Gander: Personalizing Search of the Here and Now

Jonas Michel[1], Christine Julien[1], Jamie Payton[2], and
Gruia-Catalin Roman[3]

[1] The University of Texas, Austin, TX, USA, {jonasrmichel, c.julien}@mail.utexas.edu
[2] The University of North Carolina, Charlotte, NC, USA, payton@uncc.edu
[3] The University of New Mexico, Albuquerque, NM, USA, gcroman@unm.edu

TR-ARiSE-2011-007

Gander: Personalizing Search of the Here and Now

Jonas Michel[1], Christine Julien[1], Jamie Payton[2], and Gruia-Catalin Roman[3]

[1] The University of Texas, Austin, TX, USA, {jonasrmichel, c.julien}@mail.utexas.edu
[2] The University of North Carolina, Charlotte, NC, USA, payton@uncc.edu
[3] The University of New Mexico, Albuquerque, NM, USA, gcroman@unm.edu

Abstract. In Personalized Networked Spaces (PNets), people and devices are integrated with the environment and demand fluid interactions to enable connectivity to information, services, and people. PNet applications exhibit significant *spatiotemporal* demands in which connectivity to resources and information is personalized and focused on the *here* and *now*. We introduce *Gander*, a personalized search engine for the here and now. We examine how search expectations are affected when users and applications interact directly with the physical environment. We define a formal conceptual model of search in PNets that provides a clear definition of the framework and ultimately enables reasoning about relationships between search processing and the relevance of results. We assess our model by evaluating sophisticated Gander queries in a simulated PNet.

1 Introduction

In Personalized Networked Spaces (PNets), people and machines interact to make the physical environment more digitally accessible. PNets are made possible by wireless and sensor technology, cyber-physical systems, and a deeper understanding of the importance of social networks. Key PNet features are spatiotemporal locality and a rapidly changing world; knowing what is happening to me and around me now is of the essence. PNets are likely to become ubiquitous and critical to our economy and social life.

PNets will serve different purposes and entail different interactions than the Internet, but the same fundamental need to access information exists. PNets, however, require tight spatiotemporal integration of user behavior and the immediate environment; information needs are immediate, personalized, and localized. Internet search engines treat the Internet as a large (relatively static) library for rapid access to relevant information; enormous technological, intellectual, and organizational resources support such a massive undertaking. In contrast, a PNet search engine needs to facilitate immediate interactions with one's surroundings without infrastructure and advance indexing.

Fluidity and extreme dynamics are defining features of PNets that motivate the kinds of search users need. In the Internet, one may ask for webcams in Geneva; in a home one may want to find where the dog is hiding using motion detectors and webcams. Using an Internet search engine one may find available trains from Rome to Florence; while hurrying to board a crowded train at the station one may need to find an available

This material is based upon work supported by the National Science Foundation under Grant No. 844850. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation (NSF).

seat in a second class car. In the Internet, one might ask for an aerial photo of a forest fire, which may be acceptable even if a few hours old; when running away from the fire one needs to know the relation between fire and wind in the immediate neighborhood.

An Internet-centric perspective might frame the question as how to extend an existing search engine to PNets. However, PNets' dynamic and heterogeneous nature give rise to novel challenges. We pose a fundamental research question: *Given a PNet, what mechanisms are required to provide search capabilities without reliance on Internet connectivity?* The latter requirement is justified by practical considerations. There are indeed situations in which Internet access is costly, inconvenient, nonexistent, or simply unnecessary; a home is managed better by exploiting local wireless connectivity, an exploration team may be far from any connectivity other than a satellite, in underdeveloped countries connectivity to the Internet may be sparse or financially unattainable.

This paper's contribution is to introduce *Gander*—a personalized search engine for PNets—and to investigate how existing query protocols can be used to support personalized search of the here and now. Gander queries about "me, here, and now" have a strong spatiotemporal component; the location of the user is a key point of reference. Our conceptual model is based on a worldview in which reachable nodes have locations, are semantically related, and have attributes of interest to the user. An important departure from traditional query processing is the tight integration of *personalized relevance* (the PNet analog to page ranking) into query processing. This personalized relevance is dynamic and reflects the user's personalized priorities here and now; it affects the perception of the search process in a fundamental and novel way and it holds the promise to reduce the cost of processing by leveraging relevance in the query protocol.

2 Background and Motivating Example

We first consider a concrete motivating scenario; we then look at the state of the art in related areas to demonstrate the important gap that the Gander approach fills.

2.1 Application Example

While planning a trip to an amusement park, visitors may use the Internet to find directions, opening hours, etc. Most of this information is static or updated sporadically. In the park, visitors' information needs change. A visitor may want to know which rides have the shortest wait right now, where the nearest exit is, or how others like a ride. Providing this information via the Internet is problematic: the volume of information is too great to be shipped and stored centrally, the lifespan of information is short, and the ratio of data used to data available is minute. A visitor's requests for information "here and now" must be served by local, dynamically formed networks of devices in the park.

