Large-scale transit market segmentation with spatial-behavioural features

Article *in* Transportation Research Part C Emerging Technologies · March 2018

DOI: 10.1016/j.trc.2018.03.003

Minh Le Kieu
University of Auckland
28 PUBLICATIONS **268** CITATIONS

Large-scale Transit Market Segmentation with Spatial-Behavioural Features

Le Minh Kieu[1], Yuming Ou[1], Chen Cai[1]

[1] *Data61, CSIRO, Sydney, Australia*

Abstract

Transit market segmentation enables transit providers to comprehend the commonalities and heterogeneities among different groups of passengers, so that they can cater for individual transit riders' mobility needs. The problem has recently been attracting a great interest with the proliferation of automated data collection systems such as Smart Card Automated Fare Collection (AFC), which allow researchers to observe individual travel behaviours over a long time period. However, there is a need for an integrated market segmentation method that incorporating both spatial and behavioural features of individual transit passengers. This algorithm also needs to be efficient for large-scale implementation. This paper proposes a new algorithm named Spatial Affinity Propagation (SAP) based on the classical Affinity Propagation algorithm (AP) to enable large-scale spatial transit market segmentation with spatial-behavioural features. SAP segments transit passengers using spatial geodetic coordinates, where passengers from the same segment are located within immediate walking distance; and using behavioural features mined from AFC data. The comparison with AP and popular algorithms in literature shows that SAP provides nearly as good clustering performance as AP while being 52% more efficient in computation time. This efficient framework would enable transit operators to leverage the availability of AFC data to understand the commonalities and heterogeneities among different groups of passengers.

Keywords: Customer segmentation, travel pattern, smart card, AFC

1. Introduction

Maintaining service quality and customer satisfaction is challenging for transit operators due to the heterogeneity in passengers spatial locations and travel patterns. Passenger behaviours and needs vary across different market segments from transit commuters to infrequent passengers, or from adults to students and senior passengers (Kieu et al., 2015b). Complex urban structures of spatially diverse educational, recreational and occupational locations further nurture the diversity of these segments and complicate service provisions. Ridership competition from private transport and new shared-mobility transport such as carsharing and ridesourcing (e.g. Uber or Lyft) also forces transit operators to understand more about their customers and cater individual needs.

Market segmentation is a popular procedure in economics to classify a market of customers into segments sharing similar interests, needs or locations. Market segmentation is essential for transit operators to understand the commonalities and heterogeneities among different groups of passengers. It aims to define specific subsets of passengers sharing similar characteristics in demographics, psychograpic, geographic or behavioural so that transit operators can break down the mobility requirements of everyone and align their services to specific needs. Understanding the individual needs will enable operators to implement: (1) targeted surveys to understand specific groups of passengers, (2) incentives and personalised transit service to reward loyal passengers, and (3) on-demand services for under-served areas or during incidents.

The most basic form of transit market segmentation is through age and social situations, which are usually Adult, Senior, Child, and Student. While this level of market segmentation is useful for ticketing purposes, it provides unbalanced segments where customers do not share similar interests, needs or locations. Research on transit market segmentation started early with a study by Tybout et al. (1978), but has only recently been attracting a great interests with the proliferation of automated data collection systems such as Smart Card Automated Fare Collection (AFC). Smart Card AFC data captures a rich information that can potentially reveal a more comprehensive understanding of passenger travel patterns and mobility needs. Literature of data-driven studies using Smart Card data has evolved from simple data enriching studies (Alfred Chu and Chapleau, 2008, Bagchi and White, 2005), to mining individual temporal-spatial travel patterns (Kieu et al., 2015a, Kusakabe and Asakura, 2014, Ma et al., 2013) and recently to improvements of transit operation, such as predicting passenger flow (Kieu et al., 2017, Li et al., 2017, Ma et al., 2014) or vehicle arrival time estimation (Min et al., 2016, Zhou et al., 2017).

Literature offers a number of approaches to transit market segmentation using Smart Card AFC data. Agard et al. (2006) adopted the Hierarchical Ascending Clustering to segment transit passengers using only temporal travel patterns. Lathia et al. (2013) applied Agglomerative Hierarchical Clustering to segment passengers using temporal travel profiles, aggregated in five daily time periods. Kieu et al. (2015b) adopted a bi-level Density-based Scanning Algorithm with Noise (DBSCAN) (Ester et al., 1996) to mine individual travel patterns and then proposed a priori segmentation method to segment transit passengers. Legara and Monterola (2017) introduced the concept of eigentravel matrices to capture passenger travel characteristics and developed a classification technique with promising accuracy. Langlois et al. (2016) inferred passenger travel patterns through a longitudinal representation of multi-week activities. Passenger travel areas were clustered using Agglomerative Hierarchical Clustering, and then longitudinal travel patterns were clustered by principal component analysis. Briand et al. (2017) clustered transit passengers of Gatineau City, Canada using a Gaussian mixture model with Classification Expectation Maximisation. The proposed model is capable of cluster passengers based on continuous temporal travel activities. A model-based mixture model using Expectation Maximisation was also proposed in El Mahrsi et al. (2017), where the authors proposed two approaches to cluster transit passengers from a station-oriented and a passenger-focused standpoints using their temporal travel patterns.

Understanding the geographic market segments in the transit industry is indeed essential because transit service provision is spatially limited. Transit passengers usually walk to stops, thus their rational travel choices often limit within a walking distance. Transit operators who can leverage such spatial understanding about passenger segments will be able to provide better services. For instance, travel information can be given to passenger segments at areas influenced by an incident. The impact of future transit management plans can be foreseen given the passenger segments on the impacted areas. Spatial transit market segmentation is also helpful for passenger choice modelling, such as modal and route choices, because passengers of the same segment living closely are sharing similar travel behaviours and facing a similar choice set.

However, the consideration of individual geographical characteristics in existing passenger segmentation studies is relatively limited. In the conclusion of their work, Agard et al. (2006) recognised that the inclusion of geospatial trip behaviour would enable better understanding of transit supply and demand. In Lathia et al. (2013), the spatial characteristics of each passenger are considered as the number of visited stations, rather than the spatial proximity of adjacent stations. Langlois et al. (2016) considered spatial proximities when clustering the user-specific areas using a predefined threshold distance value among stops/stations. There remains a need for integration of spatial and behavioural features in an integrated spatial-behavioural transit market segmentation.

In this paper, we define the concept of spatial-behavioural passenger segments as the spatially-limited clusters of passengers who have similar behaviours. Identifying these spatial-behavioural segments is challenging. First, a new distance metric will be required to incorporate spatial and behavioural features in segmentation, because they are in different units of measures. Second, since transit service provision is spatially limited, it is important to predefine a maximum spatial size value Δ for each passenger segment. Segment's spatial size measures the maximum distance from any two passengers belong to the same segment. If Δ is large, passengers might be less spatially relevant to each other, but there might be more chances to find passengers of similar travel behaviours. For instance, let us say that a transit provider wants to find spatial segments of regular passengers in an area to provide a coach service. To maximise the utility of each vehicle and limit the boarding time, the transit provider may stop the coach only at the centre of each spatial-behavioural passenger segment, and ask the passengers to walk to those predefined stops. In this example if the spatial size is large, the walk would be more tiresome, though may be more regular passengers can be grouped together to maximise vehicle's utility. If the spatial size is small, the average walking distance would be convenient, but there might also be less passengers of similar behaviours in each spatial-behavioural passenger segment. Too small Δ may even lead to spatial singletons, those are segments at a single location, which is unfavourable from an efficiency point of view. It is challenging to define Δ because the data-driven passenger segmentation results depend on the spatial and behavioural distributions of passengers' characteristics. Choosing a value of Δ may require inputs from an expert with domain-knowledge, since an increase in Δ may lead to an increase in both similarity among passengers of the same segment, and an increase in passengers' walking distance. Multiple implementations of the transit market segmentation algorithm with different values of Δ may be required.

The search for a value of Δ leads us to another problem in large-scale transit market segmentation: scalability. Existing literature in transit market segmentation have been often developed using a random sub-sample of limited size, such as in Lathia et al. (2013), especially when the number of market segments are usually not known, so that a hierarchical method is required (Lathia et al. (2013),Agard et al. (2006),Langlois et al. (2016)) or multiple runs with different number of segments are required (Briand et al. (2017),El Mahrsi et al. (2017)). Kieu et al. (2015b) is an exception where over a million of Smart Cards were considered, but the segmentation method is rather simplistic with a set of *a priori* heuristic rules. Table 1 shows the number of Smart cards, number of trips considered and method adopted in existing transit market segmentation studies.

Table 1: Comparison of existing literature on transit market segmentation

Paper	**Number of Cards**	**Number of trips**	**Method**
Agard et al. (2006)	25452	2147049	Hierarchical Ascending Clustering
Lathia et al. (2013)	2000	Not available	Agglomerative Hierarchical Clustering
Kieu et al. (2015b)	1 million	34.8 million	A priori Heuristic Rules
Langlois et al. (2016)	33026	Not available	Agglomerative Hierarchical Clustering
Briand et al. (2017)	82223	3492310	Expectation Maximisation
Legara and Monterola (2017)	30000	Not available	Multiple Machine Learning methods
El Mahrsi et al. (2017)	134979	5404096	Expectation Maximisation

Addressing the limitations of existing methods, this paper introduces a new approach to transit spatial-behavioural passenger segmentation on a large scale, where transit passengers of spatially close distance who behave similarly are grouped in a market segment. As discussed earlier, information on these condensed spatial segments enables transit operators to develop transit tactical and operational strategies. However, we expect that there will be thousands of segments and millions of passengers on a large-scale case study of a large metropolitan area, so a scalable and transferable method will be developed. The contributions of this paper are two fold. First, we develop a framework to incorporate spatial and behavioural features into spatial-behavioural passenger segmentation. Second, we propose an algorithm to implement spatial-behavioural market segmentation with varied maximum spatial size Δ on a large scale scenario.

The remainder of this paper describes the related works, the proposed methodology, the numerical comparison with some other algorithms in literature and finally the large-scale case study.

