transportation infrastructure. USAGraph is created as a dual road network of the street segments in a city. These street segments are improved by a specified geometric processing algorithm. Then, POIs are added to the USAGraph using a new POI-to-Street projection. Moreover, the graph weights are defined by dynamic and real traffic situations. Our current system acquires the real traffic data from taxi trajectory datasets, which however, can also be provided by other sources. The graph traversal of USAGraphs facilitates fast computation of the Traffic Regions.

We further develop USAVis system to visually manage and study the reachable regions, POIs, and street segments with a variety of diagrams and charts. The reachable regions and access times are visualized and compared on the map with geographic context. The users can interactively perform MinMaxJS operations as well as drill-down study of individual regions and structures.

In summary, the main contributions of this paper include:

- We present the first visual analysis system, USAVis, to the best of our knowledge, that allows users to study urban structures accessibility over single and joint geo-spatial and temporal constraints. A set of new visualization algorithms are introduced such as region drawing method based on concave hull and a coloring scheme for enhanced accessibility visualization.
- We propose a new computational model, MinMaxJS, to represent and compute jointly reachable regions and their POIs. We design new Min-Max set operations which naturally compute different types of access times from joint spatio-temporal constraints.
- A new graph model, USAGraph, integrates POIs and an improved dual road network with real world traffic information. It facilitates very fast, graph-based computation of urban accessibility.

We have conducted user study and domain expert interviews to demonstrate how our model and visual system advance the study of urban accessibility.

II. Related Work

A. Urban Accessibility Study

Studying the accessibility of city structures, which is a product of combining mobility and proximity, is an important work to understand transport and urban form [27]. A variety of works in urban research have been conducted to study the role of accessibility in cities (see a review in [22] and papers therein). For example, two GIS-based accessibility methods are integrated to examine the spatial accessibility to primary health care in Chicago region [20]. A dynamic accessibility measurement is proposed based on real-time travel speeds to compute the accessibility to POIs [17]. A doctoral thesis [31] shows how the accessibility can be affected by urban design of different public transit scenarios. Most of existing studies are focused on computing accessible region according to individual spatial-temporal constraints. Neutens and Versichele [23] have concentrated on the analysis of the spatio-temporal constraints that circumscribed the POIs that are accessible to one person or group of people willing to engage in one activity. However, while acknowledging their contributions, this work extracts reachable POIs for group of people by using only the regular intersection set operation. In contrast, our approach uses the new MinMaxJS operations that help not only to find reachable region from multiple starting locations, but also to study the access time (with different types) to urban structures inside the region.

B. Graph-Based Methods and Isochrone Map

Graph-based methods have been used in modeling road networks with primal [6], dual [29], and multi-granular representations [15]. Graphs are used in studying land use, in which roads, parcels and buildings are integrated [7]. Traffic information is integrated into the network [2], which focuses on congestion influence analysis. Isochrone map defines the reachable region within a given travel time in traffic [3], [8], [11]. Some approaches improve isochrone computation over transport networks in geospatial databases [3], [9].

Isochrones are visualized as colorful regions together with geographical attributes [8]. Isoscope [10] provides good appearance of isochrone visualization based on map services. Isochrone map is used to visualize the accessible areas for passengers on public transportation system [35]. Traffigram [12] designs an interactive system where isochronal map is combined with geospatial context and travel conditions. Unlike our method, These approaches do not consider the flexible operations on these isochronic regions to study accessibility from combined multiple conditions.

C. Urban Trajectory Data and Visual Analytics

Data driven study has led to advanced technologies in intelligent transportation systems (e.g., [26]). The human/vehicle trajectory data such as in crowd scenes and intersections are recognized and modeled [18], [19], [34], [37]) to discover their behavior patterns. Sparse human trajectory data is also extracted and analyzed from geo-tagged social media data [4]. Web-based city POI data is used to enrich trajectory data with semantic information [16]). These approaches can be used to give more comprehensive and accurate trajectory representation for our accessibility computation.

Visual analytics of trajectory data has contributed to many urban computing applications as shown in several surveys [1], [5], [21]. GeoDec [30] and SemanticTraj [38] allow users to effectively query geospatial and trajectory data. FromDaDy [14] visualizes airplane trajectories where users can extract relevant trajectories with Boolean queries. Our system does not focus on querying trajectories themselves. Instead, we extract traffic information from trajectory dataset to study accessibility of urban structures under multiple constraints using newly designed MinMaxJS operations. A visual analytics system [36] is developed to explore the relationship between city POIs and human mobility from massive public transportation data. Unlike our system, this approach is not designed for multiple constraints of accessible regions. Reachability query is answered efficiently by specifically designed indexing and query algorithms [33], which does not find access time to different POIs inside the reachable region like our work.

Our approach utilizes graph model to compute accessibility. A TrajGraph model created from taxi trajectories is used to visually analyze salient locations and streets in traffic by applying centrality metrics [13]. However, this method does not consider POIs and accessibility in their analysis. A similar trajectory graph model [32] is used to assist people in finding a home based on multiple criteria including reachability. However, this approach does not perform accessibility computation based on the joint spatial conditions which is our focus. Moreover, when mapping POI to the road network, this method simply finds the closest road segment. In our work, we use a new one-to-multiple projection to project a POI to a set of close road segments, which provides more accurate computing of accessibility. In general, we believe the graph based model will become a powerful tool in visual analysis of urban data.

III. USAVis System Overview

A. Design Aims

We have interviewed three urban planners and geographers for the general requirements of studying joint accessibility of urban structures. Based on the requirement analysis, three major goals of USAVis are set as:

- *Interactive exploration with joint constraints:* The visual exploration of urban accessibility should be interactive and iterative for domain users. Users should flexibly find accessible region and related POIs with respect to different seed locations and time periods. They should also be able to modify, adjust, and delete the constraints based on instant visual feedbacks for the effects of multiple constraints during the exploration.
- *Flexible management of accessible regions:* The system should allow users to directly manage and compare multiple accessible regions they generated. Users should be able to show, hide, remove, compare these regions on the map. Access times of interesting POIs under different constraints should be easily depicted and compared.
- *Integration with urban context:* The visual study should be fully integrated with the urban context through the map view. Information about regional characteristics, POIs and traffic information should be provided.

B. Visual System Functions

To achieve these goals, USAVis is designed based on the MinMaxJS model, so that users can define and examine PTRs and CTRs. Figure 1 shows the visual interface which includes:

- A *canvas* over city map, Figure 1(A), to facilitate visualizations and operations of PTRs and CTRs. The map can be shown in different styles and support smooth zooming and panning. The map view shows the regions and the access times to the road segments and POIs inside them according to different joint conditions.
- A *PTR configuration panel* to flexibly create PTRs with different methods (Figure 1(B)). Users can define parameters in the panel to create desired PTRs.
- A *CTR control panel* shown in Figure 1(C) to generate and manage CTRs with easy access to MinMaxJS operations and visualization parameters.
- A *POI panel* to study a variety of categories and the details of POIs inside an active region by a ranked list of their access times from the seed of the region (Figure 1(D)).
- A *visual report view* to display region characteristics and enable quantitative comparison. This view is popped out to display after clicking the button (Figure 1(E)).

