A Continuous Representation of Ad Hoc Ridesharing Potential

Article *in* IEEE Transactions on Intelligent Transportation Systems · October 2016

DOI: 10.1109/TITS.2016.2527052

3 authors, including:

Michael Rigby
University of Melbourne

Stephan Winter
University of Melbourne

A continuous representation of ad hoc ridesharing potential

Michael Rigby[1], *Member, IEEE,* Stephan Winter[1], *Member, IEEE* and Antonio Krüger[2]

***Abstract*—Interacting with ridesharing systems is a complex spatio-temporal task. Traditional approaches rely on the full disclosure of a client's trip information to perform ride matching. However during poor service conditions of low supply or high demand, this requirement may mean that a client can not find any ride matching their intentions. To address this within real-world road networks we extend our map-based, opportunistic client user interface concept, launch pads, from a discrete to a continuous space-time representation of vehicle accessibility to provide a client with a more realistic choice set. To examine this extension under different conditions we conduct two computational experiments. First, we extend our previous investigation into the effects of varying vehicle flexibility and population size on launch pads and a client's probability of pick-up, describing the increased opportunity. Second, observing launch pads within a real-world road network, we analyze aspects of choice and propose necessary architecture improvements. The communication of ride share potential using launch pads provides a client with a simple yet flexible means of interfacing with on-demand transportation.**

***Index Terms*—Intelligent vehicles, geographic information systems, human computer interaction.**

I. Introduction

INTELLIGENT transportation systems research is revealing a future in which humans can better access shared mobility on-demand. Enabled by location-aware, mobile devices connected in real-time, new location-based services using real-time information [1] are allowing individuals to discover opportunities and personalize trips for ad hoc travel [2], [3].

One approach, ad hoc ridesharing [4], [5], seeks to offer a user (*client*) a sustainable mobility service from a pool of human (or autonomous) driver vehicles. Yet despite technology gains, the client user interface (UI) design of existing systems remains too rigid for ad hoc travel. By requiring full disclosure of trip information for *ride matching* to determine joint trips, an application's usability may be hindered when the system is unable to find any matches—particularly during poor service conditions of high demand or low supply.

Identifying a *knowledge gap* in existing client UIs, previous work proposed a new concept called *OppRide* [6]. OppRide replaces the traditional input requirements with a novel 2-step negotiation, in which the client reveals their trip constraints sequentially for improved decision making and privacy. Here the client needs only disclose their intended drop-off constraints to obtain visual feedback called *launch pads* describing their matching pick-up opportunities. The approach can be used by both ridesharing service operators and matching agencies.

Launch pads are quantitative, geographic features representing the system's matching service potential. Communicated in map-based UI, the information facilitates flexible decision making in both time *and* space to create a more agile and thus more usable ridesharing system for ad hoc travel. However early work was conceptualized using a grid-based road network and considered a discrete representation of vehicle accessibility only. Hence when applied to real-world networks of greater heterogeneity, launch pads may become highly limited rendering services inaccessible; particularly in areas of sparse network geometry, which may require that the client travel greater distances than what is necessary.

This paper addresses this issue by extending OppRide to a continuous representation of vehicle accessibility and the creation of a new generalized method for deriving launch pads. From here we hypothesize that launch pads provide an effective representation of ridesharing service potential in real-world networks, defining an effective representation as one communicating the full service potential of rides matching a client's mobility request.

We will seek to prove our hypothesis from a computational perspective of choice using two experiments. First, we revisit previous work using a multi-agent simulation in a grid network and investigate the effects of vehicle flexibility and population size on a client's probability of pick-up using the new continuous launch pads. Here we examine the increased opportunity afforded by extended launch pads and contrast it against both discrete launch pads and traditional UIs. In a second experiment we apply the new approach to a real-world road network and examine launch pads in different contexts using appropriate spatial analyses. Then from these experiments we discuss issues towards real-world testing. The challenges of this work include the adoption of a continuous representation, irregular network geometry and the design of necessary architecture improvements.

The following Section reviews related work, discussing existing ridesharing systems. In Section III we present the main visualization idea describing concepts of time geography and our system architecture. Section IV describes an extended grid-based simulation and discusses results. Building from these, Section V describes a large, real-world simulation of launch pads and presents results. Section VI discusses computation, aspects of choice, and necessary architecture improvements. Section VII concludes the paper detailing future work required.

[1]M. Rigby and S. Winter are with the Department of Infrastructure Engineering, The University of Melbourne, Parkville 3010, Australia (e-mail: rigbym@student.unimelb.edu.au, winter@unimelb.edu.au)

[2]A. Krüger is with DFKI GmbH and Saarland University, Saarbrücken 66123, Germany (e-mail: krueger@dfki.de)

Manuscript received 28 January 2015; revised 31 December 2015.

II. Related Work

TRADITIONAL client UIs for ridesharing are too rigid for ad hoc travel. Existing systems can be divided into two groups: service operators and matching agencies [7]. Designed from the requirements of classic algorithms in operations research, both require full trip information to be identified and disclosed a priori for ride matching (pick-up, drop-off locations and arrival time-window, [8], [9]). This highly constrained approach is suitable in logistics for moving goods between fixed points such as warehouses (e.g., [10]), and is applied in ridesharing with two assumptions: a client's request is fixed in space and is assumed final, i.e., inflexible. Once a potential *ride* (joint trip) is found, it may be described using one of four patterns in respect to the original routes of vehicle and client: *identical*, *inclusive*, *partial* and *detour* [7]. From a use case perspective, the interaction describes a *3-way handshake* consisting of (1) client requests, (2) system matches and offers, and (3) client accepts [11].

Service operators typically use fully automated matching to minimize client control and focus on the accuracy and timeliness of provisioning [12], e.g., dial-a-ride [13]. Whilst being demand-responsive, such systems may not be able to find any matches to a request during poor service conditions. Persisting, the client is reported to resort to using iterative requests; varying their trip constraints in time or space until a satisfactory configuration is found. However without any knowledge of the system's state this blind approach is likely to hinder planning and lead to sub-optimal choices.

For matching agencies such as flinc[1], Carma[2], Zimride[3], Lyft[4] and LiftShare[5], control is handed to the client and less accurate matching (typically heuristics) are used to define a *choice set* comprising potential rides. With this the client must manually contact the driver(s) and negotiate using additional communication channels (e.g., voice or text) to confirm that an opportunity exists. Reliance on ad hoc communication results in three key issues which hinder accuracy, timeliness and privacy. First, there is no guarantee when (or if) a driver will respond and multiple negotiations may be required. Second, differences in spatio-temporal knowledge and other biases may yield sub-optimal choices (for either party). Third, revealing full profile and trip details to drivers before agreeing on a trip may discourage participation [14], [15]. Further, without knowing a driver's service limits, a client can not adequately conceal a sensitive location and or time [16].

The difficulty in addressing both systems' issues lies in the inherent complexity and variability of ridesharing service potential. Here a client's lack of knowledge (and its related affordance of flexibility) precludes ad hoc travel. To overcome this gap, some applications such as Sidecar[6] have included maps in their UIs, representing vehicle locations as points and routes as lines. Viewing these however, a client has no possibility to understand what is relevant to them, i.e., to what degree vehicles are willing to detour through space and how this relates to time (particularly when multiple clients are introduced). Consequently the map is not useful for decision making and manual negotiations are still required.