Consider a visitor, Maya, who wants to discover if she has time to go on a ride before meeting her friends for lunch. She submits a query over a dynamically formed network of infrastructure and personal mobile devices that are reachable (possibly across multiple hops) from her mobile device. Visitors' devices may provide information about density of visitors in regions of the park and information collected about attractions or restaurants they have recently encountered. Constraints on the query can restrict the logical search space; to meet her friends for lunch, Maya needs to find a ride close to a food vendor, so her query uses a proximity constraint to restrict the search space.

Although the constraints limit the search scope, several rides may satisfy Maya's query. Since Maya needs to make it to lunch in time, her query emphasizes a combination of the ride wait time and the distance from the ride to the food vendor. Such relevance metrics may be evaluated over devices and their associated data that are not part of the result set for the query and its constraints. For example, wait time information may be acquired from the mobile devices of other visitors in line for the ride. Many other queries are possible with different search parameters, constraints, and ranking metrics. The concepts are clearly not limited to this particular domain. The notions of using semantic relationships among data items to evaluate queries and providing a description of relevance with each query to provide personalized search of the here and now can be generalized and applied to many other applications and spaces.

2.2 Related Work

Mobile devices, location-based services, and the embedding of computation sets the stage for search capabilities that are inherently intertwined with our personalized spaces. This section examines the state of the art and identifies the gap Gander fills.

Mobile Information Retrieval. Recent work has assessed requirements for information retrieval and identified relationships between users' information needs and their *context*, including location, time, and even activity [17, 23]. Such studies have exposed influences on people's willingness to share information about themselves; for example, people may be willing to share with other co-located users (even unknown ones) what they would not be willing to share publicly (i.e., on the Internet).

Systems have extended spatiotemporal search techniques to mobile applications. WebPark [23] provides geographically relevant information using geographic filters to capture spatial relevance. Ahlers and Boll enrich spatial context with temporal aspects [1]. GeoRel [26] develops metaphors for assessing relevance of geospatial features. These systems allow users to search relatively *static* information (e.g., landmarks) and focus on how to use geographic context to determine which information is most relevant. In a PNet, the data searched is *ephemeral*; such transient data cannot be easily indexed and associated with a relevant spatial context outside of the here and now.

Moving object databases [9, 10] and spatiotemporal databases [15] have driven spatiotemporal data representation. Erwig et al. use abstract data types that map time onto points, lines, and spatial regions, enabling spatiotemporal predicates, relationships between spatial entities changing over time [13, 14]. This and similar approaches enable effective spatiotemporal search; however, they require access to a centralized library of continuously cataloged information. PNets, in contrast, demand the ability to access instantaneously available local information with transient spatial relationships.

The Internet of Things. The Internet of Things [3] envisions a world in which ordinary objects are imbued with computation, sensing, and networking capabilities. Such a vision makes possible and necessary the ability to perform real-time local searches. A fundamental building block of this vision are *smart objects* [7, 18]; the wide availability of such smart objects drastically increases the amount of dynamic and transient information in our spaces. Successful applications in the Internet of Things must be able to search this large volume of data here and now efficiently.

Dyser [24] is a search engine that supports real-time search for sensors with a user-specified current state. Searches are optimized using a technique called sensor ranking [12] by assuming that searchable entities are persistently accessible via the Internet. In a PNet, however, reachability is volatile and must be assessed instantaneously.

Relevance and Recommender Systems. Relevance in traditional web search is well-understood [22], even along multiple dimensions [2]. Geographic digital libraries [1, 5, 19] use geospatial metadata (e.g., coordinate representations of geographic objects) to support storage and retrieval of geographic information. A document's relevance may be computed in terms of its geospatial metadata's topological relation to a query (e.g., size, shape, location, distance) in addition to its content's relation to a search query [11].

Gander demands a richer awareness of *context* in determining relevance, yet contextually influenced information retrieval is challenging due to an inability to appropriately index information when it is generated [20]. Collaborative filtering combines results from multiple users to make recommendations, using a variety of techniques to acquire a user's context [27]; the approaches falter in PNets because explicit computation and comparison of rich contexts is complex even for simple queries [8]. Our approach does not *index* data relative to its context but instead performs the query *in* the context. Future extensions to Gander could dovetail more explicitly with collaborative filtering, learning how other users interact with the information in this space at this time.

3 The Gander Conceptual Model

Personalized search of the here and now entails two components: (1) *query processing*: i.e., how to distribute a query (and its responses) using local interactions; and (2) *relevance determination*: i.e., how to rate results' relevance to the query. We next describe the Gander conceptual model, which is essential for providing a rigorous understanding of the *quality* of a Gander query and its ability to represent the here and now.