C. Usage Scenario

For an urban planner (Zhang) who wanted to examine the accessible regional information from two hospitals (Sir Run Run Shaw hospital and First Affiliated hospital) in the city of Hangzhou, China. As shown in Figure 1, Zhang selected a time period 6-8am of a specific day (Figure 1(B)), and then marked the two hospitals as *seed* locations on the map. Two PTRs were computed and shown in orange and green on the map, respectively. Each of them represented the region that an ambulance can reach in 6 minutes from one hospital, computed from real traffic information. They were named as Region1 and Region2 in the control panel (Figure 1(C)). Then, a MinMaxJS "Umin" operation (Eqn. 5) is applied to create a CTR Region3 which covers both of them. Zhang observed this region which could be reached within 6 minutes from either one of the hospitals. The joint accessibility reflects the fastest time an object can be reached from either one of the two hospitals (such as by ambulances). A list of POI categories was shown in Figure 1(D). By selecting *Education*, Zhang investigated POIs of this category which were shown as dots on map. Their color and size reflect the minimum time

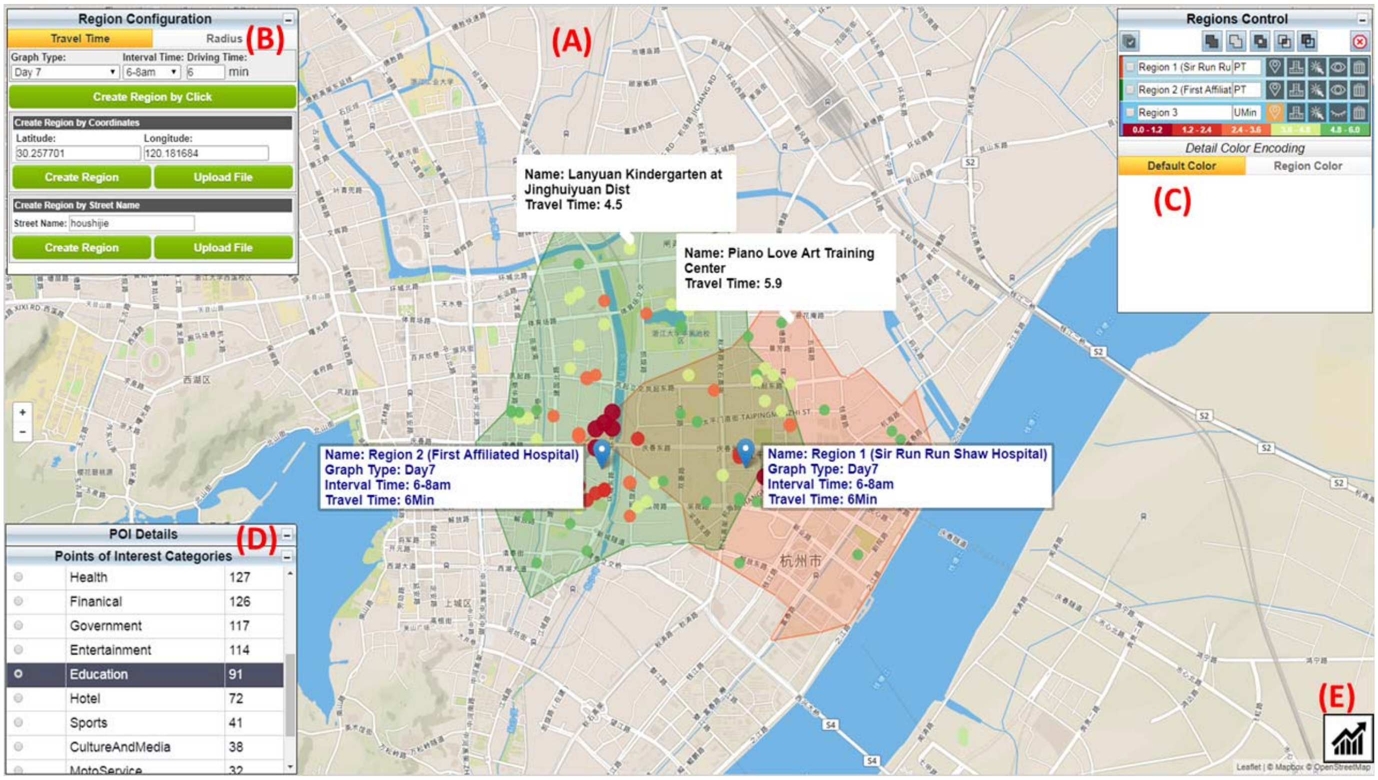


Fig. 1. Visual interface of the USAVis system. Two PTRs (orange and green) represent the isochronic PTR regions within 6 minutes of driving in the morning (6-8am) from two hospitals. They are combined to create a CTR, in which the POIs in a category (Education) are shown as colored dots. The minimum time needed for an ambulance to reach the POI from either one of the hospitals is represented by varied dot colors and sizes.

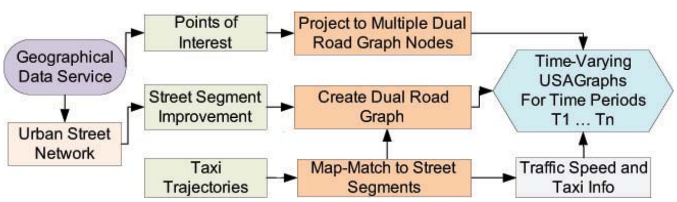


Fig. 2. The flowchart of creating USAGraphs to manage traffic, taxi and POI data and support fast construction of reachable regions. T1 to Tn are the time periods such as each hour of a day.

needed for an ambulance to reach the POI from either one of the hospitals, where red indicates shorter time than green. Zhang could further study these POIs by hovering mouse over them. Two POIs are labeled in black fonts: one (top left) is a kindergarten center which is reachable in 4.5 minutes, and another one (top right) is an art training center which needs 5.9 minutes. It can be realized that although the art training center is closer to the hospitals, it needs more time to reach. Clicking Figure 1(E) would allow Zhang to further compare these regions.

IV. Urban Structure Accessibility Graph

USAGraphs integrate real traffic information, urban road network, POIs, and taxi trip information into time-varying graph models. Figure 2 shows the process of creating USAGraphs. A USAGraph is built by (1) improving street segments retrieved from geographical data services, such as OpenStreetMap; (2) creating a dual road graph from the improved street segments and the taxi trajectory data; (3) adding POIs to the graph nodes through a one-to-multiple projection; (4) defining graph weights by traffic speeds computed from big taxi trajectory data. Then, a series of USAGraphs of different time periods are managed in a graph database for fast graph traversal to support the PTR/CTR operations.

A. Creating Improved Dual Road Graph

A dual road graph is created by mapping street segments to graph nodes and street inter-connections to graph edges. A graph edge $\overrightarrow{AB}$ means vehicles can drive from street segment A to B. This dual graph can naturally model complex road intersections reflecting traffic flow directions. In comparison, original (primal) geometric road network is awkward in modeling complex, multiple-layered road crossings which need to be represented as graph nodes [24].

1) Improving Street Segments: In our practice, we find that the predefined street segments may go across a few road intersections, may be a complex geometric shape, and may have errors. Therefore, the generated graph cannot create good traffic regions (PTR/CTR). We then design a special re-segmentation algorithm of street segments. The goal is to make the end points of street segments reside at road intersections, so that the dual graph can be directly achieved by mapping the segments to nodes and the connected segments are linked by edges. The algorithm has three steps:

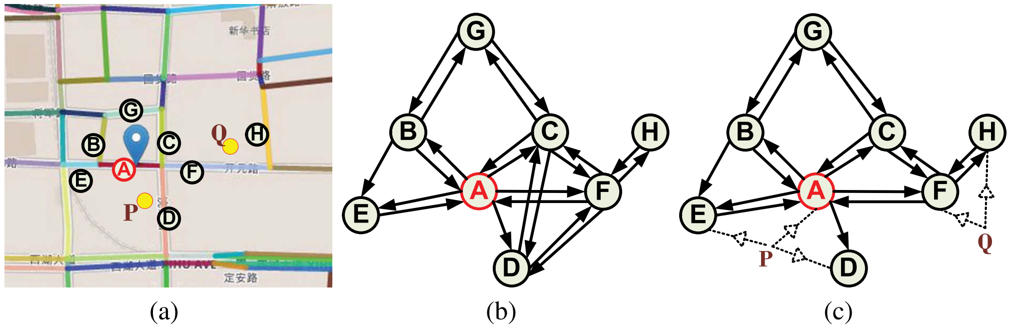


Fig. 3. Road network, POIs, and USAGraph: (a) The improved urban street segments A to H. Two POIs, P and Q, are shown. (b) Mapping the segments to nodes of a dual road graph. (c) Projecting each POI to multiple graph nodes.

(1) Identifying all intersection points among all given segments. (2) Assigning each part between any two intersection points as one new segment. (3) Dividing a long new segment into smaller ones with a predefined threshold such as 300 meters to ensure accurate representation. Figure 3a shows the generated street segments in a small part of a city. Each segment such as A to H are displayed in different colors.