[1] flinc, http://www.flinc.org
[2] Carma, http://www.carmacarpool.com
[3] Zimride, http://www.zimride.com
[4] Lyft, http://www.lyft.me
[5] LiftShare, http://www.liftshare.com
[6] Sidecar, http://www.side.cr

Addressing this, previous work has designed OppRide which provides a lens into matching [6]. Initially requiring only drop-off constraints (location and arrival time-window), OppRide computes and offers discrete, map-based features called launch pads for the client to choose a pick-up. The interaction design forms a 2-step negotiation which augments the traditional 3-way handshake for situation awareness. Understanding potential service heterogeneity across space-time, subsequent simulation work showed the value of enhancing discrete launch pads with additional dimensions using color to improve a client's decision making in ridesharing [17].

Building from time geography concepts, OppRide uses the *network time prism* (NTP) constructed from potential path trees [18] to represent a vehicle's future service potential in three-dimensions (x,y,t). Yet by constructing this from discrete network features (road intersections and end-points), only discrete launch pads can be derived. Consequently in real-world networks of sparse geometry, launch pads may become highly limited (or even null) hindering client choice and accessibility. To include latent service potential *along edges*, this perspective must be extended to a continuous model.

Such a model has been developed to represent the spatio-temporal uncertainty of moving objects in road networks [19]. Understanding that an object's location observed by a GPS device may contain measurement error, the authors trade errors in space for time to estimate potential locations between two consecutive observations. Using Dijkstra's algorithm between the two points, potential path trees are determined bounded by the shortest path in time including the error estimate. From the vertices constituting these paths, three-dimensional polygonal features can be created and aggregated to form a continuous NTP. This representation shows significant opportunity to extend OppRide if positional uncertainty is replaced by known vehicle flexibility in time. However after adapting such a model, a new method would be required to derive launch pads.

Spatio-temporal operations on objects in three-dimensions (x,y,t) have been used extensively for exploratory analysis of trajectory data [20]. Using a time slice t_a (or slab for [t_a,t_b]), the potential locations of an object may be determined [21]. Whilst we used these in our earlier work to extract vertices from a discrete NTP, a continuous model would require different operations given its polygonal features. Such operations have been studied in detail in computer graphics (e.g. [22]). With standardized tools and libraries, opportunity exists to perform such analyses on spatial objects within databases like PostgreSQL using extensions such as PostGIS.

With continuous launch pads, the choice set would be extended and representational issues must be considered (cf. [23]). Whilst the value of the increased opportunity would depend on a client's intentions and choice factors [24], greater individual utility [25] may be gained with existing spatial knowledge of the environment to facilitate personalization, e.g., en-route purchase of a coffee [26] or obfuscation of a location and or time for privacy and security [16].

To achieve such level of choice; similar to that of a taxi service, there is need to extend OppRide. Addressing the issues identified and particularly those of existing systems in regards to choice and privacy, this paper will now discuss the transfer of OppRide from a discrete to a continuous representation of vehicle accessibility.

III. Extending an opportunistic client user interface

To extend an opportunistic client UI for ad hoc ridesharing, we revisit time geography concepts. Time geography provides a framework to represent the locational uncertainty of an object sometime after or before it was first encountered (a *space-time cone*), or between two encounters (a *space-time prism*) [27]. Applied to network space, the concept of a *network time prism* (NTP) can be used to describe a vehicle's accessibility given its constraints, e.g., static (speed limit) or dynamic (traffic flow) [18].

The OppRide architecture exploits the NTP representations of all vehicles within its service pool to create a dynamic representation of the system's pick-up opportunities which match a client's drop-off request. It is the centralized authority's responsibility to maintain each vehicle's NTP, which we interpret as their individual ridesharing service potential (Fig. 1a). To create and maintain this representation the authority must know the vehicle's current location O, destination D and their associated times. From here they can determine the vehicle's time to traverse their shortest path OD, $t_{shortest}$, and depending on the vehicle's flexibility in time, t_{flex}, opportunity may exist for ridesharing.

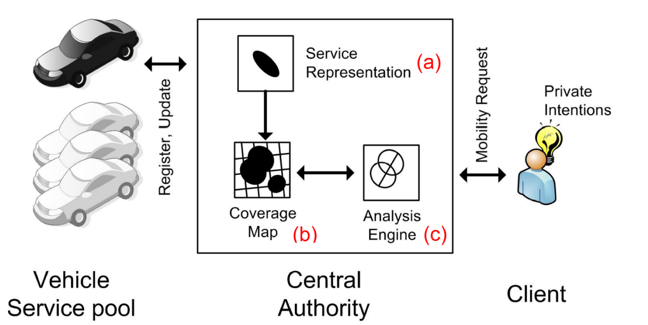

Fig. 1. OppRide architecture to provide a client with visual feedback called launchpads, showing (a) service representation, (b) coverage map, and (c) analysis engine components of the central authority.

A. A continuous representation of vehicle accessibility

In earlier work we constructed a vehicle's NTP from discrete road network features only [6]. Whilst this creates a representation analogous to public transportation stops, the choice set may become highly limited. For this reason we adapt a model of spatio-temporal uncertainty of moving objects [19] to our K shortest path approach [6].

Assume each vehicle's flexibility, t_{flex}, to be a dynamic resource defined by the ordered set of client constraints on (or soon to) board, the human driver (if there is one) and or the authority, e.g., service extents. By incorporating flexibility into the NTP a client's pick-up and drop-off choices can be made within the vehicle's intentions. Understanding this complexity, t_{flex} is managed by the authority using a *flexibility bank*.

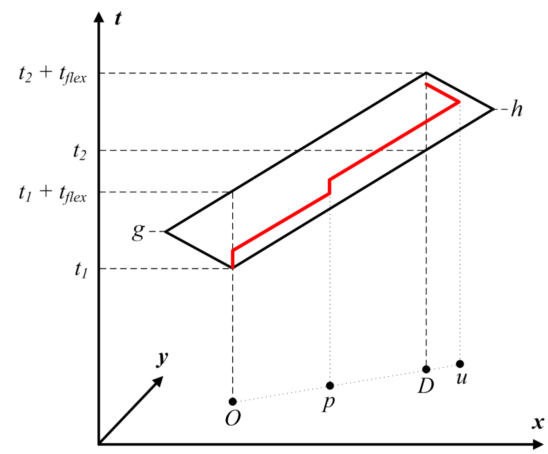

Fig. 2. Representation of a vehicle's network time prism (NTP) describing movement at a constant velocity along a single bidirectional edge from current location O to destination D with flexibility t_{flex}. The prism contains the path OpD (red) of a vehicle detouring from its shortest path OD within the constraints of t_{flex} to pick-up a client at p. Points g and h delineate the prism's extents if u-turns are allowed.