3.1 Gander Conceptual Model Components

A *PNet* is a collection of nodes connected via a dynamic topology. Nodes issue queries that are evaluated across information stored at other nodes. A *data item* is an atom (ν, d), where ν is a data value (e.g., a measure of some condition in the environment), and d is meta-data associated with that piece of data (e.g., the device(s) that generated the data, the data's location, a timestamp, etc.). This representation reflects existing work in capturing and representing context in ubiquitous computing [6].

For every valid result for a query, the result's ν must "match" the search string. A query can include one or more additional *query constraints*. For each constraint c, valid results are those whose ν values match the search string and $c((\nu, d))$ returns true. A *query relevance metric* compares valid results against each other. For each relevance metric m, $m((\nu, d))$ returns a value; these values can be ordered. A query can use multiple metrics evaluated independently or using weighted statistics. A *query processing* *protocol* executes in a distributed multihop fashion to select a subset of the PNet to distribute a query to; this subset includes only "reachable" nodes in the PNet. Gander query processing can use the search string, constraints, and relevance metrics.

3.2 Gander Query Processing

A Gander query is a function $G_h : D \rightarrow \Phi$, where D contains all data items in the world, Φ is the domain of the relevance metric, and h is the node issuing the query. A query processing protocol is a *partial function* $\text{QP}_h : D \rightarrow D$; QP_h is not required to be defined for every element of its domain, D. Informally, $\text{QP}_h((\nu, d)) = (\nu, d)$ if $(\nu, d) \in D$ is a "valid" result; otherwise $\text{QP}_h((\nu, d))$ is undefined. One can view QP_h as a *filter* on D. The three components of are the following.

Reachability. The partial function $\mathcal{R}_h : D \rightarrow D$ expresses whether the data item (ν, d) is *reachable* from h; if not, $\mathcal{R}_h((\nu, d))$ is not defined. This is a specialization of a general reachability function; $\mathcal{R} : H \times H \rightarrow \{0, 1\}$, written as $\mathcal{R}(h_1, h_2)$, which expresses whether h_2 is *reachable* from h_1. We focus on *query* reachability, the ability to send a query from h_1 to h_2 *and* receive a response [25]. $\mathcal{R}$ is related to $\mathcal{R}_h$ as: $\mathcal{R}_h((\nu, d)) = (\nu, d) \Leftrightarrow \langle \exists h_2 : (\nu, d) \in h_2 :: \mathcal{R}(h, h_2) \rangle$, where $(\nu, d) \in h_2$ indicates that (ν, d) is "owned" by h_2. Clearly $\mathcal{R}$ depends on both physical communication capabilities and communication protocols used in the PNet.

Query Resolution. The partial function $\mathcal{S} : D \rightarrow D$ is defined for each $(\nu, d) \in D$ that matches the search string. In our application scenario, $\mathcal{S}$ indicates whether or not the data item (ν, d) is a reachable data item describing a ride.

Query Constraint. The partial function $\mathcal{C} : D \rightarrow D$ is defined for each $(\nu, d) \in D$ that satisfies the query constraints. $\mathcal{C}$'s resolution may rely on the *meta-data* (d), which may provide the context of that data item in the PNet. In our scenario, an example constraint is whether (ν, d) refers to a ride near a food vendor.

These three functions filter D to the subset of reachable data items that match the search string and satisfy the constraints. Given a snapshot of all data items in the PNet, Fig. 1 shows the composition of reachability, query resolution, and query constraints; $\text{QP}_h = \mathcal{C} \circ \mathcal{S} \circ \mathcal{R}_h$. This model assumes complete knowledge of the PNet. Practically, this is not reasonable, but this formulation of query processing as a composition of partially defined functions allows us to incrementally compute a query's result.

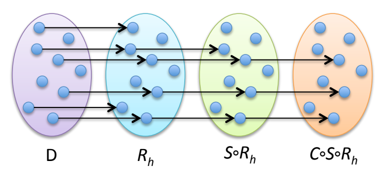


Fig. 1. Query processing functions

3.3 Gander Relevance Model

A Gander query defines regions of the PNet from which to draw data to satisfy a personalized search of the here and now. Gander query constraints and relevance definitions provide criteria by which data in these regions is selected for delivery to the query issuer. Our goal is not to collect all of the data but only the data that is most relevant.