2) Building Dual Road Graph Based on Trajectories: From the newly generated street segments, we create a dual road graph by using the graph nodes to represent the street segments, and the graph edges to represent interconnections between neighboring segments. First, the graph is an undirected graph with one edge AB between each pair of nodes A and B, since we do not know yet whether cars can go from A to B and/or B to A. Such traffic information is retrieved from the massive taxi trajectory data. We follow consecutive GPS points on each trajectory and map them to street segments. If a taxi travels from A to B, we add the directed edge $\overrightarrow{AB}$. Then, a directed dual graph is formed. Figure 3b shows a subgraph where each node represents a street segment (A to H) in Figure 3a. There is no edge $\overrightarrow{EB}$ and $\overrightarrow{DA}$ since the left turns from E to B and from D to A are prohibited.

B. Generating USAGraphs

The dual graph only stores streets and traffic information. For computing accessibility, first we need to add city POIs to the dual graph. Weng *et al.* [32] project one POI to its closest street segment. However this straightforward approach may not give an accurate accessibility computation. As shown in Figure 3a, a POI P is in a block close to street segments A, D and E. When computing the accessible time from P to Q, starting from D (which is the closest) may reach Q in a longer time than starting from A, according to real-time traffic. Therefore, the one-to-one mapping from POI to street segment should be replaced by a one-to-multiple projection. In implementation, we set a threshold distance. When a POI has a shorter distance to a street segment than this threshold, then this POI is added to the dual road graph node of this segment. Figure 3c shows that P is added to three nodes (A, D, and E) and Q is added to two (F, and H). We call the dual road graph model where each node has multiple POIs as the USAGraph. Then street information like street segments length are added to graph nodes. In implementation, USAGraphs store and manage multiple types of city data at their nodes including POIs with categories, the numbers of pickup/dropoffs of taxi trips, and the average travel speeds.

For T1 to Tn, a sequence of USAGraphs with different weights are generated for different time periods. For example, 24 graphs are generated for each day, each graph accommodates the traffic information at a specific hour of a day. The time intervals can be changed to increase temporal accuracy. In addition to the hourly USAGraphs, the daily, weekly and monthly graphs are also generated so that users can analyze dynamical traffic information in a week, a month, or in particular days. USAGraphs are managed by a graph database. USAVis allows users to choose these different USAGraphs for their analytical tasks. The PTR generation is completed through a fast graph traversal algorithm supported by the database.

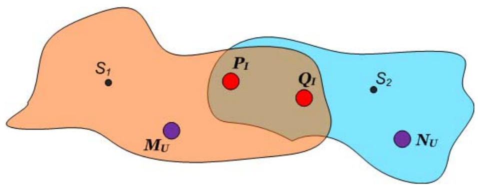
C. PTR Generation With USAGraph Traversal

Typically, a PTR is an isochronic region created from real traffic information given a seed location, S. It depicts the area where vehicles (or walkers) at S can reach in t minutes at a specific time period T. A PTR is defined as

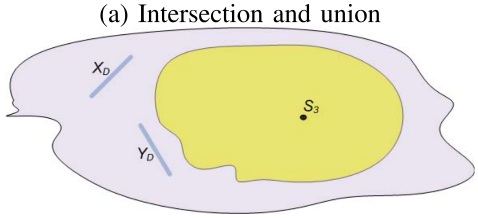
$$\pi(S, t, T) \mapsto R, \tag{1}$$

where R is the result PTR set, $T \in \{T1, T2, \ldots, Tn\}$ is the time period in a day, and t is the time length of travel. In our method, this map function π is implemented as a graph traversal process. First, the USAGraphs created from a day, a week or a month is selected according to user's interest. Then, the particular graph G of period T is retrieved. Next, a seed S (e.g., a POI) may close to multiple nodes in G. Starting from these nodes, we apply the BFS (Breadth-First Search) algorithm to traverse all possible paths in G. On each node over the paths, the travel time is computed using the stored travel speed and the length of this segment. Accumulating the travel time along each path, the BFS continues when the travel time on a path is smaller than t. Once BFS stops, we find all nodes (i.e. street segments and connected POIs) of R. For example, in Figure 3, assuming a seed POI P is selected, then the algorithm traverses the USAGraph from A, D and E to find all urban structures can be reached in the given time length. Each of them is also given the access time, τ, from P to itself.

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.



(a) Intersection and union



(b) Difference

Fig. 4. Creating a CTR from two PTRs with our specially-designed MinMaxJS operations.

In this algorithm, one street segment is either in or out of a PTR R, which implies that the region's accuracy is limited by the length of segments (e.g., a few hundreds of meters). To increase the accuracy, long segments can be further divided into smaller ones, while the algorithm is still the same.

V. Min-Max Joint Set Model and CTR Generation

A PTR, R, consists of a discrete set whose elements are reachable urban structures (street segments and POIs). Each element has a characteristic value τ of access time, so that this set is not a classic set where an element only has 1 (exist) or 0 (absent) status. Then users can construct CTRs over multiple PTRs. Here the newly designed MinMaxJS operations of union, intersection and difference are used not only to model different types of CTRs but also to compute minimum or maximum access times from multiple seed locations.

A. MinMaxJS Intersection

Assuming R_1 and R_2 are the PTRs that fire trucks can reach in 5 minutes from two fire stations S_1 and S_2, respectively, as shown in Figure 4a with R_1 in orange and R_2 in blue. A CTR C_I from their intersection represents the region that can be reached in 5 minutes by fire trucks from *both* S_1 and S_2 including POIs like P_I and Q_I. The novel contribution of our approach is: we further allow users to utilize two formulas in the computation of τ_{C_I}. First, a member element in C_I has the minimum of their values in R_1 and R_2:

$$IMin : \tau_{C_I = R_1 \cap R_2} = min(\tau_{S_1}, \tau_{S_2}). \tag{2}$$

Here a member element with a small value, τ_{C_I}, means it can be quickly reached by fire trucks from *either* S_1 or S_2.

Figure 4a illustrates the algorithm with an example. For a simple explanation here, we assume that Euclidean distance determines the access time (e.g., $|P_I S_1|$ determines τ_{S_1} for P_I). In Figure 4a, Q_I has a smaller combined τ_{C_I} than P_I because $min(|Q_I S_1|, |Q_I S_2|) < min(|P_I S_1|, |P_I S_2|)$. It means that no matter the first arrived truck coming from which station, Q_I can be reached quicker than P_I (in this case from S_2). On the other hand, we can also assign the maximum of the values in R_1 and R_2 to τ_{C_I}:

$$IMax : \tau_{C_I = R_1 \cap R_2} = max(\tau_{S_1}, \tau_{S_2}). \tag{3}$$

Then, a member element with a small result value means it can be quickly reached by fire trucks from *both* S_1 and S_2. In Figure 4a, P_I has a smaller combined τ_{C_I} than Q_I, since $max(|P_I S_1|, |P_I S_2|) < max(|Q_I S_1|, |Q_I S_2|)$. This indicates that if we consider the time that fire trucks from both stations arrive, then P_I can be accessed faster than Q_I. This is because Q_I needs more time for fire trucks from S_1 to arrive.

B. MinMaxJS Union

The CTR, C_U, computed from union operation includes all member elements in either R_1 or R_2. In Figure 4a, it includes all the parts of the blue and orange regions. This CTR includes the street segment and POIs that can be reached by fire trucks from *either* S_1 or S_2 in 5 minutes, such as P_I, Q_I, M_U, N_U. For studying the union region, we first need to compute τ_{S_1} and τ_{S_2} for all members in C_U, since some members (e.g., N_U) do not have $\tau(S_1)$ values before, and M_U does not have $\tau(S_2)$ before. This is implemented by applying the BFS algorithms in an induced sub-graph that includes all members of C_U. Then we provide two ways of computing the new $\tau_{C_U = R_1 \cup R_2}$. First, τ_{C_U} can be computed as:

$$UMax : \tau_{C_U = R_1 \cup R_2} = max(\tau_{S_1}, \tau_{S_2}). \tag{4}$$

A member having a small result value means that the slowest truck from *both* S_1 and S_2 can reach this street faster than others. In Figure 4a, M_U has a smaller τ_{C_U} than N_U, because $max(|M_U S_1|, |M_U S_2|) < max(|N_U S_1|, |N_U S_2|)$. M_U can be accessed faster than N_U for trucks from both stations as N_U has a long time waiting for trucks from S_1.

