On the Selection of Spatiotemporally Relevant Sensory Providers

Georgios Tychogiorgos
Electrical and Electronic Engineering
Imperial College
London SW7 2AZ, UK
g.tychogiorgos@imperial.ac.uk

Chatschik Bisdikian
IBM Research
Thomas J. Watson Research Center
Hawthorne, NY 10514, USA
bisdik@us.ibm.com

***Abstract*—The development of pervasive computing systems and services, where information will be distributed on-demand across heterogeneous networks, highlights the necessity for an effective framework to determine the relevancy of provided information with respect to one's needs. This paper considers the problem of selecting the most "spatiotemporally" relevant providers in order to meet a user's information needs over a time period of interest. Initially, the spatiotemporal relevancy metric is proposed to measure the degree of relevancy of sensory information with respect to both its spatial and temporal characteristics. Based on this metric, the selection of the most relevant set of providers under budget constraints is expressed as an integer programming optimization problem and a two-level dynamic programming (DP) algorithm is proposed to solve it optimally. Moreover, a number of alternative methods are proposed in order to accelerate the provider selection process by making approximations either to the overall optimization problem formulation or the relevancy calculation method itself. Finally, the performance of the proposed methods are examined both analytically and by simulation for a number of provider scenarios.**

I. Introduction

In [1][1], we considered a use scenario involving a city agency that wants to collect sensory information to build a city-wide air-quality map to support various services to the city dwellers and visitors. Because of lack of sufficient sensory resources and the geographical extend of the project, the agency (an information *consumer*) decides to engage a number of third party sensory information sources (the information *providers*). The specific challenge addressed in [1] was that of the consumer selecting the set of providers that are deemed most relevant to his information needs based on their geographical coverage, *quality of information* (QoI) [8] and cost constraints.

Clearly, this scenario can be mapped to a number of use cases involving various forms of participatory sensing where consumers may bind to providers on demand. This could be the case, for example, in coalition environments where a sensory application (e.g., for chemical presence detection) from one coalition member needs to engage sensory resources belonging to other coalition members to supplement its own resources (if any at all) in the various geographical areas of interest. The sensory resources and applications of the various coalition members may dynamically connect via a distributed publish/subscribe system, such as the *Information Fabric* [2].

Within this setting, [1] considered the use of the spatial and quality properties of information as a means to select the most spatially *relevant* of the available providers. It introduced the use of *QoI functions* to capture the quality properties of the *desired* and *provided* information, for example, in region $\mathcal{A}$, the city agency desires air-quality measurements that are within $\pm 5\%$ of their actual value. We then formulated optimization problems and developed algorithmic procedures for selecting one or more providers that provide the most spatially relevant information to the one desired with or without cost constraints. Spatial relevance relates to how close or far is, in some sense, the spatial coverage and quality of a provided piece of information to that desired, assuming, of course, the semantic equivalency between these pieces of information, e.g., concentration of chemical X.

The work in [1] considered only the spatial dimension of the problem, which allowed us to build the intuition and the foundations for this novel class of problems. That work reflected a *static* interpretation of the problem where consumer interests and provider capabilities did not change with time. In the current paper, we relax this static assumption and explicitly consider temporal variations as well and investigate the selection of providers that are spatiotemporally relevant. Specifically, the contributions made in this paper are:

- the introduction of (and metric for) *spatiotemporal relativity* between consumers and providers of information;
- the formulation of an optimization problem for selecting the most spatiotemporally relevant information providers;
- the solution to the problem using a dynamic programming (DP) based algorithm, along its performance and complexity assessment;
- the development of solution acceleration techniques, their performance and time-complexity trade-offs.

To the best of our knowledge, this research area of spa-

Research was sponsored by the U.S. Army Research Laboratory and the U.K. Ministry of Defence and was accomplished under Agreement Number W911NF-06-3-0001. The views and conclusions contained in this document are those of the author(s) and should not be interpreted as representing the official policies of the U.S. Army Research Laboratory, the U.S. Government, the U.K. Ministry of Defence or the U.K. Government. The U.S. and U.K. Governments are authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation hereon.