A continuous NTP allows time to be used along edges in various ways, e.g., waiting for a client (a space-time station in time geography terms [28]), or trading for space to detour from the shortest path and pick-up/drop-off a client. In Fig. 2, a vehicle can take the shortest path from O to D and arrive early (lower bound), stay at O for the duration of t_{flex} then travel the shortest path to D (upper bound), or use t_{flex} for ridesharing to detour through p, perform a u-turn at u and reach the desired side of the road at D. Ultimately the bank's balance defines the time available for a joint trip where $t_{budget} = t_{shortest} + t_{flex}$. The metaphor allows future scope to consider negotiated fares where time is traded between clients. This concept becomes pertinent with autonomous vehicles where the time available depends on the service queue.

Given a weighted directed graph G comprising edges E and vertices V representing the road network, where $G = (V,E)$ and $E \subseteq V \times V$, the NTP can be delineated using the function $f : O \times W \times D \rightarrow K$, where $W \subset 2^V$ is a set of waypoints and K a set of paths OD where $k_i = (v_1,v_2,..., v_n)$, $v_j \in V$, $(v_j,v_{j+1}) \in E$, $j = 1...n$ where $v_1 = O$ and $v_n = D$.

To compute f within real-world networks, adapt the model of movement uncertainty to include dynamic segmentation, network isochrone and attribution (Algorithm 1). Start by delineating W between OD using an ellipse defined by t_{budget} [29]. Using a K bidirectional Dijkstra search constrained by t_{budget}, calculate k_i's t_{flex} and attribute each w their earliest arrival t^- and latest departure times t^+. Next, compute isochrones at each w using t_{flex} (paths from u-turns) and update waypoints. Then determine vectors between consecutive vertices of k_i using their temporal bounds, $t^w = [t^-,t^+]$ and create polygonal features. Finally, merge all features.

With potential variation in real-world edge weights due to, e.g., congestion (particularly in urban areas), Dijkstra's

Algorithm 1 Compute network time prism (COMPNTP)

Restrictions: $G = (V, E)$ road network; function $f{:}O{\times}W{\times}D{\rightarrow}K$; $coverageMap$ as spatialDB (x,y,t)
Local data: K, W, $geom$, ntp initialize to zero

updateWeights(G,E)
updateTopology(G,O,D) *#dynamic segmentation* [30]
$t_{shortest}$:= pathTime(shortestPath(G,O,D))
$t_{budget} := t_{flex} + t_{shortest}$
W := defineWaypoints(G,O,D,t_{budget}) *#spatial reach* [29]
$G^{'}$:= defineSubgraph(G,W)
for each $w \in W$ **do**
  k_i := path(G',O,w,D) *#bidirectional Dijkstra*
  $\phi := t_{budget}$ - pathTime(k_i) *#ridesharing potential*
  if $\phi > 0$ **then**
    $K.add(k_i)$
    $w.t^-$:= earliestArrival(G',O,w,t_O,ϕ)
    $w.t^+$:= latestDeparture(G',D,w,t_D,ϕ)
    $W.add$(isochrone(G',w,ϕ)) *#u-turn waypoints* [31]
    updateTopology(G',W)
for each $k_i \in K$ **do**
  for each $v_j \in k_i$ **do**
    bds := newBounds($W.get(v_j.id).t^w$,$W.get(v_{j+1}.id).t^w$)
    $geom := geom$ + newGeometry(bds)
ntp := mergeGeometry($geom$) *#optimize object*
$coverageMap.update(ntp)$

algorithm may become too simplistic. Hence probabilistic routes [32], may be considered for more accurate paths (e.g., [33]). Alternative data structures may also be used (cf. [34]).

Aggregating all vehicle NTPs, the authority maintains a global service *coverage map* as a spatio-temporal database (Fig. 1b). This map is used by the authority to derive launch pads in response to a client's request.

B. Launch pads

Launch pads form the basis of our opportunistic client UI, communicating the system's global service offering during a client's 2-step negotiation. Conceptually they abstract the pick-up opportunities matching a client's drop-off request $\{p\}$, to virtual space in the form of interactive map-based features. Extending to a continuous model the authority requires a new method to derive launch pads and subsequently a new interaction design is required for the client.

Step 1. Client request: Client defines their drop-off location, d, arrival time t_d[7] with flexibility in time, t_{flex}, and submits the request to the authority.

Authority responds: Analysis engine processes request (Fig. 1c), queries the coverage map and derives launch pads. Here spatial-temporal (ST) and temporal (T) queries are used to filter vehicles, then slice matching NTPs according to the client pick-up time-window set in the UI (Algorithm 2, Fig. 3). Slabs are then projected to create maps then form launch pads.

Step 2. Client decision making: Launch pads are shown to the client overlaid on an interactive map describing the transportation network. From here only some may be relevant to their intentions: those that are nearby or more precisely those intersecting the client's own private space-time volume, e.g., maximum walking or driving time. For example, for a

[7]Uppercase D for vehicle, lowercase d for client

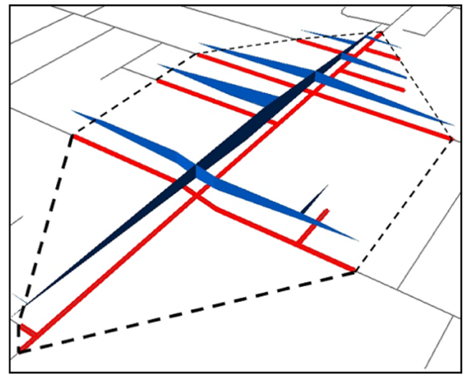

Fig. 3. Derivation of a map from a single vehicle's NTP in the Melbourne suburb of Hawthorn, showing 3D (x,y,t) slab (blue) and 2D (x,y) projected map represented using line features describing road portions (red). Dashed line describes the geometry's bounding convex hull.

client to be picked up somewhere in 15 minutes, they must be able to reach this location in 15 minutes as well. This choice—where and when to meet—is the client's own who can choose any location p within the launch pads (Fig. 4). A slider in the UI may be used to search through space-time, defining a pick-up time-window $[t_a,t_b]$. Once chosen the pick-up is initially snapped to the closest point in the road network and may be fine-tuned by dragging it within the launch pad's boundary. Whilst the UI may suggest approximate metrics to access p (which may be personalized), we make no assumptions regarding the client's accessibility for privacy reasons.

Launch pads derived from a continuous NTP are continuous (line) features. To assist reasoning at smaller map scales, these may be bounded (polygon), e.g., convex hull [35]. If a system designer prefers points however, lines may be discretized using some strategy such as minimum vehicle stopping distance.

Reasoning using continuous launch pads is initially binary: *within*$|$*outside*, differing considerably from their discrete variant. The continuity of the representation allows existing public transportation stops with accessible infrastructure to be highlighted for reasons of inclusive design. To further consider a client's individual route choice behavior, e.g., preference or historical data, a client may choose to visualize launch pads in various ways, e.g., color ramps [17].