2. Related work to the research challenge

Market segmentation is an unsupervised clustering problem. Each customer, or transit passenger in our problem, is a data point i in a multidimensional space of features.

The first research challenge, as discussed in the introduction, comes when the feature space of each data point includes both spatial and non-spatial variables. Spatial variables show the location of each transit passenger, for instance the Latitude and Longitude $[d^i_{lat}, d^i_{lon}]$ coordinates of each data point, while non-spatial variables showing other characteristics of each data point. One major problem with incorporating spatial and non-spatial feature is that the distance measure between any two points is no longer satisfy the *triangle inequality*. In other words, the sum of the distance/dissimilarity between any one data point to two other points should be larger than the distance between those two points. Incorporating spatal and non-spatial features means that one data point may be 'very close' to another point spatially, but 'very far apart' non-spatially. Popular clustering algorithms such as K-means (Hartigan and Wong, 1979) or Expectation Maximisaion (EM) assume the validity of triangle inequality. One quick solution is to perform a bi-level clustering approach where each level segments one type of features, for instance cluster all the points by their spatial features first, then non-spatial features. Each level of clustering still satisfies the triangle inequality. One very popular example of them is the Spatial Dominant CLARANS (Clustering Large Applications based on RANdomized Search) (Ng and Han, 1994), where the authors proposed to cluster large-scale spatial features first, then explored non-spatial features within each cluster. However, multiple-level clustering means that the problem is tackled from different angle multiple times, which may lead to some loss of information because the results of the earlier clustering level(s) affect the later. For instance, two very similar data points may not have any chance to be group into the same cluster on level two if they are on different cluster on level one.

There are a few other clutering algorithms which do not assume triangle inequality, such as Spectral Clustering (Von Luxburg, 2007), DBSCAN (Ester et al., 1996), Partitioning Around Medoids (Kaufman and Rousseeuw, 1990) and Affinity Propagation (Frey and Dueck, 2007). These methods, instead of relying on the 'locations' of data points to cluster data, rely on a distance or similarity measure between data points. A modification of the original algorithm is usually required to incorporate both spatial and non-spatial features into the same distance/similarity measures to cluster points, while allows a single level of clustering. Birant and Kut (2007) proposed the ST-DBSCAN algorithm, which was an extension of DBSCAN, by introducing one more neighbourhood radius value $Eps(\varepsilon)$ to represent non-spatial variables, along with the original radius value to represent spatial variables. The density-reachable neighbourhood was the intersection between the neighbourhood defined by the two radius value. Wang et al. (2004) proposed another extension of DBSCAN named Density-Based Spatial Clustering in the Presence of Obstacles and Facilitators (DBRS+). The algorithm clustered spatial data with 3 parameters Eps, $MinPts$, and $MinPur$, where $MinPur$ is the additional parameter compared to the classical DBSCAN to differentiate points with different non-spatial properties. However, there are less existing algorithms consider spatial constraints when incorporating spatial and non-spatial features. Spatial constraints are required in our problem because one of the objectives is to find spatially small cluster within an immediate vicinity of a walking distance. As discussed above, it is challenging to find an optimal value for the spatial constraints due to the complexity and computation burden of the problem.

The second research challenge involves the scalability of market segmentation. We aim to find spatially small market segments within an immediate vicinity, and within each of those areas there may be several segments with different non-spatial features, thus the number of market segments may be very large. It is even more challenging when the number of cluster is unknown before hand, so methods requiring multiple runs to get the optimal number of clusters such as K-means, Spectral Clustering and EM will be more expensive. Several other methods do not require the number of clusters predetermined, such as DBSCAN, HDBSCAN Campello et al. (2013) or hierarchical clustering. However, such algorithms possess a high computational complexity and are infeasible for a large-scale clustering problem.

Therefore, the method we will propose here should incorporate both spatial and non-spatial features under a custom distance/similarity metric, while maintain spatially small clusters. The method should not require the number of clusters and maintain high segmentation quality, while remain scalable for large-scale clustering implementations.

3. Methodology

Given a database of N passenger i ($i = 1...N$) with spatial geodetic coordinates of Latitude and Longitude $[d^i_{lat}, d^i_{lon}]$ and a vector of behavioural features $[V^i_1, V^i_2, ..., V^i_D]$, normalised between [0,1], with D is the number of features. Our objective is to cluster this data to K segments C_m $(m = 1...K)$ where K is unknown before hand, and the spatial size φ_m of C_m is less than a walking distance threshold Δ. Assuming the majority of transit passengers walk to transit stops, the spatial size φ_m is the maximum walking distance between any two points from the same segment. Since $[d^i_{lat}, d^i_{lon}]$ are geodetic coordinates and Δ is a walking distance, φ_m is defined as the maximum Euclidean distance between the coordinates of any two points i, j in C_m. Note that for short distance, Euclidean and great-circle distance metrics such as Haversine yield similar results.

To address the problem of incorporating spatial and behavioural features with spatial constraints; as well as aiming for the large-scale implementation, we propose a modification of Affinity Propagation (Frey and Dueck, 2007) in this section.

3.1. Classical Affinity Propagation

Affinity Propagation (AP) is a relatively new clustering algorithm proposed by Frey and Dueck (2007). We adopt AP as the base algorithm because it (1) does not require a predetermined number of clusters, (2) supports a custom distance/similarity metric, thus potentially supports the integration of spatial and non-spatial features, and (3) it produces clusters with much less errors than similar algorithms (Frey and Dueck, 2007). AP aims to find 'exemplars' among data points to represent the dataset and forms clusters of around these exemplars. The input of the algorithm is pair-wise similarities between data points $s(i, j)$ $(i, j = 1...N)$. Given the similarity matrix, AP finds exemplars that maximise a net similarity, which is the overall sum of similarities between exemplars and their represented data points. It first considers all data points as exemplars, where each point k is assigned a value of 'preference' $s(k, k)$ that reflects how suitable it is to be chosen as an exemplar for itself. Data points then exchanges two kinds of messages to other points, which describe the current affinity that one data point has for choosing another data point as its exemplar, until a good set of exemplars converges.

The first type of message is called 'responsibility' $r(i, j)$, which is a message from point i to j with the accumulated evidence on how suitable it would be for point j to serve as the exemplar for point i. The second type of message is called 'availability' $a(i, j)$, which is a message from point j to i with the accumulated evidence on how suitable it would be for point i to choose point j as its exemplars. Both $r(i, j)$ and $a(i, j)$ are set to zero at the first iteration, and got iterated until convergence as follows:

$$r(i,\ k) = s(i,\ k) - \max_{k' \neq k}\{a(i,\ k') + s(i,\ k')\} \tag{1}$$

$$a(i,\ k) = \min\{0, r(k,\ k) + \sum_{i' \notin \{i,k\}} \max(0, r(i',\ k))\} \tag{2}$$

While the self availability $a(k, k)$ is updated as follows:

$$a(k,\ k) = \sum_{i' \neq k} \max\{0,\ r(i',\ k)\} \tag{3}$$

The algorithm converges after the clustering results remain stable after a number of iterations. Then for each point i, the value of k that maximises $a(i,k) + r(i,k)$ either means that point i is an exemplar, if $k = i$, or means that point k is the exemplar of point i, if $k \neq i$. To avoid numerical oscillations, a damping factor λ is also added to the calculation of $r(i,k)$ and $a(i,k)$.

$$r_{i+1}(i,\ k) = \lambda * r_i(i,\ k) + (1 - \lambda) * r_{i+1}(i,\ k) \tag{4}$$

$$a_{i+1}(i,\ k) = \lambda * a_i(i,\ k) + (1 - \lambda) * a_{i+1}(i,\ k) \tag{5}$$

For more details on the classical AP algorithm including a graphical example, interested readers may refer to the original paper by Frey and Dueck (2007).

3.2. Spatial Affinity Propagation

This section proposes a modification of the Classical Affinity Propagation to solve the two aforementioned problems: (1) incorporation of spatial and behavioural features, and (2) large-scale implementation. We name this algorithm the Spatial Affinity Propagation (SAP). Three main inputs of the SAP algorithm are (1) Passenger data, (2) Similarity metric and (3) Δ. The Passenger data, as described, should include a vector of spatial geodetic coordinates $[d^i_{lat}, d^i_{lon}]$ and a vector of behavioural features $[V^i_1, V^i_2, ..., V^i_D]$ for each passenger i. To ease the explanation of SAP, we define the $\Delta - neighbourhood$ and spatially-reachable as:

Definition 1. $\Delta-$*neighbourhood* $\phi(k)$ is a subset contains all passengers within a radius Δ around passenger k

Definition 2. Passenger i is *spatially-reachable* from passenger j and vice-versa if $i \in \phi(k)$

The workflow of SAP is illustrated in Figure 1.