[1] A version of [1] had been presented in ACITA 2011.

tiotemporal relevancy of sensory information within the context of consumers and providers of information is a novel one. Even though the area appears, at first glance, to resemble that of sensor network coverage [3], [4] it is distinctively different with no prior art directly related to it. Coverage problems are concerned primarily with static deployment strategies of sensors of a single sensor network. They are not concerned with the operational aspects of dynamically selecting information providers (possibly representing multiple sensor networks) under spatially varying and temporally evolving interests and capabilities of consumers and providers, respectively, while considering *both* coverage *and* QoI aspects. Our research pursue in this area aims to build an understanding and the foundations of the process of selecting resources based on spatiotemporally evolving needs, expectations, and capabilities of multiple entities that provide and consume information.

While we could find no prior art on the subject to compare against, there are ancillary pieces of work, of course, that contribute to this research area such as [5] that reviews data models applicable to spatio-temporal databases and [6] that considers the summarization of 2D spatial shapes through a bounded number of parameters. Ref. [7] further considers approximations of multidimensional shapes via the use of spline surfaces. This background art, and references therein, could form the basis for several operational aspects of our proposals such as the QoI function management (e.g., advertisement and storage), however they do not pursue the objectives of our novel research direction.

The rest of the paper is organized as follows. Section II presents the notion of spatiotemporal relativity. Section III formulates the selection of the best set of providers as an integer programming problem and presents a DP algorithm to solve it. Section IV presents three alternative problems that provide approximation of the initial problem and Section V describes two relevancy assessment approximation methods. Section VI provides simulation results to evaluate the accuracy and time requirements of the proposed methods and, finally, Section VII concludes our work and outlines our future research.

II. Spatiotemporal relativity

In its broadest sense, the problem space at hand is that of information consumers selecting (and binding) to information providers that are most *spatiotemporally relevant* to the information needs of the consumers while satisfying stated operational constraints. In approaching this problem, first, we need to describe what spatiotemporal relevancy means.[2] We will do so by building upon the narrower focus of spatial relevancy considered in [1].

Let the *spatiotemporal point* w represent the pair (ω, t), where, say, $\omega = (x, y) \in \mathbb{R}^2$ and t is a time instant. Let R_e, $e = \{d, p\}$, be a *spatiotemporal volume* where the *desired* (d) or *provided* (p) quantities of interest are defined, e.g., R_d denotes a region of interest, such as the southern part of the city during a weekday evening rush hour. For each $w \in R_d$, let $q_d(w)$ represent the quality of desired information, e.g., its accuracy, latency, etc., i.e., the QoI a consumer, e.g., the city agency, seeks [8] . Likewise, for each $w \in R_p$, let $q_p(w)$ represent the QoI provided by an information provider, e.g., by a sensory information provider owning air-quality measuring sensors mounted on buildings or on vehicles of a vehicle fleet, etc. We refer to $q_e(\cdot)$, $e = \{d, p\}$, as the (spatiotemporal) *QoI function* and, by convention, we assume that $q_e(w) > 0$ for $w \in R_e$, and $q_e(w) = 0$ for $w \notin R_e$.

[2]As noted in the introduction, we assume semantic equivalency of the sought after and provided pieces of information.

For given $q_d(\cdot)$ and $q_p(\cdot)$, and $w \in R_d \cap R_p$, let the *value of information function* (or *VoI function*) $v\big(q_p(w); q_d\big)$ represent the benefit (or value) that the consumer attains when using information of quality $q_p(w)$ when q_d was desired. Note that from an operational standpoint, we expect that information seekers will advertise their QoI desires instead of their VoI ones. For example, the city agency may advertise its desire to collect air-quality measurements that have (at least) 90% accuracy, but not why it needs such accuracy.

We use the VoI (anticipated to be) received from a provider to define a measure of the *instantaneous relevancy* $r(q_d, q_p; t)$ of the provider at time t the ratio of the total value to be gained from the information provided (by the provider) at time t and the total value it would have attained if information at the desired quality at time t were to be received:

$$r(q_d, q_p; t) = \frac{\int_{R_d^t \cap R_p^t} v\big(q_p(\omega, t); q_d\big)\, d\omega}{\int_{R_d^t} v\big(q_d(\omega, t); q_d\big)\, d\omega} \tag{1}$$

Note that if no extra value is gained when information of higher quality is provided than what is desired, then $v\big(q_p(\omega, t); q_d\big) \leq v\big(q_d(\omega, t); q_d\big)$ and, hence, $0 \leq r(q_d, q_p; t) \leq 1$. For the numerical results later in the paper, we will use the simple VoI function $v\big(q_p(w); q_d\big) = \min\big\{q_p(w), q_d(w)\big\}$, which easily satisfies the previous condition.