A Gander *relevance metric*, $\mathcal{M}_i : D \rightarrow \phi_i$, is a partial function; ϕ_i is the domain of the relevance metric $\mathcal{M}_i$. Elements of ϕ_i should be scalar; $\mathcal{M}_i((\nu, d))$ gives the distance of a data item (ν, d) from an ideal (with respect to the relevance metric $\mathcal{M}_i$). A Gander query may entail more than one relevance metric; given a query's n relevance metrics, the complete relevance metric is the partial function $\mathcal{K}_{\{\mathcal{M}_1,\mathcal{M}_2,\ldots,\mathcal{M}_n\}} : D \to \phi_1 \times \phi_2 \times \cdots \times \phi_n$, or simply $\mathcal{K}_{\{\mathcal{M}_1,\mathcal{M}_2,\ldots,\mathcal{M}_n\}} : D \to \Phi$. A Gander query, G_h, composes the complete relevance metric with a query processing protocol: $G_h = \mathcal{K}_{\{\mathcal{M}_1,\mathcal{M}_2,\ldots,\mathcal{M}_n\}} \circ \mathcal{C} \circ \mathcal{S} \circ \mathcal{R}_h$. This combination of query processing with relevance maps valid results to a multidimensional space whose dimensions are given by $\Phi = \phi_1 \times \phi_2 \times \cdots \times \phi_n$. $G_h((\nu, d))$ is not defined if (ν, d), is not reachable or does not satisfy the search string or query constraints; otherwise $G_h((\nu, d))$ is an n-tuple, where field i has the value $\mathcal{M}_i((\nu, d))$. We can plot each valid result in a multidimensional space, where each axis represents one of the n metrics; results closer to the origin are more relevant[1].

We give two examples of computing the distance from a point in this multidimensional space to the origin: *lexicographical ordering* and *weighted combination*. The former is less expressive but more flexible since any metrics can be combined. A weighted combination requires the metrics to be expressed in domains that are comparable (or could be made comparable by applying scaling factors) but is more expressive.

Lexicographically Ordered Relevance Metrics. In lexicographical ordering, the order of the metrics matters; data items are ranked first based on the first metric, and later relevance metric values break ties. Given two n-tuples k and k' representing relevance values for (ν, d) and $(\nu, d)'$, (ν, d) is more relevant if and only if:

$$\langle \exists i : 1 \leq i \leq n : k[i] < k'[i] \wedge \langle \forall j : 1 \leq j < i :: k[j] = k'[j] \rangle \rangle$$

Maya wants to maximize the thrill for the rides she experiences. Her query looks for rides she has not yet ridden, favoring first rides with the highest thrill and breaking ties by favoring those that are closer. Her friend Alex also likes thrill rides, but he wants to go on as many rides as possible. His query also looks only at rides he has not yet ridden, but first uses a relevance metric of distance and then one for thrill.

Weighted Combination of Relevance Metrics. This approach requires as input a vector $\mathcal{W}$ that associates a weight with each metric. We map each n-tuple to a single value: $m : \Phi \to \mathbb{R}$[2]. Given $\mathcal{W} = \langle w_1, w_2, \ldots w_n \rangle$, we can map a result's n-tuple to a weighted sum of the tuple's constituents: $m^{+}_{\mathcal{W}}((\nu, d)) = \sum_{i=1}^{n}(w_i \times \mathcal{M}_i((\nu, d))))$. $\mathcal{W}$ normalizes Φ's dimensions so metrics can be numerically combined. As an example, Maya's relevance metrics can express the travel time to a ride from her location, the travel time from that ride to a food vendor, and the ride's wait time; by weighting these component metrics equally and summing their values, we can order the rides according to the total time it would take Maya to ride the ride and meet her friends.

The multiplicative combination, $m^{\times}_{\mathcal{W}}$, can be defined similarly. This combination can support a user who wants to ride the most thrilling rides but only has two hours. The wait time relevance metric maps to a value between 0 and 1. When this wait time

1 We assume the origin is absolute and relevance metrics are defined relative to an ideal. The origin could itself be relative to the results; this causes a translation of the axes of the multidimensional space to move the origin. The same distance functions can be applied.

2 We use $\mathbb{R}$ as the function's range; for other combinations, the range could be any set over which a partial order can be defined.

metric is part of a multiplicative combination with a thrill metric, rides that the user does not have time to enjoy will have a total relevance ranking of 0.

These approaches to combining multidimensional relevance linearly order the results; future work will consider more complex yet expressive relevance combinators. In the remainder of this paper, we examine how well existing query protocols for mobile networks support this conceptual model. This paper's novelty is in defining the conceptual model and using existing query protocols to distribute personalized queries of the here and now. We do not define novel protocols; instead we use our evaluation of the Gander conceptual model to posit how existing protocols could be modified (or replaced) to better support personalized search of the here and now.

4 Evaluation

In this section, we evaluate the Gander conceptual model to ascertain how well existing protocols for mobile networks can support personalized searches of the here and now.

We developed a simulation-based implementation that executes each query over a model of PNet data and dynamics and returns a list of ranked results. We use our amusement park scenario and data we collected about Disney World's Magic Kingdom in Orlando, Florida. Our simulation includes static nodes that represent 30 attractions, 12 restaurants, and 8 restrooms. We populated the park with 20,000 visitors and simulate visitors' movements along paths at an average speed of 0.5 mph. Fig. 2 shows an example of the input; yellow discs are rides, green discs are restaurants, orange discs are restrooms, and pink spots indicate visitors. In this figure we only placed 100 visitors so that other elements are visible. We use ride wait times published by Lines [21]; we collected wait time data every 60 seconds on a single day. Simulated visitors carry timestamped data about amenities they have visited recently. Static nodes provide data about fixed entities and metadata (e.g., ride name, its thrill factor and location).