Accepting p, the authority determines the times when vehicles can service the requested location and the client is

Algorithm 2 Derive launch pads (DERLP)

Restrictions: $coverageMap$ as spatialDB (x,y,t), analysis engine function $f{:}derlp(request, ui.pickupWindow)$
Local data: $maps$, $launchPads$, i initialize to zero

$dRequest := request.dropOff$ *#(d,t_d+t_{flex})*
$m := coverageMap.stQuery(dRequest)$ *#ST query*
$pRequest := ui.pickupWindow$ *#[t_a,t_b]*
for each $vehicle \in m$ **do**
  $ntp_i := vehicle.ntp.update(dRequest)$
  $slab_i := coverageMap.tQuery(ntp{,}pRequest)$ *#T query*
  map_i := project($slab_i$) #$\mathbb{R}^3 \mapsto \mathbb{R}^2$
  $maps[i] := map_i$
$vRequest := request.preferences.visualization$
$launchPads$:= aggregate($maps$,$vRequest$)

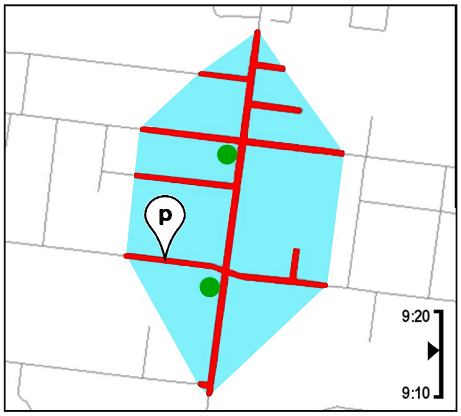

Fig. 4. Conceptual client UI showing example launch pad (formed from the single map in Fig. 3), with existing tram stops (green), movable pick-up icon p and vertical time slider to define the pick-up time, t_p. The convex hull (blue) abstracts road portions (red) at lower zoom levels.

presented with time-window(s) describing available pick-ups. Using a suitable interface such as a slider, the client then chooses a discrete pick-up time, t_p (Fig. 4). Note, with multiple vehicles with varying constraints both launch pads and pick-up time-windows may become discontiguous. Following this choice act, the authority issues the client with a *mobility contract*. Once accepted, the authority updates the matched vehicle with new route instructions to collect the client at the agreed location and the vehicle's description and current location is revealed to the client for local decision making, e.g., moving to or waiting at the agreed location. If the client is not at the pick-up by the agreed time, they forfeit the contract and the vehicle moves onto its next (or final) destination.

Compared to previous work, a continuous representation affords significantly greater choice in space than its discrete variant. This increased choice further facilitates ad hoc ridesharing and provides a client with greater opportunity to personalize their trip. Whilst the interaction design considers a client's directed travel by choosing their drop-off first, further extensions may be considered such as an explorer for undirected travel [36] (selecting a pick-up first) and *landing pads* near a chosen drop-off during, e.g., high demand or low supply.

To characterize the benefit of this paper's extension, we revisit previous work and examine the opportunity afforded by a continuous representation.

IV. Client Opportunity

We now investigate the effects of varying vehicle population size and flexibility on a client's probability of pick-up using continuous launch pads.

A. Experiment Design

The launch pads a client receives during their negotiation are dependent upon the intersection of their drop-off constraints with the system's service potential. As described in Section III(A), each vehicle's service potential is dependent on its constraints, hence the spatial characteristics of launch pads are inherently variable.

To examine the effect of this variation on a client's use of OppRide at a conceptual level, we revisit and extend our previous experiment in a grid network [6]. We again choose a regular environment as it allows vehicles to travel multiple potential paths, letting us examine how changes in vehicle population size and flexibility in time effect launch pad size, i.e., a client's pick-up choice set. This setup allows us to contrast this paper's extension against previous work using discrete launch pads and otherwise traditional UIs.

Define the transportation network as a regular grid of 15 $\times$ 15 vertices connected by edges weighted according to their Euclidean distance. For simplicity set each edge distance to 1000 units and balance the distribution of agents by dividing the network into 9 equal $zones$, each containing 5 $\times$ 5 vertices and number each zone 1-9. Define the central zone 5 as containing the *service area* within which the client agent will seek to move. This design allows us to observe client-vehicle interactions without boundary effects from network geometry.

For the creation and random assignment of vehicle and client agents within the network we again use the multi-agent simulation toolkit Repast[8]. To cater for the continuous extension, we outsource the central authority's tasks to a PostgreSQL database using PostGIS[9] and pgRouting[10] extensions. The database approach allows for efficient vehicle routing, creation of the 3D polygonal features comprising each vehicle's NTP and their subsequent storage, update, aggregation and analysis. Convex hull features are computed as the minimum bounding geometry using scripts.

Populating the network with n vehicle agents, we assign each random origin O and destination D vertices, setting their flexibility in time t_{flex} based on $t_{shortest}$. We set a minimum path length OD of 14 edges to ensure agents have some potential to cross the service area. At instantiation each vehicle agent registers with the authority agent which manages their NTP and maintains the coverage map.

We again consider a client flexible in time but fixed in space, allowing us to contrast continuous launch pads against previous work [6] and traditional approaches. The client's 2-step negotiation is automated in the following way. At instantiation, the client agent is randomly assigned pick-up p and drop-off d vertices on the boundary of the service area using a uniform distribution. We constrain this choice to ensure a shortest path pd of 6 edges (the average possible within the service area). The client then creates a request describing their drop-off d, arrival time t_d = t_{now}+$t_{shortest}$ and flexibility t_{flex} only. Following [8] we interpret client flexibility using a stretch factor which we set to one third (which we deem to be a fair compromise) and form their arrival time-window [9], [t_d, t_d+t_{flex}].

The request is received by the authority agent and the drop-off constraints are checked for intersection using the coverage map. Recalculating (if necessary) and aggregating all matching vehicles' NTPs OdD, the authority agent derives launch pads and issues them to the client to choose a pick-up. Viewing the launch pads, the client agent's determines if

[8]Repast, http://repast.sourceforge.net/
[9]PostGIS, http://www.postgis.net
[10]pgRouting, http://www.pgrouting.net

their pick-up location (their current, private location) intersects the launch pads offered—thus short-circuiting the decision strategy described in Section III(B). We deem their negotiation to be a *success* if the pick-up intersects any launch pad. Results are then used to calculate a client's probability of pick-up in different scenarios and launch pad sizes are recorded. Following exhaustion of t_{flex} without success, the client agent is removed and a new one is added to maintain a constant population. This reiteration process is the same for vehicle agents upon reaching their destination.

B. Results

For the populations of 1, 5, 10, 20 and 40 vehicles, we observe the effects of varying vehicle flexibility on a client's probability of pick-up. For this purpose we consider vehicle flexibility in the range 0-1.0 as a percentage of their shortest path length OD. Observing convergence, we delimit results to 300 requests and scale them accordingly.

Results in Fig. 5 show for a client's 50% probability of pick-up, 20 vehicles would require a flexibility of approximately 0.25. Further, for a 90% probability of pick-up, 40 vehicles would need a flexibility of approximately 0.33. As a continuous representation contains the discrete launch pads used in previous work, these observations are only slightly more conservative due to greater accuracy in both the representation and time model used. We observe that a vehicle flexibility of one third provides a good cross-section of a client's probability of pick-up for all vehicle population sizes.