Finally, we define a measure of the *spatiotemporal relevancy* $r_{st}(q_d, q_p)$ of a provider the average of the spatiotemporal relevancy of the provider over the entire time horizon of interest $\mathcal{T} = \{t : \exists \omega \text{ s.t. } (\omega, t) \in R_d\}$. We write $|\mathcal{T}|$ for the "duration" of $\mathcal{T}$ and

$$r_{st}(q_d, q_p) = \frac{1}{|\mathcal{T}|} \int_{\mathcal{T}} r\big(q_d, q_p; t\big) dt; \tag{2}$$

with a corresponding expression using sums in case of discrete time. It should be noted that although not evident at first glance, the relevancy defined in (2) could assume the form of an elaborate weighted time average with the weights implicitly incorporated within the VoI functions $v\big(q_p(\omega, t); q_d\big)$.

III. Problem Formulation and DP Algorithm

In this section, we develop a problem model and provide a solution methodology for the problem. In the following sections, we present special cases of the problem model that lead to accelerated but approximate solutions to the problem.

A. System model and assumptions

Let $t_n, n \in \mathcal{N} = \{1, \ldots, N\}$, be a sequence of time instants over the time interval $\mathcal{T} = [0, T]$; we refer to t_n as time n and the time interval $\Delta t_n = [t_n, t_{n+1})$ as slot n. Each n represents a *provider selection point* where the consumer may decide to switch providers to maximize the spatiotemporal relevancy of the information it receives under various operational constraints; we use the discrete version of (2).

The operational constraints are expressed through a resource (or budget) constraint B, which could represent a monetary constraint, an energy constraint, amount of risk, and so on. There is a set $\mathcal{K} = \{1, \ldots, K\}$ of providers that can be engaged at one time or another during the interval $\mathcal{T}$, and the cost of engaging with provider k over slot n is $c_{k,n}$. It is assumed that providers communicate their capabilities to the consumer through their respective QoI functions $q_p^k(\cdot)$; how exactly this is done is beyond the scope of this paper. The consumer combines this with its own desires, expressed by its QoI function $q_d(\cdot)$ to compute a provider's spatiotemporal relativity as described in the previous section.

Finally, we assume that, whenever it engages with two or more providers serving the same spatiotemporal point, a consumer "experiences" an aggregate QoI function that results from the QoI functions of the providers engaged. Specifically, if w is a spatiotemporal point within the volumes R_p^i of two (or more) providers $i \in \mathcal{K}$, and $\boldsymbol{x}_w$ is the provider *selection mask* indicating which providers the consumer is using at point w,[3] then the aggregate provider QoI function $q_p^{\boldsymbol{x}}(w)$ is described by the transformation $h\big(q_p^k(w); k \in \mathcal{K}, x(k) = 1\big)$. For example, for two providers i and j, if the accuracy of measurements from provider i at point w is 97% and from provider j is 95%, the aggregated quality from the two providers could be taken to be the better accuracy of the two, i.e., 97%, i.e., $h \equiv \max\{\cdot\}$. We use the latter exemplary h in our numerical results later on.

B. Problem formulation

Let $\mathbf{I} = [I(k,n)]_{K \times N}$ be the provider selection matrix, where $I(k,n)$ equals 1 when provider k is engaged during interval Δt_n, and 0 otherwise. Also, let $q_p^{\mathbf{I}}(\cdot)$ be the spatiotemporal relevancy of a "super-provider" with a QoI function aggregated from the providers indicated by selection matrix $\mathbf{I}$ as described earlier. The problem at hand is, then, described by the following optimization formulation:

Problem $\mathbf{\Pi}^{\mathcal{T}}$: Find the provider selection matrix $\mathbf{I}$ that maximizes $r_{st}\big(q_d, q_p^{\mathbf{I}}\big)$, such that:

$$\sum_{n=1}^{N} \sum_{k=1}^{K} I(k,n) c_{k,n} \leq B. \tag{3}$$

Problem $\mathbf{\Pi}^{\mathcal{T}}$ is a generalization of the Knapsack problem in two ways. First, the fact that the benefit of selecting a particular provider depends on the providers that are already selected, and, then, the fact that there is the temporal dimension of the problem. The 0-1 Knapsack problem is an NP-Hard problem [9] and, hence, problem $\mathbf{\Pi}^{\mathcal{T}}$ is also NP-Hard, which means that there is no known algorithm that calculates the optimal solution in polynomial time. The most efficient algorithm to solve the 0-1 Knapsack Problem is a dynamic programming algorithm that manages to find the optimal solution in *pseudo-polynomial* time by splitting the problem into smaller subproblems and storing intermediate results in memory.

[3] $x_w(k) = 1$ if provider $k \in \mathcal{K}$ is used, and 0 otherwise. For convenience, if context permits it, we will drop the index w from $\boldsymbol{x}_w$.

Algorithm 1 – First Level DP Algorithm for time n

1: **for** $k = 1$ to K **do**
2:   **for** $b = 0$ to B **do**
3:     **if** $c_{k,n} \leq b$ **then**
4:       $\boldsymbol{x} = \boldsymbol{x}_{k-1,n}^{b-c_{k,n}}$; where: $\boldsymbol{x}_{0,n}^{b-c_{k,n}} \stackrel{\text{def}}{=} \mathbf{0}$, $\boldsymbol{x}_{k-1,n}^{0} \stackrel{\text{def}}{=} \mathbf{0}$;
5:       $x(k) = 1$;
6:       **if** $r_s\big(q_d, q_p^{\boldsymbol{x}}, t_n\big)$ not calculated **then**
7:         Calculate $r_s\big(q_d, q_p^{\boldsymbol{x}}, n\big)$ using (1);
8:       **else**
9:         Get $r_s\big(q_d, q_p^{\boldsymbol{x}}, n\big)$ from memory;
10:       **end if**
11:       **if** $r_s\big(q_d, q_p^{\boldsymbol{x}}, t_n\big) > V^n[k-1, b]$ **then**
12:         $V^n[k, b] = r_s\big(q_d, q_p^{\boldsymbol{x}}, n\big)$; $\boldsymbol{x}_{k,n}^{b} = \boldsymbol{x}$;
13:       **else**
14:         $V^n[k, b] = V^n[k-1, b]$; $\boldsymbol{x}_{k,n}^{b} = \boldsymbol{x}_{k-1,n}^{b}$;
15:       **end if**
16:     **else**
17:       $V^n[k, b] = V^n[k-1, b]$; $\boldsymbol{x}_{k,n}^{b} = \boldsymbol{x}_{k-1,n}^{b}$;
18:     **end if**
19:   **end for**
20: **end for**

C. The solution

To solve $\mathbf{\Pi}^{\mathcal{T}}$ we have developed a two-layer dynamic programming algorithm. Initially, Algorithm 1 looks only at the optimization process one time instant at a time. It is based on the algorithm for the static case introduced in [1] updated to the index structure in this paper. Specifically, $q_p^{\boldsymbol{x}}$ in line 6 is the aggregated QoI function representing the collective behavior of the providers selected as indicated by the selection mask $\boldsymbol{x}$ and defined in the provider spatiotemporal region $R_p^{\boldsymbol{x}} = \cup_{k \in \mathcal{K} | x(k)=1} R_p^k$. The vector $\boldsymbol{x}_{k,n}^{b}$ represents the optimal selection mask when the total time horizon is n, the total available budget is b and there are in total k providers; hence, $\boldsymbol{x}_{K,N}^{B}$ corresponds to the desired selection mask for the problem at hand.

At first, Algorithm 1 is run independently for each time instant n, with $n = 1, \ldots, N$, optimizing the allocation of the total budget B at each time instant. For each n, the algorithm produces the $K \times B$ matrix $\boldsymbol{V}^n$, whose (k, b) entry contains the optimal spatiotemporal relevancy that corresponds to $\boldsymbol{x}_{k,n}^{b}$ selection mask. Hence, the last row considers all the K providers, and the last entry also considers the entire budget B being available at time n.