The first step is calculation of the Initial Similarity matrix using a predefined Similarity metric. The Similarity metric, which defines how a passenger i is similar to a passenger j, is described in Equation 6.

$$s(i,k) = \begin{cases} \sum_{d=1}^{D} \gamma_d \left[1 - \frac{|V^i_d - V^k_d|}{|V^i_d| + |V^k_d|} \right] & i \in \phi(k) \\ 0 & otherwise \end{cases} \tag{6}$$

Equation 6 is an adaptation of the Canberra distance metric (Lance and Williams, 1967), where γ_d is a predefined weight for variable d, $\sum_{d=1}^{D} \gamma_d = 1$. The Canberra distance has been adopted because we expect a highly skewed behavioural features data $[V^n_1, V^n_2, ..., V^n_D]$. For instance, in areas near the train stations, there

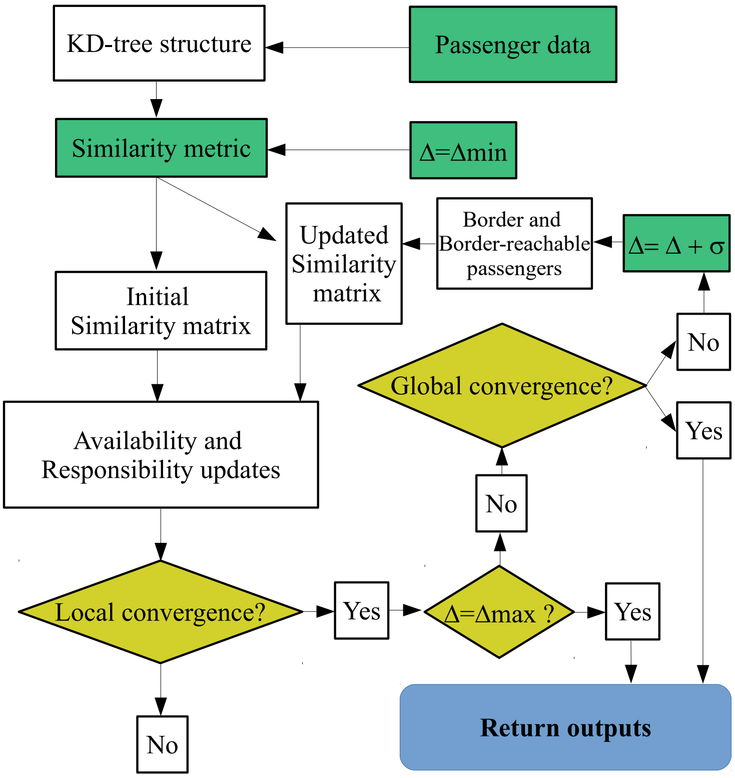


Figure 1: Spatial Affinity Propagation Algorithm

may be many more train riders than passengers of other modes. The adaptation of Canberra distance is introduced in Equation 6 to alleviate this problem, because the Canberra distance is the absolute difference between two data points over their sum. We did not use data transformation because it is easier for transit providers to interpret the values of $[V_1^n, V_2^n, ..., V_D^n]$ without data transformation. The weight γ_d defines the relative importance of variable d against other variables and allows transit providers to perform targeted market segmentation where some features are more important than the other. γ_d equals zero is equivalent to completely removal of the feature d for consideration.

Recall that $[V_1^n, V_2^n, ..., V_D^n]$ are to be normalised between [0,1], Equation 6 shows that $\max(s(i,k)) = 1$ and $\min(s(i,k)) = 0$. The value of $s(i,k)$ also equals zero if passenger i and k are not spatially-reachable, which means the two passengers will not be clustered in the same segment. Calculation of $s(i,k)$ for every $i,k \in N$ makes up the Initial similarity matrix. Thus Δ has been introduced in SAP as spatial constraints to limit the spatial sizes of each passenger segment. However, indexing the point membership for $s(i,k)$ calculation is very expensive in a naïve approach because each point k requires a neighbourhood search, i.e. calculating the spatial distances to every other points in the dataset. We implement the neighbourhood search and distance calculation using a Spatial Indexing algorithm. Spatial Indexing algorithms, such as KD-tree Bentley (1975), R-tree and R*-tree Beckmann et al. (1990) and Quadtree Samet (1984), organise the data space and data points in a clever way that only parts of the data needs to be considered in a neighbourhood search query. We adopt the KD-tree Bentley (1975) in the SAP algorithm because of its simplicity and availability in commonly used software packages.

The second step is a slightly modified Availability and Responsibility updates, which can be formulated as:

$$r(i,\ k) = s(i,\ k) - \max_{\substack{k' \neq k \\ k' \in \phi(i)}} \{a(i,\ k') + s(i,\ k')\} \tag{7}$$

$$a(i,\ k) = \min\{0, r(k,\ k) + \sum_{\substack{i' \notin \{i,k\} \\ i' \in \phi(k)}} \max(0, r(i',\ k))\} \tag{8}$$

$$a(k,\ k) = \sum_{\substack{i' \neq k \\ i' \in \phi(k)}} \max\{0,\ r(i',\ k)\} \tag{9}$$

Up until this point the value of Δ is deterministic, i.e. we choose a value of Δ, implement the modified AP, and collect the clustering results. However, recall that it is challenging to find a suitable value of Δ because it affects the spatial size, number of spatial singleton, the similarities of passengers within a segment, and the differences of passengers across segments. As Δ increases, each passenger is spatially-reachable to more passengers, facilitating the search for good clustering results to maximise the net similarity. Therefore, it is likely that the net similarity will increase each time Δ increases to $\Delta + \sigma$. Choosing the value of Δ is a trade-off between the net similarity and the spatial size of passenger segments. The best way to deal with this problem is actually to evaluate all possible values of Δ to see which is best for the dataset, and let Δ to be chosen by a transit operation expert.

In the Classical AP algorithm, segmentation outputs are only retrievable after a full implementation of the algorithm. One simple solution to evaluate different Δ would be to run AP multiple times, each time with a different value of Δ. However, one significant weakness of the Classical AP is its time complexity, which is believed to be $O(N^2T)$, where N represents the number of data points and T represents the number of iterations. If there are $\mathcal{D}$ values of Δ needs to be evaluated, the time complexity will be $O(\mathcal{D} \times N^2T)$.

Therefore, we introduce another rigorous, yet flexible extension of the Classical AP to incorporate a variable Δ, which varies between predefined values for minimum Δmin and maximum Δmax. The value of Δmax is the maximum walking distance that is considered acceptable for most passengers, while Δmin is the minimum size of each passenger segment. Our algorithm generally starts at $\Delta = \Delta min$. We then perform Availability and Responsibility updates using Equation 7 to 9, until a Local convergence is reached. The Local convergence criteria is satisfied when the changes in segmentation results are less than or equal to $\mathcal{C}$ after a certain number of iteration. Note that this is a generalisation of the Classical AP, where $\mathcal{C}$ equals zero will ensure the same convergence criteria as in Frey and Dueck (2007). The convergence variable $\mathcal{C}$ is introduced to generalise the algorithm to allow easier convergence in when there are numerical oscillating and volatility; or when computation time is an important factor.

The next step is to increase the value of Δ by a step σ, and index all passengers who are now spatially-reachable to new passengers compared to the previous step. These are the passengers that may have their segmentation changed as the result of the maximum spatial size increases from Δ to $\Delta + \sigma$. An example of changes in the $\Delta - neighbourhood$ is illustrated in Figure 2.

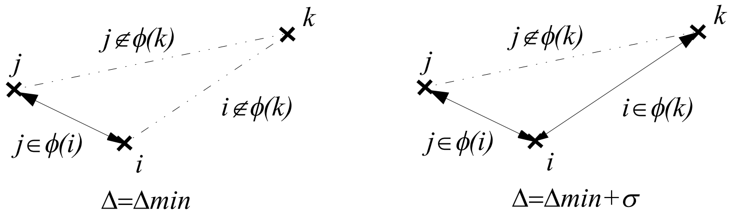


Figure 2: An example of changes in $\Delta - neighbourhood$

Figure 2 shows an example where passenger i was first only spatially-reachable to passenger j as $\Delta = \Delta min$, but after that also spatially-reachable to passenger k as Δ increases to $\Delta min + \sigma$. We introduce here two other definitions to help explaining the SAP algorithm.

Definition 3. *Border passengers* are those who are spatially-reachable to new passengers as a result of the change in the maximum spatial size Δ

Definition 4. *Border-reachable passengers* are those who are not Border passengers, but are spatially-reachable to Border passengers.

In the example in Figure 2, passenger i and k are Border passengers, and passenger j is a Border-reachable passenger. Note that since every point will reach to a further distance $\Delta min + \sigma$, any point can be a Border or Border-reachable passenger.

Compared to the Classical AP algorithm, SAP does not reset all Availability and Responsibility matrices to zero and start a fresh Availability and Responsibility updates as Δ increase to $\Delta + \sigma$, but only resets the Availability of Border and Border-reachable passengers. The algorithm carries the Responsibility matrix and the non-Border-reachable passengers' Availability matrix forward to accelerate a Local convergence, because passengers who are exemplars in the previous step of Δ are likely to be exemplars again. We still perform Availability and Responsibility updates of every passengers because passengers who are not the exemplars in the previous step may be an exemplar in the next time step. The SAP algorithm continues until $\Delta = \Delta max$ or a Global convergence criteria has been reached. The Global convergence criteria is similar to the Local convergence, which is when the segmentation results remain stable for a certain number of steps of Δ.

Recall that an increase in Δ will also facilitate the search for passenger segments, our hypothesis is that the algorithm will find near-optimum solution quicker with less numerical oscillating and volatility. We further introduce an adaptive Local convergence criteria to enhance SAP's efficiency in large-scale implementation. Algorithm 1 is a Local converge check algorithm with adaptive convergence threshold $\mathcal{C}$.

Algorithm 1: Adaptive Local convergence criteria

```
Input: Segmentation results $idx$,
Convergence threshold $\mathcal{C}$,
Convergence iteration count threshold $convits$,
$\max(netsim)$ = Net similiarity of the previous Local convergence,
$Unconverged$ = Yes
while $Unconverged = Yes$ do
    Calculate the current net similiarity $netsim_t$ by Availability and Responsibility updates;
    if $netsim_t > \max(netsim)$ then
        $\mathcal{C} = \rho^{\frac{netsim_t - \max(netsim)}{\max(netsim)}} + \mathcal{C}$
        $\max(netsim) = netsim_t$
    if $Count\ of\ unchanged\ idx \leq 1 - \mathcal{C}$ then
        Convergence iteration count + 1 ;
        if $Convergence\ iteration\ count \geq convits$ then
            Output: $Unconverged$ = No
```

Parameter ρ controls the convergence rate, with SAP reaches the convergence faster if ρ is larger, and ρ equals zero will keep $\mathcal{C}$ unchanged, similar to the Classical AP. The idea of using ρ is to adaptively relax the convergence criteria if a better solution than the previous converged net similarity has been found.

4. A framework to Spatial Transit Passenger Segmentation

This section describes the process to implement the proposed SAP algorithm using observed Smart Card data.

4.1. Dataset

This paper uses a 40-day Smart Card AFC data from New South Wales, Australia (NSW). The data consists of over 2.4 million of Smart Card over large metropolitan areas in NSW, including Sydney, Newcastle and Wollongong City. Each row of the dataset includes a hashed unique card ID, tap-on and tap-off locations and timestamps of every Smart Card transactions over the study period from February to March 2017. The data consists all the public transport modes and routes in NSW, which are bus, city train, ferry and light rail system.