Examining the average launch pad convex hull areas derived for a single vehicle during successful contract negotiations, we find that increasing vehicle flexibility leads to larger launch pads, increasing the opportunity available to a client. To illustrate this we consider the spatial characteristics of launch pads offered by a single vehicle traveling OD, responding to a client's mobility request for drop-off at d. Varying the vehicle's flexibility: 0, 0.25, 0.5, we record the launch pads derived (Fig. 6). Here we observe that in a regular grid network launch pads are symmetrical arising from uniform degrees of centrality. Also launch pads are elliptical in shape, abstracting movement behavior in free or unconstrained space [29], [19]. From a UI perspective, larger launch pads afford greater choice

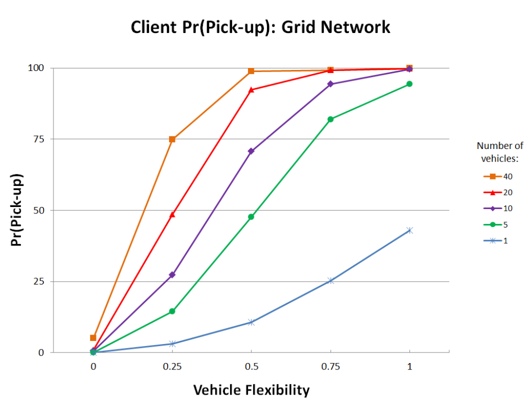

Fig. 5. Client probability of pick-up: Effect of varying vehicle flexibility and population size.

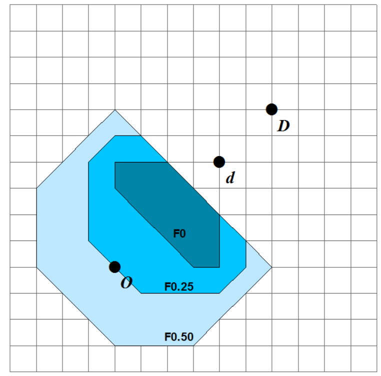

Fig. 6. Effects of varying vehicle flexibility on launch pad size derived for a single vehicle departing O at t = 0, arriving D at t = 12, for a client's mobility request for drop-off at d and seeking pick-up at some location within the time window [3, 6]. Vehicle flexibility (F): 0, 0.25 and 0.50.

to a client—choice to personalize their pick-up location or time within the available continuum. Note, as smaller launch pads decrease the size of a client's choice set, they may also reveal potentially sensitive information about the driver (or clients).

Drawing Figures 5 and 6 together, a vehicle flexible in space-time with larger launch pads, such as a taxi, may provide a client with a higher probability of pick-up compared to the inflexibility of a bus following fixed routes and schedules. Here the request's success relies on the coincidence of the client's constraints with the vehicle's route and schedule which would require sufficient spatio-temporal information to coordinate.

Whilst this perspective can be used by a system designer to balance supply-demand from a global perspective, a client does not need to know about this—what matters to them is simply knowing when and where to get a pick-up. When visually communicated in the map-based UI, launch pads provide knowledge of the system's current service potential matching. Taking a use case approach, if a client sees that their current location is *within* a launch pad, they may choose a pick-up immediately there or use this information in an opportunistic way, modifying their intentions, e.g., to delay scheduled events or obfuscate a sensitive location and or time. However as results in Fig. 5 illustrate in all scenarios, a client's probability of pick-up is asymptotic, meaning they may never be guaranteed any ride.

Contrasting a client's successful negotiation against a traditional mobility request using single vehicle routing [8], the same result is achieved as in retrospect the launch pads derived will always contain the requested pick-up. For an unsuccessful request however, a traditional approach would provide a client with no feedback. Considering this an unresolved use case, negotiation offers a solution by revealing the system's matching potential. Here a client would see their current location is *outside* the launch pads offered and can reason within their private constraints, e.g., physical needs, by, i.e., moving to a more serviceable area or using an alternate mobility service, thus avoiding use of blind, iterative requests.

Whilst results in Fig. 5 define the lower bound of a client's probability of pick-up; mirroring a traditional approach which is similarly fixed in space, as discussed in previous work, launch pads provide a client with opportunity to increase this. Further, by extending from discrete to continuous launch pads, new results in this paper illustrate how the increased choice set affords even greater (a) potential accessibility to, and (b) personalization of, a pick-up location and time. From this computational perspective it is explicitly here that we find continuous launch pads to be more effective at communicating service potential of vehicles in ridesharing compared to discrete launch pads and traditional methods.

Applying these observations we now investigate the potential choice and privacy characteristics of continuous launch pads within a real-world road network of greater heterogeneity.

V. Launch pads in context

REAL-WORLD road networks are characterized by heterogeneous road geometry and degrees of centrality. Building from the previous experiment's observations, the goal of this second experiment is to examine launch pads in various contexts. Following the experiment issues of computation and choice will be discussed.

A. Experiment Design

To examine the spatio-temporal characteristics of launch pads, a large study area of approximately 15200km^2 is selected which includes the geographic area of Greater Melbourne. Located around a large water body, this area is characterized by a diverse range of land uses influencing both network geometry and centrality. Given potential influences of land use on the road network [37], the following spatial characteristics of both launch pads and their broader geographic context will be observed:

- Launch pad: Total road length (meters)
- Launch pad: Convex hull area (meters squared)
- Context: Total road length (meters)
- Context: Area (meters squared)
- Context: Land use type

PSMA Australia's Transport product[11] is chosen for our network, describing roads as edge geometry. Creating a network topology for routing, vertices are generated at road intersections and end-points. Focusing on geometric factors, edges are weighted according to their Euclidean distance only.

This experiment is conducted using PostgreSQL scripts alone. A client's mobility request is simulated by randomly selecting a drop-off location d and setting a standardized arrival time-window with flexibility [t_d, t_d+t_{flex}]. To simulate a vehicle satisfying this request, a vehicle is created local to d traveling between some random location O and destination D with some flexibility in time. This is then registered with the authority which calculates the corresponding NTP and automatically derives a launch pad in response to the client's request. Note, in this design the vehicle may (or may not) be required to detour from its shortest path OD to reach d (inclusive, partial and detour ridesharing patterns only, the identical pattern is not considered in order to focus on the potential influences of network geometry).

[11]PSMA Australia, http://www.psma.com.au

The geometry of launch pads and their context is quantitatively analyzed using PostGIS functions[12]. The launch pad road density ratio LR is calculated as meters of road per square meter of convex hull area:

$$LR = \frac{\Sigma(\text{road length})}{\text{convex hull area}}$$

To consider the feature's context the size of this area is then broaden by buffering the shortest path's midpoint:

$$CR = \frac{\Sigma(\text{road length})}{\text{context area}}$$

Comparing both densities of the same index, $\frac{CR}{LR}$, a third ratio defines a naïve accessibility score, AS. To compare the continuous extension against previous work, the choice set will be discretized to points using various strategies and the opportunities afforded using these ratios.

Two preliminary investigations were used to tune the experiment's parameters. The first examined the distances between bus and tram stops along routes in the study area[13]. Results showed an average stop distance of approximately 300 meters. Discretization of the continuous representation at this interval (D300), at road intersections and end points (EP) and a denser 100 meter distribution (D100), allows the different representations to be compared. These strategies use the following rule incorporating local regulations: create point feature 20 meters from a road intersection then at the defined interval.