Fig. 2. The data input to our query simulator

In our prototype, a Gander query can choose from one of four protocols: *flooding*, *probabilistic propagation*, *location-based*, and *probabilistic sampling*, which are representative of commonly used paradigms in mobile querying. Fig. 3 depicts each protocol. Our evaluation focuses on the utility of Gander's relevance metrics to personalize search and the impact of different query protocol behaviors on the quality of search results.

We issued six queries, all with the same string ("thrill ride") and constraint ("within 75 ft of a restaurant") but different relevance metrics: **Q1:** wait time; **Q2:** distance from me; **Q3:** thrill; **Q4:** wait time then distance from me; **Q5:** wait time then thrill; **Q6:** thrill then wait time. We report results using three of Gander's protocols: flooding with a range of three hops, probabilistic with a range of three hops and propagation probability of 0.5, and probabilistic sampling with a probability of responding of 0.5; we limit propagation to within 10 hops for tractability. A node is considered reachable from

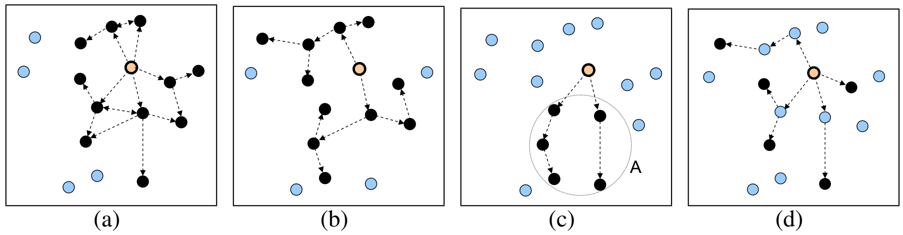


Fig. 3. Query routing protocols. (a) Flooding. Every node in a given range (2 hops) retransmits the query. (b) Probabilistic Propagation. Every node in a given range (3 hops) that receives the packet retransmits it to 2 random neighbors. (c) Location. The query reaches nodes in region A. (d) Probabilistic Sampling. The query reaches any 5 nodes.

another if the two are within 7.5 feet of each other; we model dropped packets using a loss probability associated with a link and experiment with different link qualities.

Fig. 4 shows a sample result. Here, the Gander result is a subset of the ideal with small differences. One result is missing (Tom Sawyer Island), and some data is old (as measured by timestamps). This results in stale data (e.g., the wait time for Pirate Tutorial). Such differences are likely to be small in PNets (since the data collected is relatively fresh), but they can impact relevance rankings. The alternative, i.e., searching via the Internet, would be more severely impacted by even moderately out of date data. This highlights a fundamental differences between PNets and the Internet: data volatility.

```
2011-04-09 15:00:00.0 Tom Sawyer Island {waittime=5.0, thrill=2}
2011-04-09 15:00:00.0 Regal Carrousel {waittime=5.0, thrill=3}
2011-04-09 15:00:00.0 Mickey's PhilharMagic {waittime=10.0, thrill=4}
2011-04-09 08:37:00.0 Pirate Tutorial {waittime=26.0, thrill=5}
2011-04-09 15:00:00.0 Dumbo {waittime=19.0, thrill=5}
2011-04-09 15:00:00.0 Snow White {waittime=17.0, thrill=6}
2011-04-09 15:00:00.0 Pirates of the Caribbean {waittime=12.0, thrill=6}
2011-04-09 15:00:00.0 Tomorrowland Speedway {waittime=29.0, thrill=8}
2011-04-09 15:00:00.0 Splash Mountain {waittime=29.0, thrill=10}
```

(a)

```
2011-04-09 15:00:00.0 Regal Carrousel {waittime=5.0, thrill=3}
2011-04-09 15:00:00.0 Mickey's PhilharMagic {waittime=10.0, thrill=4}
2011-04-09 15:00:00.0 Dumbo {waittime=19.0, thrill=5}
2011-04-09 08:32:00.0 Pirate Tutorial {waittime=21.0, thrill=5}
2011-04-09 15:00:00.0 Snow White {waittime=17.0, thrill=6}
2011-04-09 14:55:00.0 Pirates of the Caribbean {waittime=12.0, thrill=6}
2011-04-09 15:00:00.0 Tomorrowland Speedway {waittime=29.0, thrill=8}
2011-04-09 14:20:00.0 Splash Mountain {waittime=29.0, thrill=10}
```

(b)

Fig. 4. Results for Query 3. (a) ideal query results. (b) Gander query results with flooding