We focus on the passengers who have at least 10 journeys over the 40-days study period to increase the reliability of data. The 10 journeys threshold has been chosen because of the nature of the Smart Card data in practice, where a passenger may hold several Smart Cards, and there are many less active Smart Cards from private transport users. The data used for the analysis contains 587,791 passengers satisfying this requirement.

4.2. Spatial features extraction

This section aims to find the spatial geodetic coordinates $[d^n_{lat}, d^n_{lon}]$ of each passenger n ($n = 1...N$). The challenge comes when we do not have the actual home locations of Smart Card users in our AFC data. Therefore, we take the most frequent first transit stop that a passenger usually take in the morning, and all other stops within the an immediate vicinity of 500m from this stop. Without any loss of generality of the proposed algorithm, we assume that the calculated centroid of all these stops is also the passenger's home location. The centroid is used instead of the most frequent stop itself because it allows some flexibility in passenger's route and stop choice behaviours, as well as to increase in diversity in the spatial dataset. In fact, it is a real data challenge to obtain the actual locations of Smart Card users due to privacy issues related to tracking passengers' home locations. Figure 3 shows the estimated home locations of transit passengers in New South Wales, Australia.

4.3. Behavioural features extraction

The behavioural features in this study represent the travel behaviour of each passenger. For each individual passenger, they are Frequency of use, Randomness of travel behaviour, Frequency of Train usage, and Frequency of Transfers. They are calculated as follows:

- Frequency of use (V^n_1) is the number of days that passenger n made at least one journey over the total number of days (40 days). For transit operator, this feature provides a reliable prediction for patronage, because frequency of use is generally the probability of passenger n to use public transport on the study day.
- Randomness of travel behaviour (V^n_2) is the number of unique tap-on and tap-off pairs that the passenger n travelled over the overall number of journeys. For transit operator, this feature enables an prediction of demand mobility, such as Origin-Destination matrix, because regular passengers are more likely to maintain a travel pattern.
- Frequency of Train usage (V^n_3) is the number of train journeys that the passenger n made over the overall number of journeys. This feature is a representation of the modal choice.
- Frequency of Transfers (V^n_4) is the number of journeys with at least one transfer that the passenger n made over the overall number of journeys. This feature is generally a representation of the route

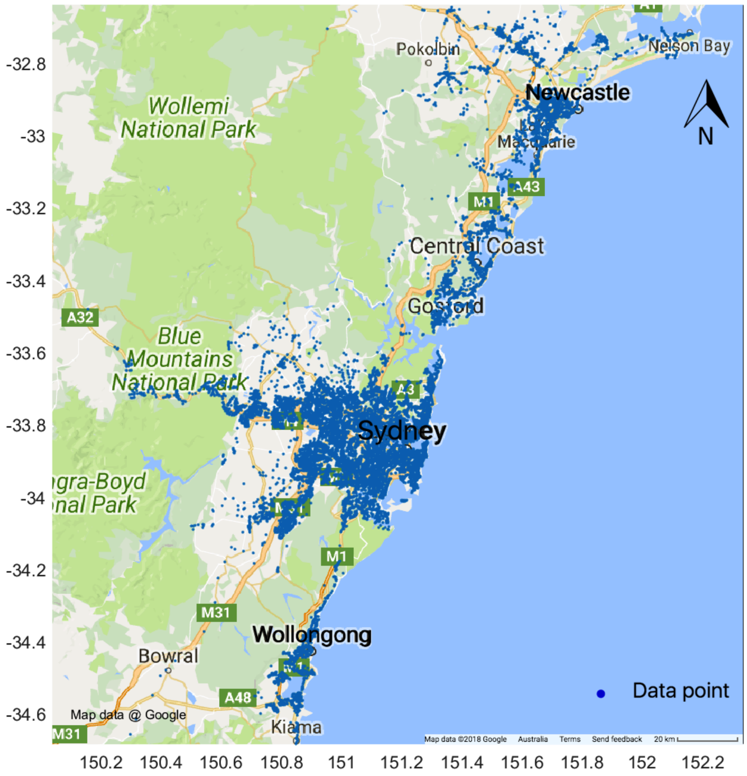


Figure 3: The spatial locations of transit passengers in the dataset

choice, to see if the passenger is willing to take an alternative with transferring or not. It also shows the connectivity characteristic of the transit network.

5. Numerical case studies

This section describes two numerical case studies to show the passenger segmentation results from SAP. The first case study is a numerical experiment of only over 4000 passengers, which aims to compare the proposed Spatial Affinity Propagation (SAP) with the Classical Affinity Propagation (AP) and two popular algorithms: K-means (Hartigan and Wong, 1979) and Hierarchical Agglomerative Clustering (Johnson, 1967). The second case study is a large-scale study using the whole dataset of 587,791 passengers.

5.1. First case study: a limited sample experiment for comparison

This section compares the proposed Spatial Affinity Propagation (SAP) with the K-means, Hierarchical Agglomerative Clustering and Classical Affinity Propagation (AP). To simplify the comparison, we implement the algorithms on a subset of the whole dataset where the Latitude d_{lat}^{n} is larger than -32.8 degree. The comparison subset is 4031 passengers. It is fairly straight forward to choose the range $[\Delta min, \Delta max]$. Δmin should be large enough so that clustering makes sense, so we set Δmin equals 500 m. Δmax should be the maximum walking distance, so Δmax equals 1500m in this case study. The spatial step has been chosen as 200m.

First, we adopt a bi-level clustering approach for both K-means (Hartigan and Wong, 1979) and Hierarchical Agglomerative Clustering (Defays, 1977), which are called K-K and H-H, respectively. A major challenge comes in implementation of K-means and Hierarchical Agglomerative Clustering is to choose the number of clusters. There are multiple solutions to this problem and among them we adopt the Silhouette index, which is a popular, proven and effective method to find the number of clusters (Rousseeuw, 1987). Silhouette statistic for each passenger n is calculated as follows:

$$s(n) = \frac{b(n) - a(n)}{\max\{a(n), b(n)\}} \tag{10}$$

Where $a(n)$ is the mean distance between passenger n and all other passengers within the same cluster, and $b(n)$ is lowest mean distance between passenger n and all other cluster where n is not a member. The mean value of all $s(n)$ $(n = 1, 2, ...N)$ is the mean Silhouette index.

The first level of both K-K and H-H is spatial only, i.e. clustering of geodetic coordinates $[d^n_{lat}, d^n_{lon}]$. For the first level of H-H, we adopt the Complete-linkage Hierarchical Agglomerative Clustering, and use the maximum walking distance Δ as the cut-tree threshold to limit the spatial size of each cluster to be Δ, which will fit well to the Criterion 1 as described in Section **??**. For the first level of K-K, we run the K-mean algorithm with the number of clusters varies from small to very large, and then choose the outputs that (1) maximise the mean Silhouette index, (2) has the maximum spatial size of segments to be smaller than Δ, and (3) minimise the number of clusters. Euclidean distance is used as the distance metric in the first level.

The second level of both K-K and H-H deals with behavioural features only $[V^n_1, V^n_2, ..., V^n_D]$. We implement K-means and Complete-linkage Hierarchical Agglomerative Clustering multiple times until the Silhouette index is maximised.

Second, we implement the Classical Affinity Propagation (AP) as described in Frey and Dueck (2007). The calculation of the Similarity matrix follows Equation 6 using KD-tree structure for indexing spatially-reachable passengers. The algorithm is implemented multiple times for each value of Δ from Δmin to Δmax. The Local convergence criteria strictly follows Frey and Dueck (2007), which means $\mathcal{C}$ equals zero in this experiment.

Third, we implement a slight modification of the Classical AP by introducing the Local convergence threshold $\mathcal{C}$. Multiple experiments show that a small value of $\mathcal{C}$ is sufficient to enhance the computation time. Thus $\mathcal{C}$ is set equal 0.005. This algorithm is called AP-C. The algorithm is also implemented multiple times for each value of Δ from Δmin to Δmax.

Fourth, we implement another modification of the Classical AP by introducing the variable Δ. This algorithm is closer to SAP than AP, because similar to SAP, it only updates the Availability and Responsibility of non-border-reachable passengers after the first step of Δ. $\mathcal{C}$ is also set equal 0.005. Note that we only have to run this algorithm once with a variable Δ from Δmin to Δmax. This algorithm is called SAP-C.

Finally, the Spatial Affinity Propagation (SAP) is implemented as described in Section 3.2. The adaptive convergence is implemented with ρ equals 0.1 and $\mathcal{C}$ starts at 0.005. Only one run of SAP is required for a variable Δ between Δmin and Δmax. Table 2 summaries the algorithms for comparison in this experiment.

Table 2: Description of the algorithms in comparison.

Algorithm	Description	Objective function
K-K	K-means clustering on geodetic coordinates first, then another K-means clustering on the behavioural features	Silhouette index
H-H	Hierarchical Clustering on geodetic coordinates first, then another Hierarchical Clustering on the behavioural features	Cut-tree & Silhouette index
AP	KD-tree structure, multiple runs with different Δ	Net similarity
AP-C	Similar to AP, but with Local convergence threshold $\mathcal{C}$ equals 0.005	Net similarity
SAP-C	Similar to AP-C, but with variable Δ and selective Availability and Responsibility updates	Net similarity
SAP	Similar to SAP-C, but with adaptive $\mathcal{C}$	Net similarity

Figure 4 shows an example of the segmentation results from K-K and H-H algorithms on an area of mixed population density. Each colour represents a spatial passenger segment. For segments of more than 2 spatial locations, we also plot and fill a convex hull of the area that the segment covers. Thus in Figure 4, a point without bounded envelope is a spatial singleton. From a segmentation point of view, spatial singletons are unfavourable because they are clusters of a single data point.

Figure 4 shows that both K-K and H-H produces a significant amount of spatial singleton. While more clusters can be found as Δ increases from 500m to 1500m, spatial singletons still dominates other segments. Figure 5 shows examples of segmentation results from AP and SAP for the same area.