Second, τ_{C_U} could also be computed as:

$$UMin : \tau_{C_U = R_1 \cup R_2} = min(\tau_{S_1}, \tau_{S_2}). \tag{5}$$

In this way, a member element has a smaller result value means that it can be reached by the quickest fire truck from *either* S_1 or S_2 easily than others. In Figure 4a, N_U has a smaller combined τ than M_U for $min(|N_U S_1|, |N_U S_2|) < min(|M_U S_1|, |M_U S_2|)$. This means that compared with M_U, N_U can be accessed first by a truck (from S_2).

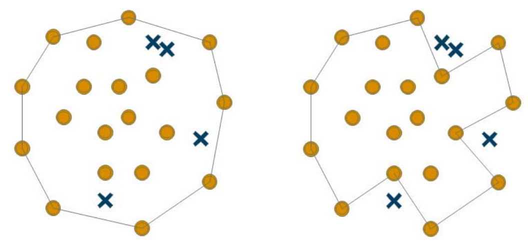
C. MinMaxJS Difference

Computing a CTR as set difference $R_2 - R_1$ keeps the member elements in R_2 but not in R_1. A member in the CTR means that it can be reached from S_2 but not from S_1. Then the new τ_{C_D} is simply computed as

$$Diff : \tau_{C_{D = R_2 - R_1}} = \tau_{S_2}. \tag{6}$$

In some cases users want to use two PTRs from the same seed S. For an example illustrated in Figure 4b, R_1 (yellow) and R_2 (purple) are the regions that people can walk to a subway station S_3 in 20 minutes and 40 minutes, respectively. Note that we can apply a constant walking speed on streets to create such PTRs. A CTR can immediately find those urban

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.



(a) Convex hull (b) Concave hull

Fig. 5. Drawing a region as a specific concave hull from the points of a set of street segments. A concave hull is created by removing area covering other segments not belonging to this region.

structures reachable between 20 to 40 minutes. Users can study the numbers of taxi pickups happen in the members of this CTR to find two street segments X_D and Y_D as candidate locations of new bus/subway stations.

VI. Interactive Visualization

USAVis system allows users to visually construct, manage and investigate PTRs and CTRs. Next we discuss the functions in USAVis.

A. Creating PTRs

Users can conveniently define the parameters, T, S, t, in Equation 1 to create PTR regions as shown in Figure 1(B). First, users flexibly define a PTR seed, S, in different ways by (1) clicking on the map; (2) giving street/POI names; and (3) loading geo-locations with longitude and latitude. Users can also load a file to define multiple seeds for batch processing. For some cases, users can also choose a region having a fixed radius from S. Second, users select time interval T and driving time t. Moreover, users can choose different graph types for the traffic information of one specific day, of one week, or of one month. In this way, the created PTRs show immediately on the map.

B. Constructing CTR and Managing Regions

Icons in the control panel (Figure 1(C)) are used for (1) showing POIs, (2) highlighting streets with access time, (3) selecting this region to construct CTR, (4) making this region visible/invisible, or (5) deleting the region. To create a new CTR, users can select multiple existing regions on the panel, and then click one of the buttons on the top to choose IMin, IMax, UMin, UMax, or Diff. Some or all regions can be selected to compare them in the visual report view.

C. Drawing a Region as Concave Hull

Given the set of reachable street segments in a PTR or CTR, a specific drawing method is implemented to draw the region on the map. There exist many ways to draw a bounding region enclosing all points of the segments, from a convex hull to different concave hulls. Figure 5a shows the convex hull of the orange dots. It may enclose the points (shown as X) of other street segments that do not belong to this set. These segments need to be removed from the convex hull, so the region becomes a concave hull. Unlike convex hull, concave hulls are not unique for a set of points. They capture the shape of the boundary of a dataset in different levels measured by concaveness [25]. We first calculate the convex hull with a Divide and Conquer algorithm and then uses the Gift Opening algorithm (peeling external triangles after Delaunay triangulation) to create the hull [28]. However, the gift opening algorithm only uses the angles between consecutive bounding edges to achieve given concaveness threshold. It does not consider the enclosed X points. We modify this algorithm for our purpose. When peeling triangles from outside to inside on the convex hull, if we found X points inside a triangle, then the corresponding edges are removed. Figure 5b illustrates the result hull.

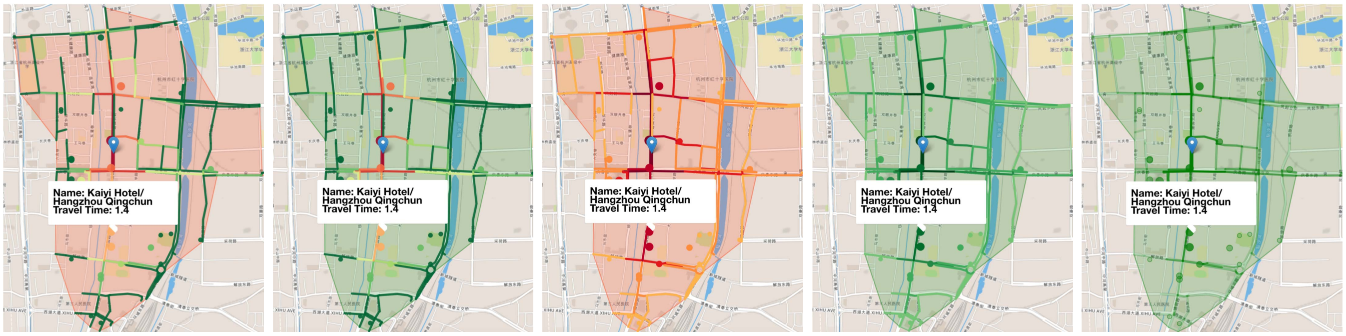
D. Visual Cues for Access Time

The street segments and POIs inside a region are visualized based on their access time from the seed. The hull of the region is visualized as a transparent area with color C_R, and the access time to the inside street segments and POIs (shown in dots) are mapped to a selected color spectrum $\hat{C}_s$ which is important as visual cues. Matching C_R with $\hat{C}_s$ for good perception is important to promote easy understanding. Figure 6 shows three different designs: (1) a fixed $\hat{C}_s$ where a typical distinct spectrum from red to green is used. Figure 6a-b are the results when C_R are from red (small access time) to green (large access time). (2) $\hat{C}_s$ is selected to be chromatically close to C_R. Figure 6c has the orange C_R and $\hat{C}_s$ changes from dark orange (small access time) to light orange (large access time), and in Figure 6d C_R is green and $\hat{C}_s$ has similar dark green (small access time) to light green (large access time) spectrum. (3) Figure 6e uses the same color as C_R (green) for $\hat{C}_s$ while only the opacity is different. Here the dot sizes of POIs also vary to show the access time. In user study, we found that the first design is good for studying specific POIs in one region. Based on our domain experts' suggestion, warm and hot colors (red or orange) are preferred visual cues to highlight those objects that can be reached faster. Traditionally, fast traffic speed on roads is shown in green while red is used for jams. However, our goal is different to show the accessibility of a POI/Street. The domain users pointed out that the warm colors can help viewers realize fast reaching structures better because they are more attractive.

The second design is good when showing POIs in multiple regions without confusion. It also helps to make the visual cues friendly for color-blinded people. The third design is not very good on a complex map view. So in USAVis, users can select the first or the second method based on their interest. In implementation, we pre-define twenty C_Rs. For the second design, we artificially find the chromatically similar $\hat{C}_s$ to these C_Rs. Users can also assign their preferred colors (as well as transparency) on the fly during investigation.

E. Studying and Highlighting POIs

For a PTR/CTR region, users can find the numbers of POIs in each category in the POI panel. When selecting one



(a) (b) (c) (d) (e)

Fig. 6. Visualizing a region with color C_R, and the access times of the inside street segments and POIs are mapped to a color spectrum $\hat{C}_s$: (a) C_R is orange and $\hat{C}_s$ is from red (small access time) to green (large access time); (b) C_R is green and $\hat{C}_s$ is from red to green; (c) C_R is orange and $\hat{C}_s$ is from dark orange (small access time) to light orange (large access time); (d) C_R is green and $\hat{C}_s$ is from dark green (small access time) to light green (large access time); (e) C_R is green and $\hat{C}_s$ is green with varying opacity.

category, the list of POIs are visualized by the order of access time τ. Users can click to highlight a POI and display its details and the access time in a popup box.