Gander Queries Reflect the Ground Truth. We evaluate Gander using the discounted cumulative gain (DCG) [16] of a ranked list of results, which compares how useful a result is to the user and its ranked position in the set. The intuition is simple: since a result with a lower ranking is less likely to be used, a result's gain is discounted as the position decreases. We compare the result of executing a Gander query with the *ground truth*, i.e., we compute the DCG for a list of search results as:

$DCG_p = \sum_{i=1}^{p} \frac{rel_{d_i}}{\log_2(i+1)}$, where p is the size of the result set, and rel_{d_i} is an integer that reflects the result's position in the ideal query's rankings. A query result d returned with ranking i is graded on a scale from 1 (least relevant) to n, where n is the number of items returned by the ideal query. The value of rel_{d_i} is determined by the ranked position j of the item in the ideal result: $rel_{d_i} = n/j$. We normalize computed DCG values based on the ideal result by calculating the DCG for the ideal result ($IDCG$) and dividing each DCG by $IDCG$, generating a set of normalized values, $nDCG$.

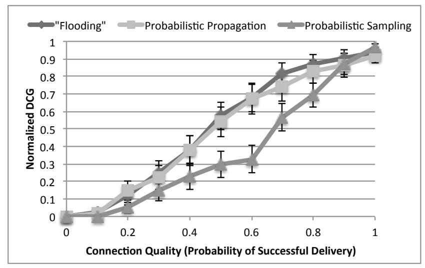
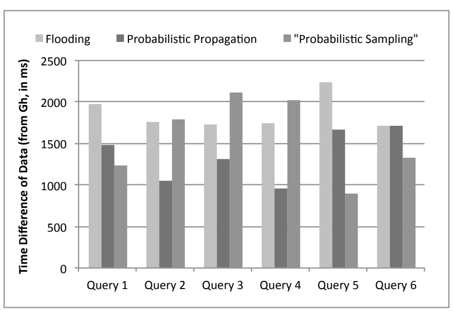
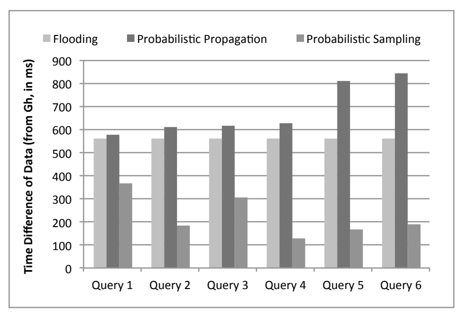


Fig. 5. Query Quality vs. Link Quality

We use DCG values to evaluate how well queries reflect the PNet, and we plot this for varying connection qualities for Query 1 (Fig. 5). Different query protocols perform differently. All of the protocols do a good job of reflecting the ground truth, especially under reasonable connection qualities ($> 50\%$ success of delivery). Probabilistic Sampling performs poorly when the connection quality is poor; this is because the queries propagate over longer paths, sampling a wider space of the PNet. These longer paths are more susceptible to low link quality, as the probability of a dropped packet is multiplicative. However, when the quality of the links is high, Probabilistic Sampling's performance surpasses the other protocols, motivating that certain queries need to broaden their search space and query protocols that enable such capabilities are necessary. Flooding and Probabilistic Sampling, on the other hand, sampling sometimes suffer from localization around the query issuer.



(a)



(b)

Fig. 6. Age of responses (a) 50% of packets dropped (b) perfect connectivity.

Another way to highlight the difference between the Gander query and the ideal one is to look at the difference in results' time stamps. Fig. 6 shows the measured time stamp differences for the three protocols across all six queries. Fig. 6(a) shows that, given faulty communication links, the time stamp differences between an ideal query and a Gander query result vary widely. However, as the quality of the links increases, we see a drastic difference between the freshness of the data returned by Probabilistic Sampling in comparison to the other protocols. This is especially evident when the loss probability decreases to 0, shown in Fig. 6(b). The density of connections in a PNet (our nodes had an average of 18 network neighbors) makes it possible to select nodes at random to sample, and this Probabilistic Sampling achieves better coverage than flooding or probabilistic sampling since the latter end up restricting their searches to nearby devices. This latter result specifically motivates that some situations may demand reliable communication protocols that better enable search to reflect the ground truth efficiently. More generally, these results demonstrate that having a formal understanding of the structure of the results of queries given different query processing protocols could be informative in selecting the appropriate query protocol for a given query, user, and environment. Future work will build on our existing conceptual model to create an understanding of incremental Gander query processing that relates the results of a query processing protocol (including relevance rankings) to the ideal result. This will allow formal mathematical reasoning about the quality of a Gander query result.