Second, we examined the average path lengths of motor vehicle and passenger trips across our study area using VISTA[14] data. Gauging trip context, land uses were examined in our study area using planning data[15]. Results delineate five major land use types in the study area: urban and new residential, industrial, coastal and rural. They show that trips within rural areas average 10km. compared to trips within urban (including suburban and industrial) areas averaging 6km. (coastal areas were divided based on their relative metropolitan location and composition). Trips between both areas average 27km. Further analysis of network geometry in these areas revealed road densities of 0.31% in rural compared to 1.08% in urban areas. These observations will now be applied to the experiment's design and its results will be used to characterize launch pads in various contexts and present a formal discussion.

B. Results

A random sample of 75 launch pads was observed in response to simulated mobility requests (Fig. 4 describes one sample from this set). Responding to each request, a single vehicle was created local to d and set plans to travel OD with the following constraints determined from the above tuning:

- O: Random vertex within 1000 meters of d
- D: Unique random vertex within 1000 meters of d
- $|OD|$: Minimum 1000 meters, maximum 2000 meters

[12]PostGIS, http://www.postgis.net/

[13]Public Transport Victoria (PTV), http://www.ptv.vic.gov.au/

[14]Victorian Integrated Survey of Travel and Activity (VISTA) 2009/10, http://www.transport.vic.gov.au/research/statistics/

[15]Victorian Resources Online (VRO), Department of Economic Development, Jobs, Transport & Resources, http://vro.depi.vic.gov.au/

- Flexibility: 0.33
- Context radius: 1000 meters

We deem the variability of OD to be suitable given that vehicles in reality inherently travel different distances.

Following calculation of the vehicle's NTP and satisfaction of the request's d constraints, launch pads were derived using a standardized pick-up time-window: $[t_a, t_b]$, where $a = \frac{1}{3}x$, $b = \frac{2x}{3}$ and $x = \frac{|OD|}{v}$, where vehicle velocity v is assumed limited at the speed limit of 50km/h. Based on the tuning, a context buffer radius of 1000 meters was chosen, approximating the vehicle's maximum path length. Reviewing the 75 random samples (Fig. 7), we observe good diversity of both road network geometry and context.

TABLE I
AVERAGE RATIOS OF ROAD DENSITIES IN LAUNCH PADS (LR) AND CONTEXT (CR) WITH NAÏVE ACCESSIBILITY SCORE (AS) FOR 75 SAMPLES GROUPED BY LAND USE TYPE.

	LR	CR	AS
Urban res.	**1.82%**	1.16%	0.641
New res.	1.44%	0.92%	0.639
Coastal	1.56%	0.66%	0.425
Rural	1.54%	0.51%	0.329
Industrial	1.63%	0.81%	0.494
Other	1.44%	1.22%	0.851

Observations of road densities within launch pad LR and context CR presented in Table I clearly illustrate the effects of land use as indicated by their accessibility score, AS. Whilst greater LR may be observed in one environment, vehicles are dynamic and consequently their service potential represented within launch pads would be variable, particularly as they may traverse multiple contexts en-route OD.

From this observation a general discussion of the results is now presented, acknowledging the complexities involved with assumptions presented for discursive purposes only. Launch pads observed in urban residential areas have both high LR and CR, indicating greater regularity. Such areas can generally be characterized by smaller lots such as housing, which may require a higher density of public roads for vehicle accessibility. By comparison, coastal areas with a high LR have a considerably lower CR, indicating greater irregularity due to recreational land uses or water bodies for example. In contrast, industrial areas are characterized by similar ratios for different reasons: larger, yet more regular lot sizes for warehousing for example. Similarly rural areas with the lowest ratios may be characterized by larger land lots for, e.g., agriculture, requiring less publicly accessible roads. Note, the consistently high LR observed in comparison with CR for all land use types is due to the vehicle's OD constraints chosen. In practice these results would vary with heterogeneous vehicle flexibility and client requests. As previously observed in Figure 6, these factors would influence the size of launch pads offered.

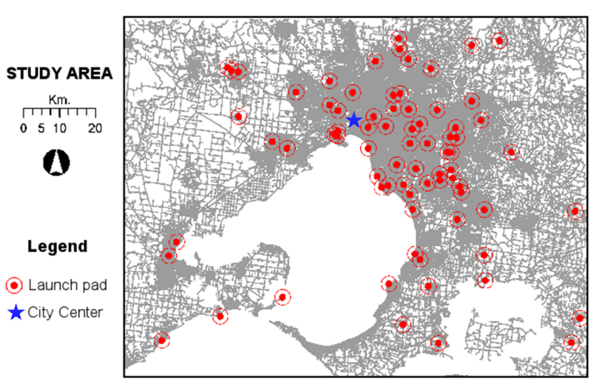

Fig. 7. Locations of 75 launch pad samples (red) within the Greater Melbourne study area (gray).

TABLE II
CONTINUOUS LAUNCH PADS (LR) COMPARED TO THEIR DISCRETE VARIANTS DETERMINED USING END POINT (EP), 300 METER (D300) AND 100 METER (D100) STRATEGIES, SHOWING AVERAGE FREQUENCY OF POINTS AND DENSITY PER CONVEX HULL AREA (DENS).

		Discretization Method					
	LR	EP	*DENS*	D300	*DENS*	D100	*DENS*
Urban res.	**1.82%**	42.0	***0.064***	4.4	***0.005***	14.4	***0.016***
New res.	1.44%	43.8	*0.047*	5.1	*0.004*	18.5	*0.014*
Coastal	1.56%	65.9	*0.042*	7.6	*0.005*	23.4	*0.014*
Rural	1.54%	31.2	*0.030*	3.6	*0.004*	13.4	*0.015*
Industrial	1.63%	44.9	*0.042*	5.4	*0.005*	15.8	*0.015*
Other	1.44%	33.0	*0.044*	4.1	*0.005*	12.6	*0.014*

Comparing the characteristics of continuous launch pads against their discrete variants (Table II), the influence of network geometry on a client's choice set is again evident. Earlier results of road density in Table I shows how LR varies with context and this is mirrored in the results of discretization.

Using an end point method (EP), the average frequency of discrete options is highest within urban residential areas, resulting on average in a larger choice set. Compared to coastal, industrial and rural areas this ratio varies considerably due to road density (cf. Table I), which together with further analysis of average road lengths reveals the potential unsuitability of this strategy. Conversely, whilst discretization using D300 or D100 methods creates even representations, both strategies are naïve as all edges are equally weighted, signaling a need for improvement using probabilistic edge weights.

Whilst our earlier experiment showed launch pads to be effective in a grid network from a computational perspective, this second experiment characterizes their inherent variability across space—variability influencing both the assumptions used to determine the NTP and the related issues of computation time and client choice. Each of these perspectives will now be discussed.

VI. TOWARDS REAL-WORLD TESTING

A. Probabilistic edge weights

Whilst simply weighting edges according to their Euclidean distance and a constant velocity assumption serves to demonstrate OppRide at a conceptual level, the potential impacts of real-world factors such as congestion would likely hinder service quality. Seeking edge weights of greater accuracy, new sensing technologies both within road infrastructure and vehicles are revealing representations of the road network at finer spatio-temporal granularities.

Such big data is facilitating the determination of probabilistic edge weights using complex cost functions. Recent work in the direction of probabilistic time geography [38] is including such attributes within the NTP (e.g., [39]). Yet once a probabilistic NTP is constructed and launch pads derived, this raises the question: what level of uncertainty, in terms of multiples of standard deviation, can guarantee some level of service satisfaction for clients?