Figure 5 shows that AP and SAP produce more meaningful segments compared to K-K and H-H. There are much less spatial singleton observed, especially at Δ equals 1000 and 1500m. Figure 5 clearly shows that AP and SAP are better candidates for our spatial passenger segmentation problem according to the Criterion 1. Visually, there are little differences between the segmentation results of AP and SAP. We look further into the differences between AP and SAP in Table 3, where the net similarity and total computation time of AP, AP-C, SAP-C and SAP at different values of Δ are presented.

Table 3: Comparison of net similarity and computation time of AP, AP-C, SAP-C and SAP

Criteria		AP	AP-C	SAP-C	SAP
Net similarity at different Δ (m)	500	3650.84	3650.65	3650.10	3650.81
	700	3689.18	3685.17	3688.96	3684.57
	900	3716.54	3709.69	3715.51	3714.50
	1100	3741.62	3739.22	3737.86	3740.35
	1300	3756.29	3751.95	3755.43	3753.19
	1500	3765.22	3762.99	3764.91	3764.50
Total computation time (s)		353	237	186	168

Table 3 shows that the four algorithms start very similarly at Δ equals 500m, where all of them produce a similar net similarity of approximately 3650. As Δ increases from 500 to 1500m, AP generally produces the highest net similarity. However, while AP takes 353 seconds for 6 full implementation as Δ increases from 500 to 1500m with step σ equals 200m, other algorithms are significantly more efficient, while producing very close clustering performance. Compared to AP, other algorithms (AP-C, SAP-C and SAP) are 38%, 47% and 52% faster, respectively, while all producing less than 1% lower in net similarity. The difference between the 4 algorithms are further demonstrated in Figure 6.

Figure 6 shows the progressions in each iterations of AP, AP-C, SAP-C and SAP at different value of Δ. Figure 6 shows that without the Local convergence threshold $\mathcal{C}$, AP usually takes the maximum number of iteration (500 iterations) to converge. This is due to the numerical oscillation and volatility in the net similarity progression, even though a very large damping factor (Equation 4 and 5) of 0.9 has been used in all algorithms. The introduction of parameter $\mathcal{C}$ is necessary in our data because of this nature of the data, and because computational efficiency is also important for our problem. In fact, Table 3 shows that the introduction of $\mathcal{C}$ saves at least 38% of computation time by enabling a faster Local convergence in AP-C, SAP-C and SAP.

As expected, Figure 6 also shows that there are less numerical oscillation and volatility as Δ increases. Adaptation of $\mathcal{C}$ (Algorithm 1) enables SAP to converge faster than SAP-C, while maintaining similar net similarity. Figure 6 shows that while SAP-C takes 130 and 100 iterations to converge when Δ equals 1000m and 1500m, SAP only takes 100 and 70 iterations, respectively.

Compared to AP and AP-C, SAP-C and SAP take much less iterations to converge. This is because SAP-C and SAP carry the Responsibility and a selective part of Availability matrix forward as Δ increases to accelerate a Local convergence. Figure 6 shows that as Δ increases from 500m to 1500m, while AP and AP-C always start the net similarity progression from approximately 2400, the first iteration in SAP-C and SAP has a head-start from the previous step of Δ. In fact, SAP-C and SAP starts the net similarity progression of $\Delta = 1000$m from 2800, compared to 2400 in AP and AP-C.

The first case study on an limited sample size shows that AP and SAP are better candidates for spatial passenger segmentation than K-means and Complete-linkage Hierarchical Agglomerative Clustering. AP, SAP and their variety AP-C and SAP-C produce very similar clustering performance, measured in the sum of net similarity between passengers and their exemplars, but SAP takes 52% less in computation time using the same Windows machine.

5.2. Second case study: a large scale study of transit passengers in New South Wales, Australia

This section describes the process to implement the proposed SAP algorithm on the whole dataset of 587,791 passengers. We also aim to show several examples of usages for the segmentation results to demonstrate how this study would be useful for them. The study process for the large-scale case study follows what has been described in Section 4.

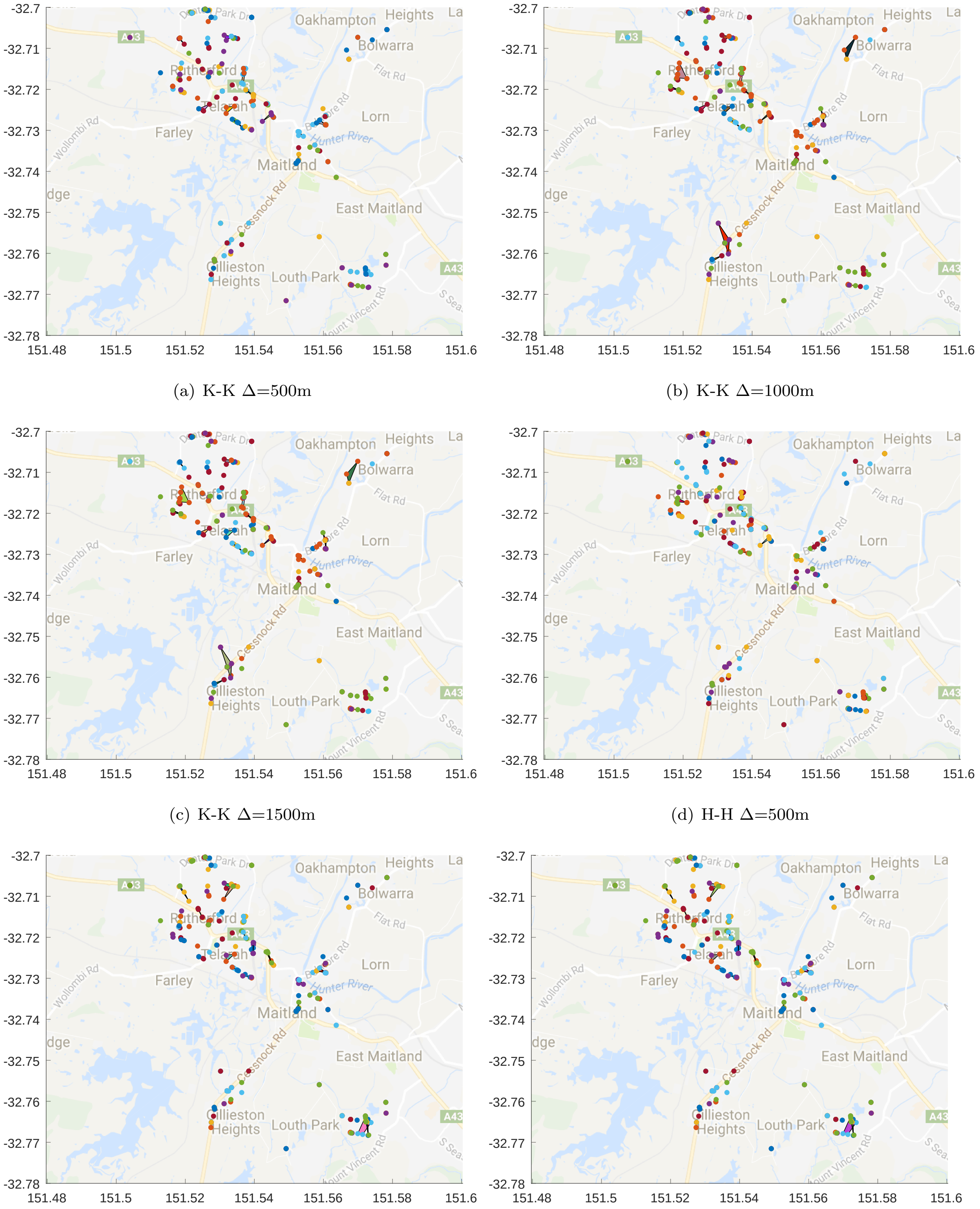
Figure 7 shows the spatial distribution of passenger segments in two different spatial areas of Sydney, Australia. The first area is Inner South Sydney, an area of high population density and well-serviced public transport. The second area is Northern Beaches, an area of low population density and limited public transport options, with only local buses servicing the area. SAP is implemented with Δ equals 700m.

For visualisation purpose, each segment of passengers is colour-coded according to the mean of each behavioural features, where Red represents the mean value being larger than 50%, and Green represents the mean value being smaller than 50%. It is noticeable in Figure 7 that Inner South Sydney has significantly more public transport passengers than Northern Beaches, due to its high population density and better transit service. There are significantly more segments of Frequent and Train passengers in Inner South Sydney than Northern Beaches, while Northern Beaches' passengers often have to make more transfers than Inner South Sydney's passengers. The results show the interplay between supply and demand, where better supply and higher population density bring more demand. However, Figure 7 also shows the dominance of Infrequent segments of passengers in both areas, notwithstanding the good public transport service in Inner South Sydney. In fact, segments which are close to train stations are more likely to be Frequent and vice versa, suggesting the transit operators to address the needs of passengers at different spatial locations with different travel patterns. For instance, transit operators may open feeder services to take Infrequent Train passengers segments to the nearest train station, or provide incentives for them to encourage more Train usage. SAP enables transit operators to discover these spatial passenger segments, and to find out the changes in passenger market segments before and after a new policy without a costly passenger survey.

A similar approach can be implemented for Bus Bridging services, which are temporary bus services that serve Train passengers during a Train disruptions. To limit the resources needed and maximise the Bus Bridging vehicle's occupancy, passengers can be asked to walk to the centre points of each spatial passengers segment, where they will be served by a Bus Bridging vehicle. Depends on the available resources and the passengers' acceptance for a maximum walking distance, a value of Δ can be chosen. Larger Δ means the segments' spatial sizes are larger, so that less Bus Bridging vehicles are needed, but also longer walking time for passengers. Figure shows the segments of Train users in the Inner South Sydney area at different values of Δ. As could be seen in the figure, larger value of Δ means less spatial passenger segments and less spatial singleton, but also means longer mean walking distance for passengers.