Relevance Matters. It is common to compare rankings by counting *inversions* [4]. Given two rankings, we fix one set of rankings and compute how much out of order results in the other ranking are. To demonstrate that using different relevance metrics has a significant impact on results, we compute the number of inversions returned by queries that are identical other than their use of relevance. Fig. 7 compares our six queries, given everything else (the network, the location of the query issuer, and the state of the data) is held constant. This identifies two equivalence classes of queries: $\{$Query 1, Query 4, Query 5$\}$ and $\{$Query 3, Query 6$\}$. The average number of inversions of rankings from Query 3 and Query 6 was 0.78. Within the class $\{$Query 1, Query 4, Query 5$\}$, the average number of inversions was always less than 1. For all other combinations, there was a significant number of inversions ($>\sim 2$) when comparing two result sets' rankings. Given our relevance metrics, this is reasonable. For example, Query 1 uses only the single relevance metric wait time, while Query 4 uses a relevance metric of wait time breaking ties using the distance from the query issuer. Inversions in rankings returned by these two queries result exactly from resolving ties.

Query	1	2	3	4	5	6
1	0	3.74	2.29	0.85	0.91	1.08
2		0	3.40	3.65	3.93	3.53
3			0	2.35	2.12	0.78
4				0	0.96	2.12
5					0	1.91
6						0

Fig. 7. Average inversion counts

Clearly, personalized relevance metrics have a dramatic effect on which results are presented and in what order. This could drive future work in assistive Gander interfaces that help users identify similar queries or queries that will help them more effectively distinguish physical aspects of the PNet environment.

5 Conclusions

It is apparent that existing query processing protocols for dynamic networks are a feasible starting point for supporting personalized search of the here and now. Our evaluation and conceptual model have exposed two key aspects necessary for Gander queries in these dynamic environments: (*i*) search processing must be tailored to PNets and search of the here and now and (*ii*) the semantics of search execution must be formalized (i.e., Q, the Gander query, must be formally related to G_h, the ideal result).

PNet Search Requires Tailored Reflective Protocols. Different styles of search processing provide different qualities of results in terms of their timely reflection of the ground truth. Gander queries inherently carry information that can benefit the protocols' execution, for example by directing queries towards areas of the PNet more likely to have relevant results or by taking advantage of reliable connectivity when possible to achieve a wider coverage of the space sampled by the query. Thus the results above motivate *reflective* protocols that can adapt their behavior to the nature of the PNet, its capabilities, the data it contains, and the results of previous similar searches.

We omitted a performance evaluation in this paper. As we develop protocols tailored for Gander queries, it will be essential to perform a thorough performance evaluation that measures the network aspects of Gander query processing from all directions, including measuring the overhead (in terms of communication, storage, and energy consumed on users' devices) and delay in receiving query results, coupling evaluation in simulation with evaluation in live PNets.

Search Execution Demands Formal Semantics. More interestingly, the use of partial functions in our conceptual model maps intuitively to *incremental evaluation* in which a search execution starts with a completely undefined function and gradually increases the size of the subset of the domain for which the function is defined.

One may expect that query execution should yield G_h, but achieving this exact reflection of the environment is unreasonable in practical PNets. Instead, we want to provide guarantees with which a query protocol realizes G_h in computing Q. If the protocol maintains a relationship between Q and G_h that has certain structural properties, we can reason about the results in the context of the posed query. An example of a query protocol that exhibits good structure is one that ensures that $Q \subseteq G_h$, i.e., Q will never contain a result that is not in G_h, but there may exist valid results we have not (yet) discovered. Results' relevance values are correct in G_h, i.e., the relative ordering of elements a and b in Q is the same as their relative ordering in G_h. We could envision a weaker relationship in which Q is always less defined than G_h, but in addition the values returned in Q may be somewhat different (within some τ) of the actual values. We could also envision the opposite: that $Q \supseteq G_h$, i.e., we start by assuming that the set of valid results is *larger* than the ideal query and gradually trim results out of Q. Explicitly relating the structure of Q to G_h allows us to reason about the *fidelity* of a query (and its query protocol). Our goal is to create query protocols that (provably) maintain structural relationships between G_h and Q, though practically we can never know G_h.

As the digital world becomes increasingly accessible, we need to access the vast amount of information available in our Personalized Networked Spaces. We introduced Gander, a search engine for personalized search of the here and now. A key contribution of Gander is use of a multi-dimensional specification of relevance, which provides a personalized search of a PNet from the perspective of a query issuer. We derived a conceptual model that mathematically defines Gander query processing and relevance in terms of partial functions. This model gives a formal foundation that maps to the formulation of protocols that incrementally evaluate a Gander query and allows us to reason about the semantics of results returned by a distributed implementation of a Gander query. We have provided a simulation-based evaluation in a large-scale PNet; our results illustrate the expressiveness of this approach and highlight the fact that using personalized relevance metrics can give increased value to the user of ranked search results.