F. Visual Comparison of Region Characteristics

The visual report view includes three tabs with a set of charts and diagrams to compare characteristics of PTRs/CTRs: (1) time-varying attributes of the selected regions such as taxi pickup and taxi drop-offs; (2) static attributes of the selected regions such as total number of POIs, and geographic area; (3) the numbers of POIs in different categories.

VII. Case Studies and Performance

We describe several usage scenarios proposed by our collaborative urban researchers. We use a taxi trajectory dataset that is sampled in one month by 8,120 taxis at Hangzhou city in China. The dataset of the whole month (Dec. 1-31, 2011) has a raw size of 77GB for about 270 million GPS sampling points. Each sampling point contains information like trajectory id, latitude, longitude, time stamp, speed, state (occupied or not). For urban structures, we collect 247,642 POIs which are grouped into 18 categories including real estate, shopping, education and training, hotels, government agencies, medical care, and etc. The road network data of Hangzhou is acquired from OpenStreetMap including a collection of 9,764 raw road segments. By applying algorithm described in Sec. IV-A, 14,639 road segments are generated for USAGraphs. Each USAGraph has 14,639 nodes and 58,386 edges with a size of about 13.5MB.

A. Use Cases

Our team includes one active urban geography researcher. The researcher has a PhD degree in geography for urban mobility study. He has also worked as urban transportation planner in Hangzhou city. As our system is also designed for casual users in their urban life, we also recruit two local residents of Hangzhou who lived in the city for more than 20 years to help in our system design. In addition to usage scenario in Section III, the following use cases are proposed by these users to show the usability.

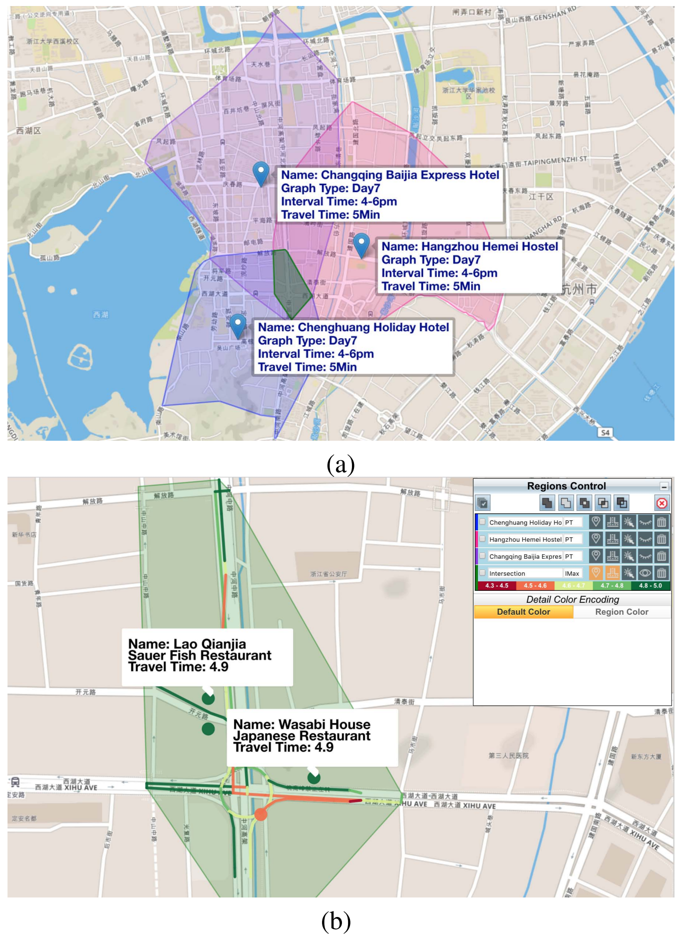
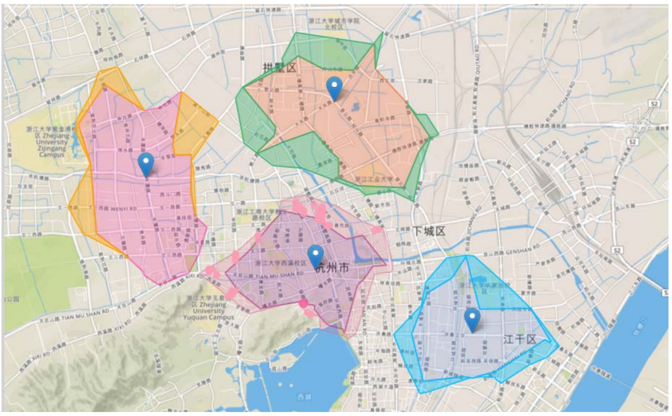
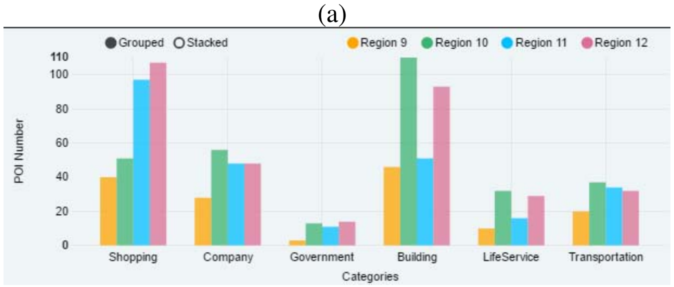


Fig. 7. Finding a restaurant from three starting hotels. (a) CTR of three PTRs using IMax operation (Eqn. 3); (b) Restaurants inside this CTR.

1) Finding a restaurant to meet friends: Three friends residing in three different hotels wanted to meet in a restaurant close to each other. Three PTRs are then created which are seeded at their hotels respectively. The travel time is set as 5 minutes and the time period is from 2-4pm. Figure 7a shows the three reachable regions in different colors. The seed hotels are labeled to show the information. IMax is applied over their



(a)



(b)

Fig. 8. Studying the difference of reachable regions between morning and night from four fire stations.

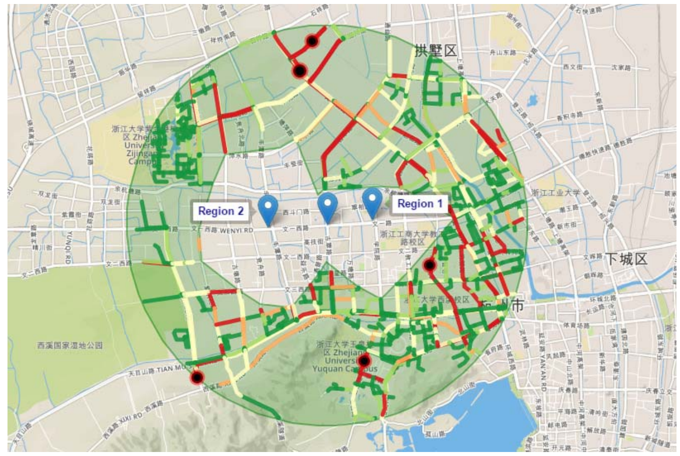


Fig. 9. Finding hot locations of taxi pickups using accessibility difference.

PTRs to form a CTR, where τ of each object is the maximum travel time of the three friends. This accessibility value reflects the earliest time these friends can meet together. Figure 7b is the zoomed view of the CTR to find a good restaurant, where two restaurants are displayed as green dots showing that they may meet around 4.9 minutes.

2) Dynamic Accessibility of Multiple Fire Stations: Users can effectively study time-varying accessibility. For instance, fire fighters want to find what area can be reached in t minutes from their stations in normal daytime, but cannot be reached within the same t in morning rush hours. Figure 8a shows four different fire stations. On each station, two PTRs are computed with $t = 4$ minutes, one for $T = 6 - 8am$ shown as the inner region and another one for $T = 10 - 12pm$ shown as the outer region. After applying the difference operations, the difference regions of accessibility are computed which are shown as the ring shapes between inner and outer regions. Using the visual report view, Figure 8b shows the different numbers of POIs in six categories for the four regions. The resident buildings shown as pink dots in Figure 8a cannot be reached in 4 minutes by fire trucks in the morning.