Another example of a SAP's benefit on areas of limited public transport supply and demand such as the Northern Beaches area in Sydney. The idea is to approach the supply-demand interplay from a demand point of view. Traditional public transport deals with coarsely estimated long-term transit demand and aims to serve the demand by deterministic supply of fixed routes and timetables. The availability of data-driven spatial passenger segmentation methods such as SAP may open a more versatile approach to meet the demand by adapting the transit services to the spatial locations and travel patterns of passengers. Similar to the above Bus Bridging service example, buses may serve the centre point of each spatial passenger segment. Large volume and sparse stop-to-stop transit Origin-Destination (OD) demand can now be represented as segment-to-segment OD. Passengers will walk to the "virtual stops" at the centre of their spatial segment, where they will share the same service with other passengers who travel to similar locations, and having the same mobility requirements. The locations of virtual stops and the planned services can be changed dynamically to adapt to passengers' needs using the spatial passenger segmentation results from SAP. While the exact modelling and implementation of this approach are outside the scope of this paper, Figure 9 gives an insight into the spatial segments at different values of Δ.

The second case study is about the large-scale implementation of the proposed SAP algorithm on the whole dataset of NSW, Australia. The case study also shows several examples where we can localise the segmentation results into specific spatial areas for a number of applications.



(a) K-K Δ=500m

(b) K-K Δ=1000m

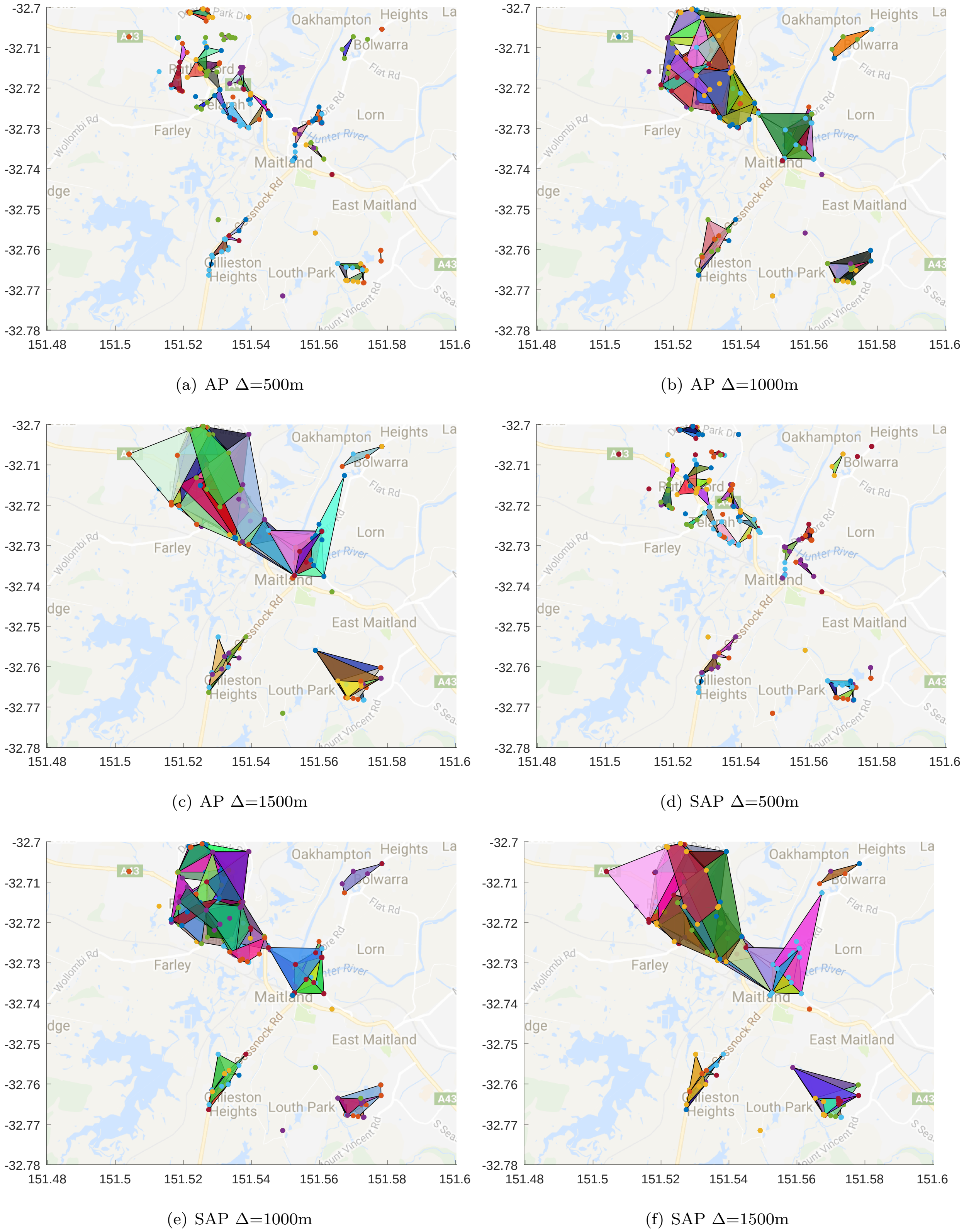
(c) K-K Δ=1500m

(d) H-H Δ=500m

(e) H-H Δ=1000m

(f) H-H Δ=1500m

Figure 4: Comparison of clustering results: K-K & H-H. Each colour and convex hull represents a spatial passenger segment. Map data at Google.



(a) AP Δ=500m

(b) AP Δ=1000m

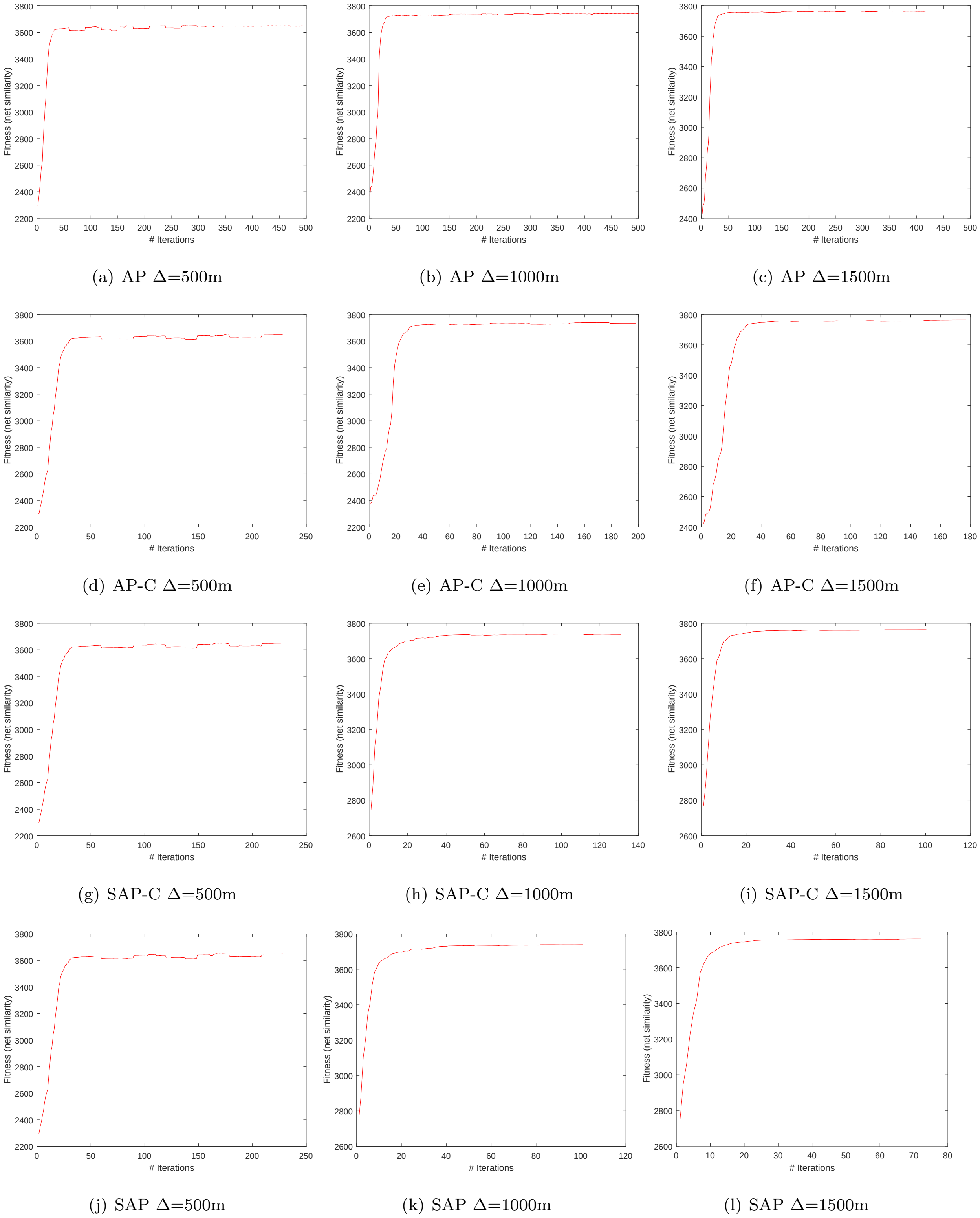
(c) AP Δ=1500m

(d) SAP Δ=500m

(e) SAP Δ=1000m

(f) SAP Δ=1500m

Figure 5: Comparison of clustering results: AP & SAP. Each colour and convex hull represents a spatial passenger segment. Map data at Google.