B. Architecture and choice considerations

Whilst unhinging traditional matching to provide a client with greater choice is OppRide's goal, extension to a continuous representation also presents challenges for system design due to the related issues of computation and choice. Results in Table II illustrate how the size of a client's matching choice set would be greatest in urban areas due to denser network geometry. Yet with potentially a greater number of potential paths, the time required to compute a NTP may increase significantly. Further considering real-world supply, subsequent analyses of VISTA data reveal that simulated requests on trunk roads near the city center in peak hour (Fig. 7), may be matched to potentially hundreds of vehicles and the total computation time may preclude ad hoc travel.

To constrain matching some estimate of the client's pick-up location—which does not hinder OppRide's opportunistic and privacy preserving approach—is required. Consequently a pick-up *region* may be introduced to Algorithm 2 for spatial relevance. Here m vehicles matching the client's request would be reduced to only those with NTPs intersecting the region. However definition of this requires care for privacy and accessibility reasons, i.e., large enough to conceal a sensitive location such as a current location, whilst capturing sufficient opportunity in dynamic conditions. To further examine the impacts of region size on a client's probability of pick-up, OppRide must be stress tested in real-world scenarios.

Following this definition, the set of m vehicles may still require filtering. Here a distribution, e.g., Gaussian, and or client preferences may be introduced. With heterogeneous service potential, additional vehicle attributes such as fare or seating may be critical to decision making and various studies have already identified potential factors (e.g., [40], [41]). However with potentially increased cognitive load, a client's decision time must be investigated (cf. [42]). If launch pads were deemed to be exclusive, longer decisions may deny other clients opportunity (particularly during peak hour commutes).

Whilst such work would extend research into new fields such as human-computer interaction, an integrated approach is necessary. Any improvement would need to consider potential impacts on the three levels of mobility management: political, managerial and user [43].

VII. Conclusion and Future Work

This paper has extended an opportunistic client UI for ad hoc ridesharing by adopting a continuous representation of vehicle accessibility. It describes a generalized method to derive continuous launch pads offering a client greater opportunity compared to the previously studied discrete variant. The novel contributions include: development of a generalized method for deriving continuous (and discrete) launch pads, investigation of continuous launch pads in a grid network, characterization of launch pads in a real-world network, design and testing of discretization strategies, and a discussion of necessary requirements for real-world testing.

From the results of two experiments we deem the new approach to be effective in regards to communicating the full service potential of vehicles in OppRide at a conceptual level only. Towards fully answering the hypothesis, further work needs to simulate launch pads in road networks incorporating time varying flows data using probabilistic edge weights.

Drawing results from both experiments together, choice of which representation to use must consider human factors, e.g., spatial cognition. While the new approach can be validated at a computational level, gauging human understanding and use of the metaphor remains. Real-world case studies using human participants must be performed in-situ under dynamic conditions to observe choice behavior and establish requirements (functional, data and environmental [44]). These results would assist the design of a prototype client UI and simulations.

The application of launch pads in demand-responsive transportation with multiple clients may also be studied. Simulations are needed to examine new agents behaviors and stress test configurations of authority and vehicle constraints to determine an appropriate service mix. Results would also allow the design of dynamic discretization strategies varying with demand. Here the feasibility of ad hoc client refinement and renegotiation features may be examined. Such additions would be pertinent in future systems using autonomous vehicles.

Acknowledgments

Thanks to Walied Othman, Maria Luisa Damiani, Matt Duckham and Nicole Ronald. Ian Tangert provided PostGIS assistance.

References

[1] J. Raper, G. Gartner, H. Karimi, and C. Rizos, "Applications of location-based services: a selected review," *Journal of Location Based Services*, vol. 1, no. 2, pp. 89–111, 2007.

[2] D. Steger-Vonmetz, "Improving modal choice and transport efficiency with the virtual ridesharing agency," in *Proceedings of Intelligent Transportation Systems 2005*. IEEE, 2005, pp. 994–999.

[3] A. Vishwanath, H. S. Gan, S. Kalyanaraman, S. Winter, and I. Mareels, "Personalised public transportation: A new mobility model for urban and suburban transportation," in *Intelligent Transportation Systems (ITSC), 2014 IEEE 17th International Conference on*, Oct 2014, pp. 1831–1836.

[4] S. Winter and S. Nittel, "Ad-hoc shared-ride trip planning by mobile geosensor networks," *International Journal of Geographical Information Science*, vol. 20, no. 8, pp. 899–916, 2006.

[5] M. Raubal, S. Winter, S. Teßmann, and C. Gaisbauer, "Time geography for ad-hoc shared-ride trip planning in mobile geosensor networks," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 62, no. 5, pp. 366–381, 2007.

[6] M. Rigby, A. Krüger, and S. Winter, "An opportunistic client user interface to support centralized ride share planning," in *Proceedings of the 21st ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*. ACM, 2013.

[7] M. Furuhata, M. Dessouky, F. Ordóñez, M. E. Brunet, X. Wang, and S. Koenig, "Ridesharing: The state-of-the-art and future directions," *Transportation Research Part B: Methodological*, vol. 57, pp. 28–46, 2013.

[8] M. Caramia, G. Italiano, G. Oriolo, A. Pacifici, and A. Perugia, "Routing a fleet of vehicles for dynamic combined pick-up and deliveries services," in *Operations Research Proceedings 2001*, P. Chamoni, R. Leisten, A. Martin, J. Minnemann, and H. Stadtler, Eds. Heidelberg: Springer-Verlag, 2001.

[9] A. Fabri and P. Recht, "On dynamic pickup and delivery vehicle routing with several time windows and waiting times," *Transportation Research Part B: Methodological*, vol. 40, no. 4, pp. 335–350, 2006.

[10] M. Savelsbergh, "The vehicle routing problem with time windows: Minimizing route duration," *ORSA Journal on Computing*, vol. 4, no. 2, pp. 146–154, 1992.

[11] M. Dogru, "Car pooling with GIS map server and web services," Master's thesis, University of Zurich, 2004.

[12] N. Agatz, A. Erera, M. Savelsbergh, and X. Wang, "Optimization for dynamic ride-sharing: A review," *European Journal of Operational Research*, vol. 223, no. 2, pp. 295–303, 2012.

[13] J.-F. Cordeau and G. Laporte, "A tabu search heuristic for the static multi-vehicle dial-a-ride problem," *Transportation Research Part B*, vol. 37, no. 6, pp. 579–594, 2003.

[14] M. Brereton and S. Ghelawat, "Designing for participation in local social ridesharing networks: grass roots prototyping of it systems," in *Proceedings of the 11th Biennial Participatory Design Conference*. ACM, 2010, pp. 199–202.

[15] K. Radke, M. Brereton, S. Mirisaee, S. Ghelawat, C. Boyd, and J. Nieto, "Tensions in developing a secure collective information practice-the case of agile ridesharing," in *Human-Computer Interaction–INTERACT 2011*. Springer, 2011, pp. 524–532.

[16] M. Duckham and L. Kulik, "A formal model of obfuscation and negotiation for location privacy," in *Pervasive Computing*, H.-W. Gellersen, R. Want, and A. Schmidt, Eds. Springer-Verlag, 2005, pp. 152–170.