3) Finding Locations for New Bus Stations: Figure 9 shows an example combining CTR operations with taxi trips. Two PTRs, $R_{station1}$ and $R_{station2}$, are created for two subway stations. Each of them is the region where people can *walk* to the corresponding station in 20 minutes. Here a constant walking speed 5km/hour is used in the graph traversal. Meanwhile, users choose a PTR, R_{circle}, by setting a fixed radius as 4km where the seed is centered between the two stations. Inside the circle, they want to find places where residents cannot walk to the subway in 20 minutes. So commuter buses may be provided to transfer residents to the subway stations. A combined MinMaxJS operation: $R_{circle} - (R_{station1} \cup R_{station2})$ is applied to create a green CTR as shown in Figure 9. The important streets inside this CTR are colored by the number of taxi pickups that happened in one whole month, where red indicates more pickups. Five top locations with the largest number of pickups are marked as black dots, which are the candidate commuter bus stations.

4) Studying Time-Varying Accessibility With Complex Joint Operations: As a comprehensive example, we study time-varying accessibility from multiple seed locations with complex joint operations. Two police stations, $S1$ (Xixi) and $S2$ (Tianshui), are used as seeds. For each seed, we generate two PTRs, one in the morning ($T1 =$ 10am-12pm) and one in the afternoon ($T2 =$ 4pm-6pm), while $t = 4$ minutes as the driving time. So that, four PTRs are created, R_{S1}^{T1}, R_{S1}^{T2}, R_{S2}^{T1}, R_{S2}^{T2}, as shown in Figure 10a. First, Figure 10b shows a CTR computed by

$$CTR1 = (R_{S1}^{T1} \cap R_{S2}^{T1}) - (R_{S1}^{T2} \cap R_{S2}^{T2}) \tag{7}$$

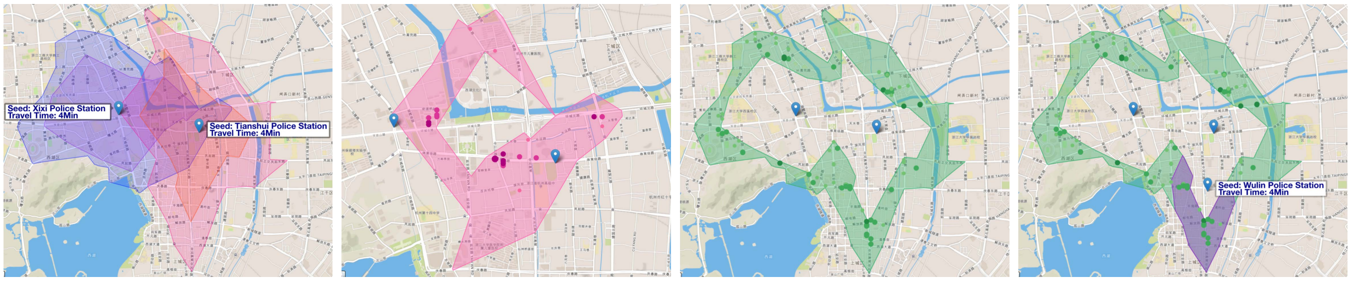
$CTR1$ is the accessible region (with bank POIs shown as dots) from both $S1$ and $S2$ in 10am-12pm but cannot be reached from both $S1$ and $S2$ in 4-6pm possibly due to the afternoon traffic. Second, Figure 10c shows another CTR computed by

$$CTR2 = (R_{S1}^{T1} \cup R_{S2}^{T1}) - (R_{S1}^{T2} \cup R_{S2}^{T2}) \tag{8}$$

It instead shows the green area that can be reached from either $S1$ or $S2$ in 10am-12pm, but cannot be reached from either $S1$ or $S2$ in 4-6pm. The banks (in dots) in region may be of risk in the afternoon while no police can reach it in the given time. Thus, users want to check the accessibility from another police station, $S3$ (Wulin). Figure 10d is the result after performing

$$CTR3 = ((R_{S1}^{T1} \cup R_{S2}^{T1}) - (R_{S1}^{T2} \cup R_{S2}^{T2})) \cap R_{S3}^{T2} \tag{9}$$

The purple area shows the region that Wulin station can reach in the afternoon. Then, the remaining part of the green area needs consideration to improve their accessibility.



(a) (b) (c) (d)

Fig. 10. Studying time-varying accessibility with complex joint operations. (a) Four PTRs; (b) $CTR1$; (c) $CTR2$; (d) $CTR3$.

TABLE I
PERFORMANCE OF CREATING A PTR WITH USAGRAPH

Travel time of PTR	No. of graph nodes (street segments)	NO. of POIs in PTR	PTR generation time (sec)	PTR visualization time (sec)
5 minutes	236	1,714	0.004	0.01
10 minutes	1,268	7,987	0.03	0.13
15 minutes	2,750	17,682	0.15	0.37

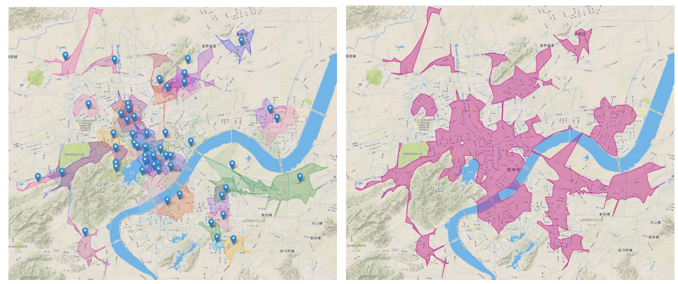
TABLE II
PERFORMANCE OF THE SET OPERATIONS OVER TWO 5-MINUTES PTRS

MinMaxJS Operation	No. of result street segments	NO. of result POIs	MinMaxJS operation time (sec)	Visualization time (sec)
Intersection	38	59	0.03	0.06
Difference	198	1,615	0.02	0.17
Union	375	2,732	0.11	0.22

B. System Performance

We have implemented USAVis as a web-based system using client-server architecture. The server side uses Python, NetworkX, and Neo4j graph database. The client side uses HTML5 and some JavaScript libraries, including Leaflet Map and D3.js. We perform the system test on a Mac laptop (2.5G Intel Core i7, 16G Memory).

Table I shows the performance of creating a PTR (the pink PTR of Fig. 7). Besides this 5-minute PTR, the performance of creating the 10 and 15 minutes PTRs from the same seed location are also reported. The table includes the number of traversed graph nodes, the number of POIs, the PTR generation time, and the visualization time. Here the generation time includes the BFS graph traversal algorithm from the seed and the computation of all reachable nodes' access time. The worst-case BFS complexity is $\mathcal{O}(V + E)$ for V vertices and E edges in the USAGraph, when all nodes need to be visited. Obviously, our computation is faster due to stopping condition of the time length limit, a PTR will not include all streets in the city but just a very small portion. The visualization time is the time to compute the concave hull and draw it on the map. The running time of the concave-hull algorithm is $\mathcal{O}(n \log n)$ for n points in the input PTR/CTR. Table II shows the performance of computing a CTR from the two 5-minutes PTRs (the pink and blue PTRs in Fig. 7). Three MinMaxJS set operations are tested, whose computing time is reported together with the visualization time. The CTR computation is fast with a complexity of $\mathcal{O}(N)$, where N represents the number of streets and POIs in the PTRs. These tables show that the system can interact with users smoothly for the study of several accessible regions.



(a) (b)

Fig. 11. Studying city-wide hospital accessibility. (a) 54 PTR regions reachable from hospitals in 5 minutes' driving at the rush hours of 4-6pm; (b) Accessible map of the city from (a).

C. Scalability Study

Our system is scalable in representing many spatiotemporal constraints, when users set a large set of seed positions for city-wide accessibility study. Figure 11 shows an example where users find how the people in Hangzhou can be reached by emergency centers in general hospitals all over the city at a specific time. Figure 11a illustrates 54 PTRs created by using 54 emergency centers in Hangzhou. Each PTR represents the accessible region from one center (a marker on the map) in 5 minutes at the afternoon rush hour 4-6pm. Using "Umin" for all the 54 PTRs, Figure 11b shows the CTR, which indicates the emergency accessible region in the city in 5 minutes.