(a) AP Δ=500m

(b) AP Δ=1000m

(c) AP Δ=1500m

(d) AP-C Δ=500m

(e) AP-C Δ=1000m

(f) AP-C Δ=1500m

(g) SAP-C Δ=500m

(h) SAP-C Δ=1000m

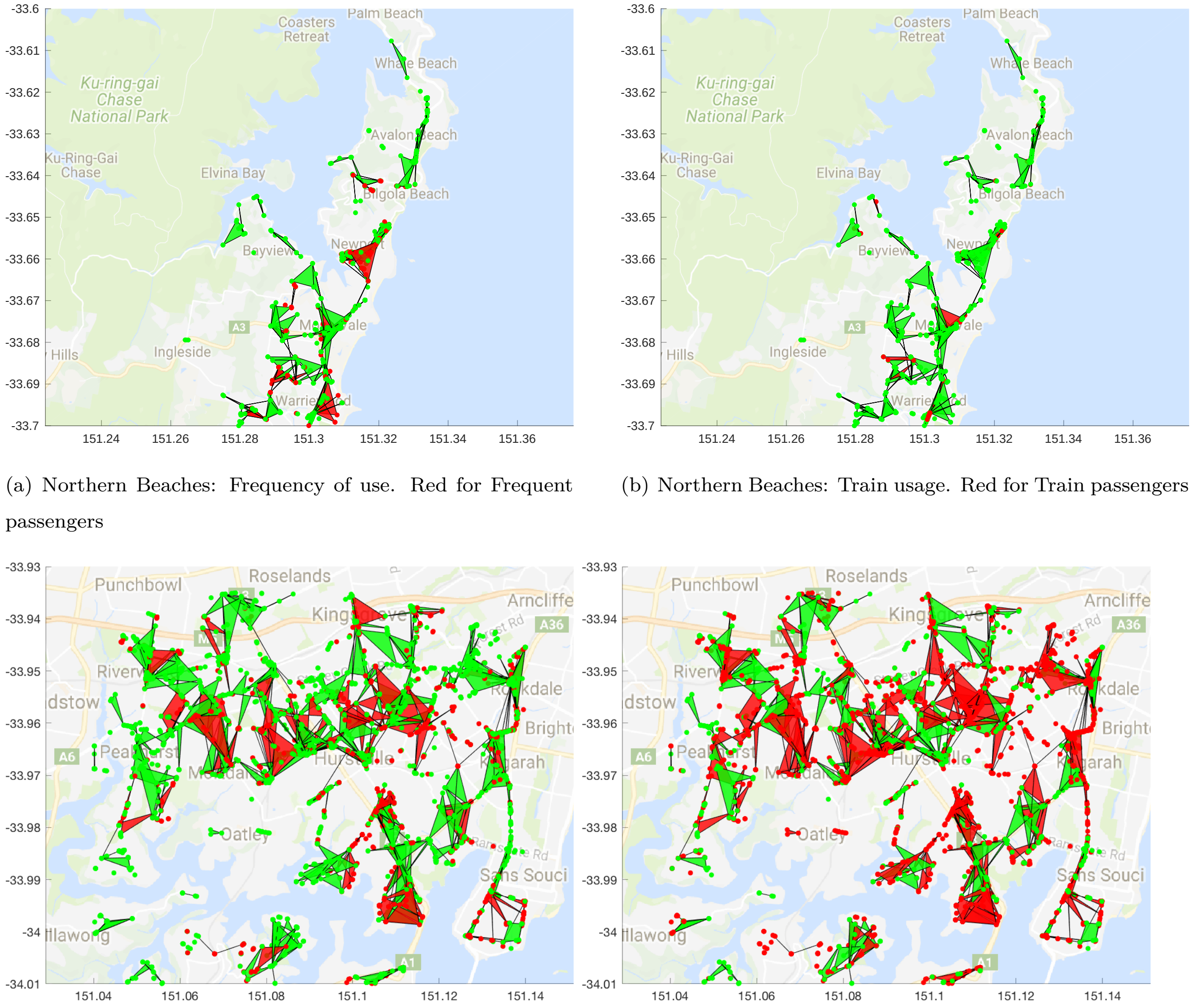
(i) SAP-C Δ=1500m

(j) SAP Δ=500m

(k) SAP Δ=1000m

(l) SAP Δ=1500m

Figure 6: Comparison of net similarity progression.



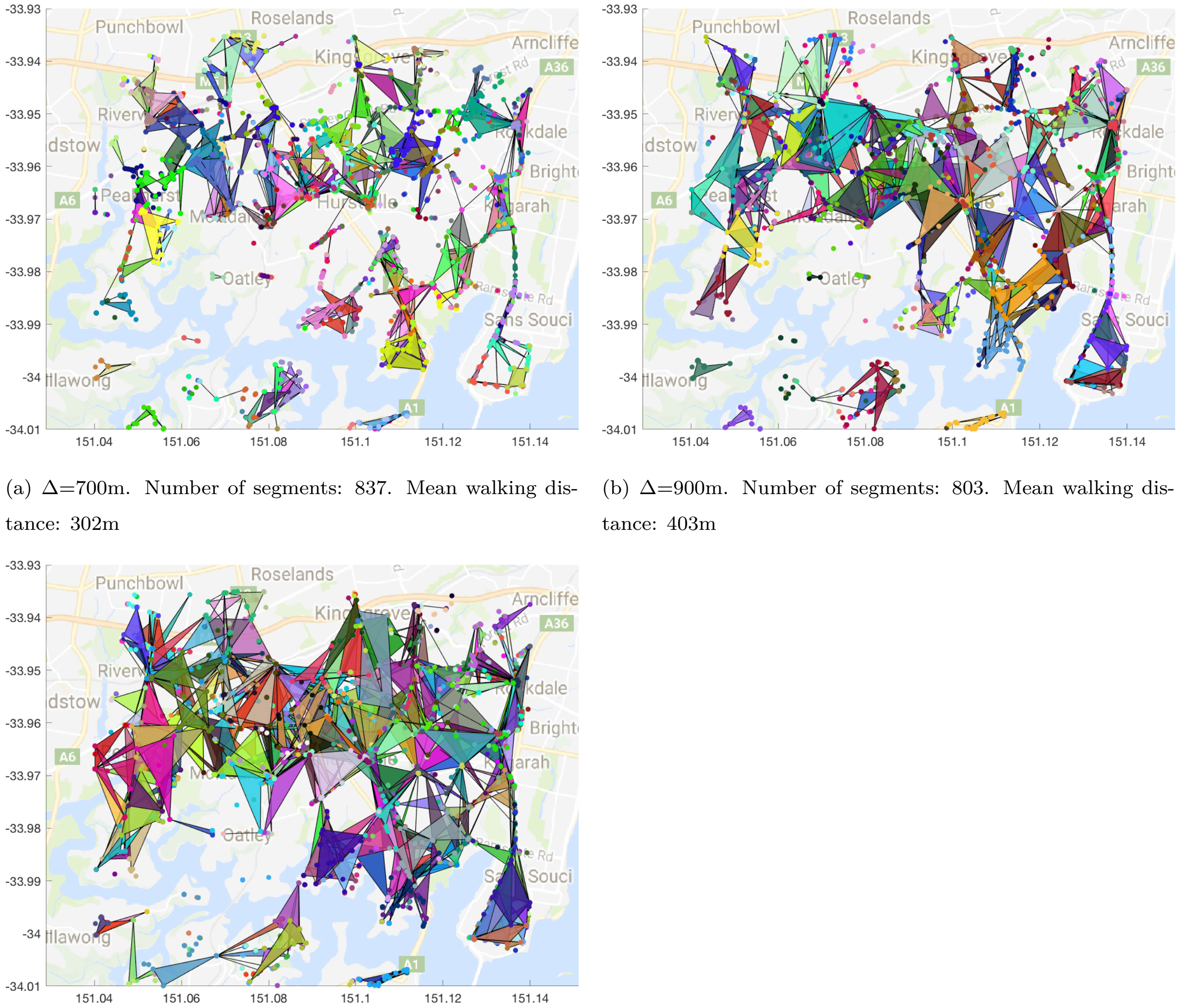
(a) Northern Beaches: Frequency of use. Red for Frequent passengers

(b) Northern Beaches: Train usage. Red for Train passengers

(c) Inner South: Frequency of use. Red for Frequent passengers

(d) Inner South: Train usage. Red for Train passengers

Figure 7: SAP results on Northern Beaches and Inner South Sydney area. Map data @ Google.