References

1. D. Ahlers and S. Boll. Beyond position–spatial context for mobile information retrieval systems. In *Proc. of WPNC*, pages 129–134, 2009.
2. J. Assa, D. Cohen-Or, and T. Milo. Displaying data in multidimensional relevance space with 2d visualization maps. In *Proc. of Visualization*, pages 127–134, 1997.
3. L. Atzori, A. Iera, and G. Morabito. The internet of things: A survey. *Computer Networks*, 54(15):2787–2805, 2010.
4. M. Bansal and D. Fernandez-Baca. Computing distances between partial rnakings. *Information Processing Letters*, 109(4):238–241, 2008.
5. K. Beard and V. Sharma. Multidimensional ranking for data in digital spatial libraries. *Int'l. J. on Digital Libraries*, pages 153–160, 1997.
6. C. Bettini, O. Brdiczka, K. Henricksen, J. Indulska, D. Nicklas, A. Ranganathan, and D. Riboni. A survey of context modelling and reasoning techniques. *Pervasive and Mobile Computing*, 6(2):161–180, 2010.
7. P. Bolliger and B. Ostermaier. Koubachi: A mobile phone widget to enable affective communication with indoor plants. In *Proc. of MIRW*, pages 63–66, 2007.
8. A. Chen. Context-aware collaborative filtering system: Predicting the user's preference in the ubiquitous computing environment. In *Proc. of LoCA*, 2005.
9. V. de Almeida, R. Guting, and T. Behr. Querying moving objects in secondo. In *Proc. of MDM*, page 47, May 2006.
10. N. Djafri, A. A. Fernandes, N. W., Paton, and T. Griffiths. Spatio-temporal evolution: querying patterns of change in databases. In *Proc. of ACMGIS*, pages 35–41, 2002.
11. M. Egenhofer and R. Franzosa. Point-set topological relations. *Int'l. J. of Geographical Info. Sys.*, 5(2):161–174, 1991.
12. B. M. Elahi, K. Römer, B. Ostermaier, M. Fahrmair, and W. Kellerer. Sensor ranking: A primitive for efficient content-based sensor search. In *Proc. of IPSN*, pages 217–228, 2009.
13. M. Erwig and M. Schneider. Developments in spatio-temporal query languages. In *Proc. of DEXA*, pages 441–449, 1999.
14. M. Erwig and M. Schneider. Spatio-temporal predicates. *IEEE Trans. on Knowledge and Data Eng.*, 14:881–901, 2002.
15. R. Guting, M. Bohlen, M. Erwig, C. Jensen, N. Lorentzos, E. Nardelli, M. Schneider, and J. Viqueira. Chapter 4: Spatio-temporal models and languages: An approach based on data types. In *Spatio-Temporal Databases*, pages 117–176. 2003.
16. K. Järvelin and J. Kekäläinen. Cumulated gain-based evaluation of IR techniques. *ACM Trans. on Info. Sys.*, 20:422–446, October 2002.
17. Q. Jones, S. Grandhi, S. Karam, S. Whittaker, C. Zhou, , and L. Terveen. Geographic place and community information preferences. *CSCW*, 17(2–3):137–167, 2008.
18. G. Kortuem, F. Kawsar, D. Fitton, and V. Sundramoorthy. Smart objects as building blocks for the internet of things. *IEEE Internet Computing*, 14(1):44–51, 2010.
19. M. Kreveld, I. Reinbacher, A. Arampatzis, and R. Zwol. Multi-dimensional scattered ranking methods for geographic information retrieval. *GeoInform.*, 9:61–84, 2005.
20. M. Lamming and W. Newman. Activity-based information retrieval: Technology in support of personal memory. In *Proc. of IFIP World Computer Congress on Personal Computers and Intelligent Systems – Information Processing*, 1992.
21. Disney World Lines App (Touring Plans). `http://www.touringplans.com/walt-disney-world-lines`.
22. C. D. Manning, P. Raghavan, and H. Schutze. *Intro. to Information Retrieval*. 2009.
23. D. Mountain and A. MacFarlane. Geographic information retrieval in a mobile environment: evaluating the needs of mobile individuals. *J. of Info. Science*, 33:515–530, 2007.

24. B. Ostermaier, K. Römer, F. Mattern, M. Fahrmair, and W. Kellerer. A real-time search engine for the web of things. In *Proc. of IOT*, pages 1–8, 2010.
25. V. Rajamani, C. Julien, J. Payton, and G.-C. Roman. Inquiry and introspection for non-deterministic queries in mobile networks. In *Proc. of FASE*, pages 401–416, 2009.
26. T. Reichenbacher. Geographic relevance in mobile services. In *Proc. of LocWeb*, 2009.
27. H. Si, Y. Kawahara, H. Kurasawa, H. Morikawa, and T. Aoyama. A context-aware collaborative filtering algorithm for real world oriented content delivery service. In *Proc. of Metapolis and Urban Life Workshop*, 2005.