The algorithm creates the 54 PTRs all over the city and generates the access times to all involving street segments and POIs in 0.28 seconds. The time of generating the 54 convex hulls and drawing them costs only 1.1 seconds. When the MinMaxJS union operation is conducted to combine these 54 PTRs into one CTR, it is completed in 32.7 seconds. The visualization time is still fast at 1.8 seconds. Here, the CTR generation time is mostly used to re-compute the access times of a large amount of 6,794 roads and 39,489 POIs from 54 seeds. This re-computation is necessary since the 54 PTRs have overlaps and the objects' *joint* access time inside the big

TABLE III
INTERVIEW QUESTIONNAIRES

Question No.	Question
Q1	Is the system practically applicable to be used in transportation studies, urban planing? (Your opinion)
Q2	Does the system achieve the established goals of studying the reachability of the urban structures based on single and multiple constrains? (Your opinion)
Q3	Do you think the visual report view (visualization diagrams) shows the useful information for analyzing PTR and CTR regions? (Your opinion)
Q4	Is the interactive visual design interface friendly and easy to use? Give us your opinion about the interactive visual design of the way of creating the PTR and CTR, controlling the regions, and showing the access time to the POIs and road segments inside the regions.
Q5	Is the performance of the system good for interactive exploratory analysis? (Is the system fast enough?)
Q6	What are the limitations of the system?
Q7	What are your suggestions for improving the system?

CTR is different than the access time from those individual seeds. This shows one limitation of our current system, partly due to the sequential implementation of the graph traversal algorithm. This limitation can be improved with a server with fast computation and a high performance graph database. We will also improve the system scalability and performance by parallel acceleration.

VIII. Task-Based User Study and Feedback

We conducted task-based user study with a group of 12 active domain experts in the areas of urban planning and transportation, GIS, remote sensing, and geography.

A. User Study Procedure

First, we explained the system by giving a presentation to the group. Second, we showed them some usage scenarios. Third, we taught them how to use and interact with the visual interface. Then, we allowed each one of them to use and explore the system for 5 minutes. After these steps, we asked each one of them to implement two tasks. Finally, each expert was interviewed by answering a set of questions listed in Table III to provide his/her evaluation and suggestions. This interview was performed by talking with each person and the process was recorded in audio files, they also wrote their feedback by answering the questions.

B. User Study Analysis

The participants were asked to implement two tasks individually. In the first task, the participants were asked to (1) create three PTRs seeded at three given hotels A, B, and C, using $t = 5$ minutes and T is 4-6pm; (2) construct the CTR from the three PTRs using IMax operation; and (3) they were asked to answer the following questions: "Which POI category has more POIs in the created CTR?," "How Many restaurants are there in the same CTR?," "Which restaurant can be accessed faster than others by driving from the three hotels?," and "What is the access time to the fastest access restaurant?." This task represents a typical process of investigating dynamic accessibility with multiple constraints. The average completion time of 92% of participants was 3 minute. 92% of participants achieved the correct answers. 75% of them agreed that the system was *very easy* to use, 17% said that the system was *easy* to use, and 8% said that system was *fair* to use. Next, we summarize their feedback.

In the second task, the participants were asked to create four PTRs seeded at four given points, using $t = 4$ minutes and T is 8-10am. They were asked to use the visual report view to answer the following questions: "What is the area of the biggest region?," "What is the total length of the reachable street segments?," "How many education places are there in the biggest region?," "How many taxi pick up happened in the biggest region on time interval 4-6pm?," and "How many taxi drop off happened in the biggest region on time interval 8-10am?." This task shows that our system can be used for urban applications such as location-based recommendation. The average task completion time was 4 minutes. 83% of participants achieved correct answers. Most of the participants agreed that the system was easy to use.

C. Participants' Feedback

By summarizing the participants' answers on Q1-Q3 from Table III, we found that 92% of the participants' showed unanimous agreement on the usefulness of the system. One of them said "This system is very interesting, and really useful. It shows great potential in urban service planning and applications.." Another one said "I can tell this system is excellent tool for parents to improve their schedule. For example, using this system I can tell which schools are good for my kids based on the reachability of the schools from my home, my job, …." The participants were satisfied about the visual interface and functions. All of them agreed that the interface is friendly and easy to use. They liked the interactive way of creating PTRs and CTRs. The labels and color mapping of showing the access times to the POIs were considered effective. They commented "I admit the effectiveness of creating CTR from multiple PTRs to study the reachability of the POIs from multiple constrains"; "It is easy to get familiarized with the interface." Most of them believed that the coordinated views were designed effectively, which provide a conducive way for comprehensive analysis. One said "using a unique and same color to represent each region on the map view, regions control panel, and in all visualization diagrams of the report view is wonderful and really useful to allow the user to differentiate and recognize the regions." Another commented "the feature of allowing the user to change default name of the region to desired one is very important." Moreover, all of them agreed that the report view provides extra useful information about the regions. Finally, the users agreed that the system performance is fast for interactive visual analysis tasks.

The domain experts pointed out the system limitation and gave valuable suggestions to improve it. First, they realized the relatively steep learning curve for the first time users. They suggested to make more labels and explanations in the interface and add interactive tutorial to the system. Second, the system should have the managerial functions such as *save* and *load*, so that users can save their work progress and share with other. Third, more POI information may be added and the seed points may also be defined by mailing addresses and zipcodes for convenience. We will take these advice in furture work.

IX. Conclusion and Discussion

We have developed new computational models and visualization tools for users to study dynamic and joint-constrained accessibility of urban structures. Reachable regions are easily formed to satisfy joint geospatial-temporal constraints by a USAGraph model and newly designed MinMaxJS set operations. The visualization system provides intuitive, easy-to-use interface so that users can efficiently perform their investigation on the regions and their POIs.

The major limitation of the graph-based accessibility model is its dependency on data quality. The completeness and correctness of street network determines whether or not the graphs can be correctly formed. Taxi trajectories also need to provide enough traffic data for each street. There are several future directions. First, the public transit based accessibility will be integrated in the system. Second, we will enhance the system with realtime traffic data from APIs such as Google Map. Third, we will extend our system to an area bigger than one city, such as a state, a province, or a country. The huge number of street segments and POIs will require new techniques such as utilizing parallel graph databases and algorithms.

References

[1] G. Andrienko, N. Andrienko, W. Chen, R. Maciejewski, and Y. Zhao, "Visual analytics of mobility and transportation: State of the art and further research directions," *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 8, pp. 2232–2249, Aug. 2017.

[2] T. Anwar, C. Liu, H. L. Vu, and M. S. Islam, "RoadRank: Traffic diffusion and influence estimation in dynamic urban road networks," in *Proc. 24th ACM Int. Conf. Inf. Knowl. Manage. (CIKM)*, 2015, pp. 1671–1674.

[3] V. Bauer, J. Gamper, R. Loperfido, S. Profanter, S. Putzer, and I. Timko, "Computing isochrones in multi-modal, schedule-based transport networks," in *Proc. 16th ACM SIGSPATIAL Int. Conf. Adv. Geograph. Inf. Syst. (GIS)*, 2008, pp. 78:1–78:2.

[4] S. Chen *et al.*, "Interactive visual discovering of movement patterns from sparsely sampled geo-tagged social media data," *IEEE Trans. Vis. Comput. Graphics*, vol. 22, no. 1, pp. 270–279, Jan. 2016.

[5] W. Chen, F. Guo, and F. Y. Wang, "A survey of traffic data visualization," *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 6, pp. 2970–2984, Jun. 2015.

[6] P. Corcoran, M. Jilani, P. Mooney, and M. Bertolotto, "Inferring semantics from geometry: The case of street networks," in *Proc. 23rd SIGSPATIAL Int. Conf. Adv. Geograph. Inf. Syst. (GIS)*, 2015, pp. 42:1–42:10.

[7] M. Domingo, R. Thibaud, and C. Claramunt, "A graph-based model for the representation of land spaces," in *Proc. 21st ACM SIGSPATIAL Int. Conf. Adv. Geograph. Inf. Syst.*, 2013, pp. 516–519.

[8] A. Efentakis, N. Grivas, G. Lamprianidis, G. Magenschab, and D. Pfoser, "Isochrones, traffic and DEMOgraphics," in *Proc. 21st ACM SIGSPATIAL Int. Conf. Adv. Geograph. Inf. Syst. (SIGSPATIAL)*, 2013, pp. 548–551.

[9] J. Gamper, M. Böhlen, W. Cometti, and M. Innerebner, "Defining isochrones in multimodal spatial networks," in *Proc. 20th ACM Int. Conf. Inf. Knowl. Manage. (CIKM)*, 2011, pp. 2381–2384.

[10] F. Gortana, S. Kaim, M. von Lupin, and T. Nagel, "Isoscope—Visualizing temporal mobility variance with isochrone maps," in *Proc. Poster Abstr. IEEE VIS*, 2014, pp. 1–2.