[17] M. Rigby and S. Winter, "Enhancing launch pads for decision-making in intelligent mobility on-demand," *Journal of Location Based Services*, vol. 9, no. 2, pp. 77–92, 2015.

[18] H. J. Miller, "Activities in space and time," in *Handbook of Transport Geography and Spatial Systems*, D. Hensher, K. Button, K. Haynes, and P. Stopher, Eds. Emerald Group Publishing, 2004, pp. 647–660.

[19] B. Kuijpers and W. Othman, "Modeling uncertainty of moving objects on road networks via space-time prisms," *International Journal of Geographical Information Science*, vol. 23, no. 9, pp. 1095–1117, 2009.

[20] G. Andrienko, N. Andrienko, P. Bak, D. Keim, and S. Wrobel, *Visual Analytics of Movement*. Berlin Heidelberg: Springer-Verlag, 2013.

[21] A. M. MacEachren, *How Maps Work: Representation, Visualization, and Design*. New York, U.S.A.: The Guilford Press, 1995.

[22] J. F. Hughes, A. van Dam, M. McGuire, D. F. Sklar, J. D. Foley, S. K. Feiner, and K. Akeley, *Computer graphics: Principles and Practice*, 3rd ed. Addison-Wesley Professional, 2013.

[23] A. Skupin, "Discrete and continuous conceptualizations of science: Implications for knowledge domain visualization," *Journal of Informetrics*, vol. 3, no. 3, pp. 233–245, 2009.

[24] P. H. L. Bovy and E. Stern, *Route Choice: Wayfinding in Transport Networks*. Dordrecht: Kluwer Academic Publishers, 1990.

[25] M. E. Ben-Akiva and S. R. Lerman, *Discrete Choice Analysis: Theory and Application to Travel Demand*. Cambridge, MA, USA: The MIT Press, 1985.

[26] M.-P. Kwan and X.-D. Hong, "Network-based constraints-oriented choice set formation using GIS," *Geographical Systems*, vol. 5, pp. 139–162, 1998.

[27] T. Hägerstrand, "What about people in regional science?" in *Papers of the Regional Science Association*, vol. XXIV, 1970, pp. 7–21.

[28] M. Raubal, H. J. Miller, and S. A. Bridwell, "User-centred time geography for location-based services," *Geografiska Annaler: Series B, Human Geography*, vol. 88B, no. 4, pp. 245–265, 2004.

[29] A. Pfoser and C. Jensen, "Capturing the uncertainty of moving-object representations," in *Advances in Spatial Databases*, R. Güting, D. Papadias, and F. Lochovsky, Eds. Springer, 1999, pp. 111–131.

[30] H. J. Miller and S.-L. Shaw, *Geographic Information Systems for Transportation: Principles and Applications*. New York, NY, USA: Oxford University Press, 2001.

[31] V. Bauer, J. Gamper, R. Loperfido, S. Profanter, S. Putzer, and I. Timko, "Computing isochrones in multi-modal, schedule-based transport networks," in *Proceedings of the 16th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*. New York, NY, USA: ACM, 2008.

[32] C. F. Daganzo and Y. Sheffi, "On stochastic models of traffic assignment." *Transportation Science*, vol. 11, no. 3, pp. 253–274, 1977.

[33] E. Hyytiä, S. Aalto, A. Penttinen, and R. Sulonen, "A stochastic model for a vehicle in a dial-a-ride system," *Operations Research Letters*, vol. 38, no. 5, pp. 432–435, 2010.

[34] E. Royer and C.-K. Toh, "A review of current routing protocols for ad hoc mobile wireless networks," *Personal Communications, IEEE*, vol. 6, no. 2, pp. 46–55, 1999.

[35] A. U. Frank and S. Timpf, "Multiple representations for cartographic objects in a multi-scale tree - an intelligent graphical zoom," *Computers & Graphics*, vol. 18, no. 6, pp. 823–829, 1994.

[36] N. Ronald, "Workers, adventurers, explorers: uncovering activity patterns in Melbourne," in *Proceedings of the 36th Australasian Transport Research Forum (ATRF)*. Brisbane, Australia, 2013.

[37] B. Zondag and K. Guers, "Coupling a detailed land-use model and a land-use and transport interaction model," in *Land-Use Modelling in Planning Practice*, E. Koomen and J. Borsboom-van Beurden, Eds. Springer, 2011, pp. 79–95.

[38] S. Winter and Z.-C. Yin, "Directed movements in probabilistic time geography," *International Journal of Geographical Information Science*, vol. 24, no. 9, pp. 1349–1365, 2010.

[39] Y. Song, H. Miller, and X. Zhou, "Modeling visit probabilities within network time prisms using continuous-time semi-markov techniques," in *Transportation Research Board 94th Annual Meeting*. Washington DC, USA: Transportation Research Board, 2015, p. 15:4241.

[40] G. Correia and J. M. Viegas, "Carpooling and carpool clubs: Clarifying concepts and assessing value enhancement possibilities through a stated preference web survey in Lisbon, Portugal," *Transportation Research Part A: Policy and Practice*, vol. 45, no. 2, pp. 81–90, 2011.

[41] F. K. Ozenc, L. F. Cranor, and J. H. Morris, "Adapt-a-ride: Understanding the dynamics of commuting preferences through an experience design framework," in *DPPI'11: Proceedings of the 2011 Conference on Designing Pleasurable Products and Interfaces*, no. 61. New York, NY, USA: ACM, 2011, pp. 1–8.

[42] S. Iyengar, *The Art of Choosing*. London, UK: Little Brown, 2010.

[43] B. Barabino, S. Salis, and A. Assorgia, "Application of mobility management: a web structure for the optimisation of the mobility of working staff of big companies," *IET Intelligent Transport Systems*, vol. 6, no. 1, pp. 87–95, 2012.

[44] H. Sharp, Y. Rogers, and J. Preece, *Interaction Design: Beyond Human-Computer Interaction*, 4th ed. West Sussex, UK: John Wiley & Sons, 2015.

Michael Rigby received the B.GeomE (Hons.) degree in Engineering from The University of Melbourne and B.Des (Multimedia Design) degree from Monash University, Australia. He is a PhD candidate in the Dept. of Infrastructure Engineering at The University of Melbourne. His research focuses on human-computer interfaces for intelligent transportation systems in on-demand scenarios.

Mr. Rigby is a member of ACM SIGSPATIAL and the Surveying & Spatial Sciences Institute.

Stephan Winter is Professor for Geographic Information Science, and Discipline Leader, Geomatics, at the University of Melbourne, Australia. His research centers on human wayfinding, spatial cognitive engineering, and intelligent transportation.

Prof. Winter is a member of the ISPRS International Science Advisory Committee and currently holds the chair of the Australasian Spatial Information Education and Research Association.

Antonio Krüger is Professor for Computer Science at Saarland University, heading the Ubiquitous Media Technology Lab, and the Scientific Director of the Innovation Retail Laboratory (IRL) at the German Research Center for AI (DFKI). His major research areas include intelligent user interfaces, mobile and ubiquitous context-aware systems.

Prof. Krüger has served on various committees in intelligent mobile systems e.g. Intelligent User Interfaces, Ubicomp, Mobisys, and Pervasive Computing.