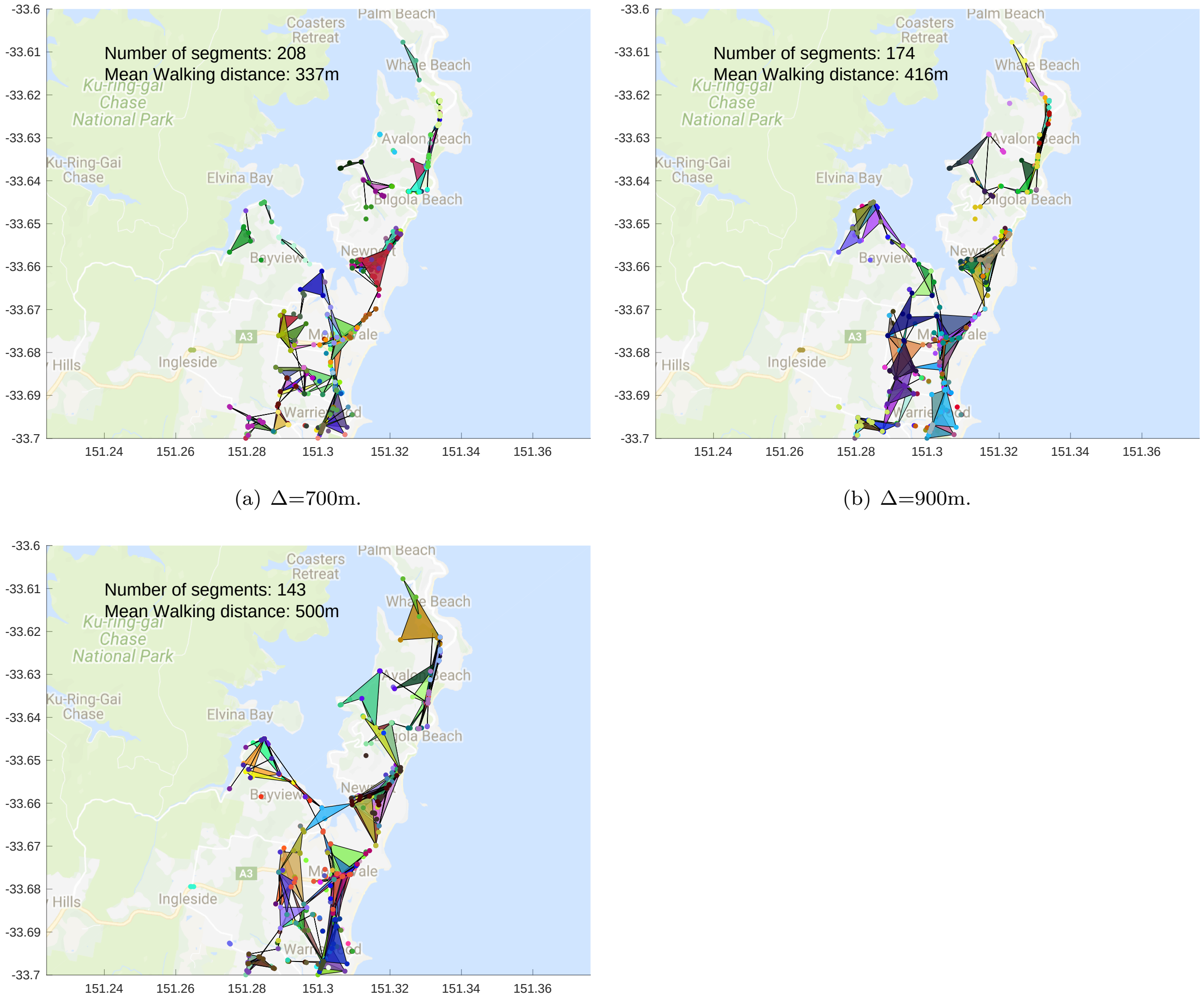


(a) Δ=700m. Number of segments: 837. Mean walking distance: 302m

(b) Δ=900m. Number of segments: 803. Mean walking distance: 403m

(c) Δ=1100m. Number of segments: 774. Mean walking distance: 473m

Figure 8: Spatial Passenger Segmentation at the Inner South Sydney area. Each colour and convex hull represents a spatial passenger segment. Map data @ Google.



(a) Δ=700m.

(b) Δ=900m.

(c) Δ=1100m.

Figure 9: Spatial Passenger Segmentation at the Northern Beaches area. Each colour and convex hull represents a spatial passenger segment. Map data @ Google.

6. Conclusion

This paper proposes a new algorithm named Spatial Affinity Propagation (SAP) based on the classical Affinity Propagation algorithm (AP) for large-scale spatial-behavioural transit market segmentation. The proposed SAP algorithm (1) incorporates both spatial and behavioural features into a comprehensive clustering methodology; and (2) enables large-scale transit market segmentation by a variable maximum spatial size threshold Δ. The proposed SAP algorithm clusters passengers based on their spatial geodetic coordinates, where passengers from the same segment are located within a walking distance from each other; and based on their behavioural features, where Frequency of use, Randomness of travel behaviour, Frequency of Train usage, and Frequency of Transfers are mined from Smart Card AFC data. The algorithm can be implemented very efficiently with a selective rules for Availability and Responsibility updates; and an adaptive convergence criteria at different values of Δ. On our numerical case studies, SAP is 52 % more efficient than the classical AP algorithm, while does not significantly compromise the clustering performance.

The proposed SAP algorithm will enable transit operators to understand the spatial-behavioural distributions of their customers. Customised and better catered services can be given to transit passengers who are spatially close to each other and behave similarly. Examples of them include a feeder service that takes passengers to the nearest transit hubs, Bus-Bridging services during train disruptions, or to adapt a bus service with dynamic demand.

Due to the limitation of Smart Card data, we used an estimation of passengers' home location from the most frequent first stop and the ones in the immediate vicinity instead of the actual home location for each passenger. Transit operators may be able to implement the proposed SAP algorithm using the actual home locations collected from their customers. Future developments of this paper include the incorporation of more spatial and behavioural features. Spatial features may include a regular origin-destination locations of each passengers. Behavioural features may include travel distance, travel time and other travel choice-related features. Simple adaptation will also allow this method to be applied to other problems, such as customer segmentation in banking.

7. Acknowledgement

This study is supported by Strategic Research Funding at the Data61, CSIRO, Australia and partly by the NSW Premier's Innovation Initiatives. The data used in this study is provided by the Transport for NSW, the transport operator of NSW, Australia. The conclusions in this paper reflect the understandings of the authors, who are responsible for the accuracy of the data.

References

Agard, B., Morency, C., and Trépanier, M. (2006). Mining public transport user behaviour from smart card data. *IFAC Proceedings Volumes*, 39(3):399–404.

Alfred Chu, K. and Chapleau, R. (2008). Enriching archived smart card transaction data for transit demand modeling. *Transportation Research Record: Journal of the Transportation Research Board*, (2063):63–72.

Bagchi, M. and White, P. R. (2005). The potential of public transport smart card data. *Transport Policy*, 12(5):464–474.

Beckmann, N., Kriegel, H.-P., Schneider, R., and Seeger, B. (1990). The r*-tree: an efficient and robust access method for points and rectangles. In *Acm Sigmod Record*, volume 19, pages 322–331. ACM.

Bentley, J. L. (1975). Multidimensional binary search trees used for associative searching. *Communications of the ACM*, 18(9):509–517.

Birant, D. and Kut, A. (2007). St-dbscan: An algorithm for clustering spatial–temporal data. *Data & Knowledge Engineering*, 60(1):208–221.

Briand, A.-S., Côme, E., Trépanier, M., and Oukhellou, L. (2017). Analyzing year-to-year changes in public transport passenger behaviour using smart card data. *Transportation Research Part C: Emerging Technologies*, 79:274–289.

Campello, R. J., Moulavi, D., and Sander, J. (2013). Density-based clustering based on hierarchical density estimates. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 160–172. Springer.

Defays, D. (1977). An efficient algorithm for a complete link method. *The Computer Journal*, 20(4):364–366.

El Mahrsi, M. K., Côme, E., Oukhellou, L., and Verleysen, M. (2017). Clustering smart card data for urban mobility analysis. *IEEE Transactions on Intelligent Transportation Systems*, 18(3):712–728.

Ester, M., Kriegel, H.-P., Sander, J., Xu, X., et al. (1996). A density-based algorithm for discovering clusters in large spatial databases with noise. In *KDD*, volume 96, pages 226–231.

Frey, B. J. and Dueck, D. (2007). Clustering by passing messages between data points. *science*, 315(5814):972–976.

Hartigan, J. A. and Wong, M. A. (1979). Algorithm as 136: A k-means clustering algorithm. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 28(1):100–108.

Johnson, S. C. (1967). Hierarchical clustering schemes. *Psychometrika*, 32(3):241–254.

Kaufman, L. and Rousseeuw, P. J. (1990). Partitioning around medoids (program pam). *Finding groups in data: an introduction to cluster analysis*, pages 68–125.

Kieu, L. M., Bhaskar, A., Almeida, P. E., Sabar, N. R., and Chung, E. (2017). Transfer demand prediction for timed transfer coordination in public transport operational control. *Journal of Advanced Transportation*.

Kieu, L.-M., Bhaskar, A., and Chung, E. (2015a). A modified density-based scanning algorithm with noise for spatial travel pattern analysis from smart card {AFC} data. *Transportation Research Part C: Emerging Technologies*, 58, Part B:193 – 207. Big Data in Transportation and Traffic Engineering.

Kieu, L. M., Bhaskar, A., and Chung, E. (2015b). Passenger segmentation using smart card data. *IEEE Transactions on Intelligent Transportation Systems*, 16(3):1537–1548.

Kusakabe, T. and Asakura, Y. (2014). Behavioural data mining of transit smart card data: A data fusion approach. *Transportation Research Part C: Emerging Technologies*, 46:179–191.

Lance, G. N. and Williams, W. T. (1967). Mixed-data classificatory programs i - agglomerative systems. *Australian Computer Journal*, 1(1):15–20.

Langlois, G. G., Koutsopoulos, H. N., and Zhao, J. (2016). Inferring patterns in the multi-week activity sequences of public transport users. *Transportation Research Part C: Emerging Technologies*, 64:1–16.

Lathia, N., Smith, C., Froehlich, J., and Capra, L. (2013). Individuals among commuters: Building personalised transport information services from fare collection systems. *Pervasive and Mobile Computing*, 9(5):643–664.

Legara, E. F. T. and Monterola, C. P. (2017). Inferring passenger types from commuter eigentravel matrices. *Transportmetrica B: Transport Dynamics*, pages 1–21.

Li, Y., Wang, X., Sun, S., Ma, X., and Lu, G. (2017). Forecasting short-term subway passenger flow under special events scenarios using multiscale radial basis function networks. *Transportation Research Part C: Emerging Technologies*, 77:306–328.

Ma, X., Wu, Y.-J., Wang, Y., Chen, F., and Liu, J. (2013). Mining smart card data for transit riders' travel patterns. *Transportation Research Part C: Emerging Technologies*, 36:1–12.

Ma, Z., Xing, J., Mesbah, M., and Ferreira, L. (2014). Predicting short-term bus passenger demand using a pattern hybrid approach. *Transportation Research Part C: Emerging Technologies*, 39:148–163.

Min, Y.-H., Ko, S.-J., Kim, K. M., and Hong, S.-P. (2016). Mining missing train logs from smart card data. *Transportation Research Part C: Emerging Technologies*, 63:170–181.

Ng, R. T. and Han, J. (1994). Efficient and effective clustering methods for spatial data mining. In *Proceedings of the 20th International Conference on Very Large Data Bases*, VLDB '94, pages 144–155, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Rousseeuw, P. J. (1987). Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of computational and applied mathematics*, 20:53–65.

Samet, H. (1984). The quadtree and related hierarchical data structures. *ACM Computing Surveys (CSUR)*, 16(2):187–260.

Tybout, A. M., Hauser, J. R., and Koppelman, F. S. (1978). Consumer oriented transportation planning: An integrated methodology for modeling consumer perceptions, preference and behavior. *Advances in Consumer Research Volume 05*.

Von Luxburg, U. (2007). A tutorial on spectral clustering. *Statistics and computing*, 17(4):395–416.

Wang, X., Rostoker, C., and Hamilton, H. J. (2004). Density-based spatial clustering in the presence of obstacles and facilitators. In *European Conference on Principles of Data Mining and Knowledge Discovery*, pages 446–458. Springer.

Zhou, Y., Yao, L., Chen, Y., Gong, Y., and Lai, J. (2017). Bus arrival time calculation model based on smart card data. *Transportation Research Part C: Emerging Technologies*, 74:81–96.