[11] S. L. Handy and K. J. Clifton, "Evaluating neighborhood accessibility: Possibilities and practicalities," *J. Transp. Statist.*, vol. 4, nos. 2–3, pp. 67–78, 2001.

[12] S. R. Hong, Y.-S. Kim, J.-C. Yoon, and C. R. Aragon, "Traffigram: Distortion for clarification via isochronal cartography," in *Proc. SIGCHI Conf. Hum. Factors Comput. Syst. (CHI)*, 2014, pp. 907–916.

[13] X. Huang, Y. Zhao, J. Yang, C. Ma, C. Zhang, and X. Ye, "TrajGraph: A graph-based visual analytics approach to studying urban network centralities using taxi trajectory data," *IEEE Trans. Vis. Comput. Graphics*, vol. 22, no. 1, pp. 160–169, Jan. 2016.

[14] C. Hurter, B. Tissoires, and S. Conversy, "FromDaDy: Spreading aircraft trajectories across views to support iterative queries," *IEEE Trans. Vis. Comput. Graphics*, vol. 15, no. 6, pp. 1017–1024, Nov./Dec. 2009.

[15] M. Jilani, P. Corcoran, and M. Bertolotto, "Multi-granular street network representation towards quality assessment of openstreetmap data," in *Proc. 6th ACM SIGSPATIAL Int. Workshop Comput. Transp. Sci. (IWCTS)*, 2013, pp. 19:19–19:24.

[16] R. Krüger, D. Thom, and T. Ertl, "Visual analysis of movement behavior using Web data for context enrichment," in *Proc. IEEE Pacific Vis. Symp.*, Mar. 2014, pp. 193–200.

[17] Q. Li, T. Zhang, H. Wang, and Z. Zeng, "Dynamic accessibility mapping using floating car data: A network-constrained density estimation approach," *J. Transp. Geogr.*, vol. 19, no. 3, pp. 379–393, 2011.

[18] W. Lin, Y. Mi, W. Wang, J. Wu, J. Wang, and T. Mei, "A diffusion and clustering-based approach for finding coherent motions and understanding crowd scenes," *IEEE Trans. Image Process.*, vol. 25, no. 4, pp. 1674–1687, Apr. 2016.

[19] W. Lin *et al.*, "A tube-and-droplet-based approach for representing and analyzing motion trajectories," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 8, pp. 1489–1503, Aug. 2017.

[20] W. Luo and F. Wang, "Measures of spatial accessibility to health care in a gis environment: Synthesis and a case study in the Chicago region," *Environ. Planning B, Planning Des.*, vol. 30, pp. 865–884, Dec. 2003.

[21] N. Markovic, P. Sekuła, Z. V. Laan, G. Andrienko, and N. Andrienko, "Applications of trajectory data from the perspective of a road transportation agency: Literature review and maryland case study," *IEEE Trans. Intell. Transp. Syst.*, to be published.

[22] A. De Montis and A. Reggiani, "Cities special section on 'Analysis and planning of urban settlements: The role of accessibility,"' *Cities*, vol. 30, pp. 1–3, Feb. 2013.

[23] T. Neutens and M. Versichele, "A GIS toolkit to evaluate individual and joint accessibility to urban opportunities," in *Proc. Colloq. Vervoersplanologisch Speurwerk (CVS)*, 2009.

[24] D. Papadias, J. Zhang, N. Mamoulis, and Y. Tao, "Query processing in spatial network databases," in *Proc. 29th Int. Conf. Very Large Data Bases*, vol. 29, Sep. 2003, pp. 802–813.

[25] J.-S. Park and S.-J. Oh, "A new concave hull algorithm and concaveness measure for n-dimensional datasets," *J. Inf. Sci. Eng.*, vol. 28, pp. 587–600, May 2012.

[26] M. Picone, S. Busanelli, M. Amoretti, F. Zanichelli, and G. Ferrari, *Advanced Technologies for Intelligent Transportation Systems*. Cham, Switzerland: Springer, 2014.

[27] P. Rode *et al.*, *Accessibility in Cities: Transport and Urban Form*. Cham, Switzerland: Springer, 2014, doi: 10.1007/978-3-319-51602-8_15.

[28] E. Rosen, E. Jansson, and M. Brundin, "Implementation of a fast and efficient concave hull algorithm," Dept. Inf. Technol., Uppsala Univ., Uppsala, Sweden, Tech. Rep., 2014.

[29] M. Rosvall, A. Trusina, P. Minnhagen, and K. Sneppen, "Networks and cities: An information perspective," *Phys. Rev. Lett.*, vol. 94, pp. 28701–28704, Jan. 2005.

[30] C. Shahabi, F. Banaei-Kashani, A. Khoshgozaran, L. Nocera, and S. Xing, "GeoDec: A framework to visualize and query geospatial data for decision-making," *IEEE MultiMedia*, vol. 17, no. 3, pp. 14–23, Jul. 2010.

[31] A. F. Stewart, "Visualizing urban accessibility metrics for incremental bus rapid transit projects," Ph.D. dissertation, Dept. Urban Stud. Planning, Massachusetts Inst. Technol., Cambridge, MA, USA, 2014.

[32] D. Weng, H. Zhu, J. Bao, Y. Zheng, and Y. Wu, "HomeFinder revisited: Finding ideal homes with reachability-centric multi-criteria decision making," in *Proc. ACM CHI Conf.*, 2018, p. 247.

[33] G. Wu, Y. Ding, Y. Li, J. Bao, Y. Zheng, and J. Luo, "Mining spatio-temporal reachable regions over massive trajectory data," in *Proc. IEEE 33rd Int. Conf. Data Eng. (ICDE)*, Apr. 2017, pp. 1283–1294.

[34] H. Xu, Y. Zhou, W. Lin, and H. Zha, "Unsupervised trajectory clustering via adaptive multi-kernel-based shrinkage," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 4328–4336.

[35] W. Zeng, C. W. Fu, S. M. Arisona, A. Erath, and H. Qu, "Visualizing mobility of public transportation system," *IEEE Trans. Vis. Comput. Graphics*, vol. 20, no. 12, pp. 1833–1842, Dec. 2014.

[36] W. Zeng, C.-W. Fu, S. M. Arisona, S. Schubiger, R. Burkhard, and K.-L. Ma, "Visualizing the relationship between human mobility and points of interest," *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 8, pp. 2271–2284, Aug. 2017.

[37] B. Zhou, X. Wang, and X. Tang, "Random field topic model for semantic region analysis in crowded scenes from tracklets," in *Proc. (CVPR)*, Jun. 2011, pp. 3441–3448.

[38] S. Al-Dohuki *et al.*, "SemanticTraj: A new approach to interacting with massive taxi trajectories," *IEEE Trans. Vis. Comput. Graph.*, vol. 23, no. 1, pp. 11–20, Jan. 2017.

Farah Kamw is currently pursuing the Ph.D. degree with the Department of Computer Science, Kent State University. Her research interests include visual analytics of urban trajectory data, spatial database, urban accessibility, and urban navigation.

Shamal AL-Dohuki is currently pursuing the Ph.D. degree with the Department of Computer Science, Kent State University. His research interests include urban data management and visualization, visual query of trajectory data, and semantic data query and analytics.

Ye Zhao is currently a Professor with the Department of Computer Science, Kent State University, Ohio, Kent, USA. His current research interests include visual analytics of urban transportation data, multidimensional text and animated information visualization, medical image processing, and computational hemodynamics modeling.

Thomas Eynon received the B.S. degree from the Department of Computer Science, Kent State University. His research interests include urban data visualization, visual analytics, and process automation.

David Sheets is currently pursuing the Ph.D. degree with the Department of Computer Science, Kent State University. His research interests include spatial database, distributed urban data management, and semantic transformation of urban data.

Jing Yang is currently a Professor with the Department of Computer Science, University of North Carolina at Charlotte. Her research interests include visual analytics of multidimensional data, time-oriented data, text data, urban transportation data, and sports data.

Xinyue Ye is currently a Professor with the Department of Informatics, New Jersey Institute of Technology. His research interests include spatial social network analysis, urban computing, and GIS.

Wei Chen is currently a Professor with the State Key Lab of CAD and CG, Zhejiang University. His current research interests include visualization, visual analytics, and bio-medical image computing.