Subsequently, Algorithm 2 uses the $\boldsymbol{V}^n$ matrices to construct the $N \times B$ matrix $\boldsymbol{F}$ whose n-th row is the last row of

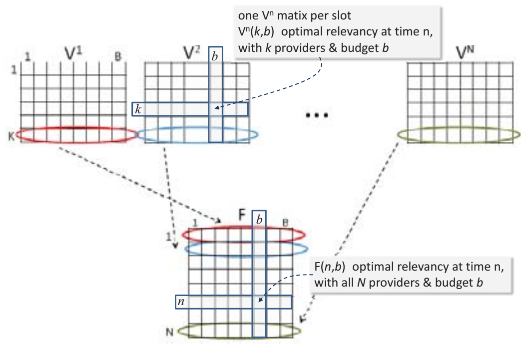

Fig. 1. Matrix merging at the second level DP Algorithm

matrix $\boldsymbol{V}^n$, see Figure 1. The resulting matrix $\boldsymbol{F}$ is used to construct the final matrix $\boldsymbol{S}$, necessary for the calculation of the optimal selection masks of the overall problem. The main steps of the creation of matrix $\boldsymbol{S}$ are lines 7–10 in Algorithm 2. At each iteration n, we choose the allocation of budget β^{max} (and subsequently the allocation of $b - \beta^{max}$ to all previous $n-1$ slots) that gives higher aggregate relevancy and store a reference towards the optimal solution of the previous time instants (the element $S(n-1, b-\beta^{max})$), denoted by $Prev(n, b)$. Finally, element $S(N, B)$ will have the optimal aggregate relevancy for the overall time period $\mathcal{T}$ and by following the values of matrix $\boldsymbol{Prev}$ it is possible to trace the optimal selection masks to achieve the maximum aggregate relevancy. For instance, $Prev(N, B)$ will contain the budget that should be allocated for the previous $n-1$ time instants and $S\big(N-1, Prev(N, B)\big)$ will be the maximum aggregate relevancy for instants 1 to $N-1$. Then, the optimal selection mask at time slot $N-1$ will be $\boldsymbol{I}_{N-1}^{Prev(N,B)}$, and so on. Note that the optimal selection matrix $\boldsymbol{I}$ will consist of the N optimal selection masks for times $n \in \mathcal{N}$.

The time complexity and memory requirements of Algorithm 2 can be calculated based on the complexity of Algorithm 1 and the additional matrix operations of the second level of the optimization algorithm. Algorithm 1 has a worst case complexity of $O(K^2B)$ [1] and is repeated once for each time instant n. Then, lines 4–12 of Algorithm 2 are independent of the number of providers and only depend on the total budget B and the number of time instants N. Hence the worst case complexity of Algorithm 2 is $O(K^2BN)$. Then, the memory requirements of the algorithm can be calculated as follows. The N runs of Algorithm 1 need to store all V^n matrices, with $n = 1, 2, \ldots, N$, of total size $N \times K \times B$, and the optimal selection masks $[\boldsymbol{x}_{K,n}^b]_{N \times B \times K}$. Then, the additional matrix manipulations of Algorithm 2 need to store matrices $[\boldsymbol{F}]_{N \times B}$ and $[\boldsymbol{S}]_{N \times B}$, the optimal selection mask of size K for each of their cells, and matrix $[\boldsymbol{Prev}]_{N \times B}$. However, the execution time of Algorithm 1, and therefore Algorithm 2, can be accelerated by calculating the *greatest common divisor* (gcd) of all the provider costs $c_{n,k}$ and the available budget B, and then running the algorithm in the

Algorithm 2 – Second level DP Algorithm

1: **for** $n = 1$ to N **do**
2: Run Algorithm 1 for $q_d(\cdot, n)$ and $q_p^k(\cdot, n)$ $\forall$ providers $k = 1, \ldots, K$;
3: **end for**
4: Create matrix F, where $F[n, b] = V^n[K, b]$ (see Figure 1), and keep vectors $\boldsymbol{x}_{K,n}^b$ for $n = 1, \ldots, N$ and $b = 1, \ldots, B$;
5: **for** $n = 2$ to N **do**
6: **for** $b = 1$ to B **do**
7: $\mathcal{B} = \{1, \ldots, b\}$;
8: $S(n, b) = \max_{\beta \in \mathcal{B}} \big(S(n-1, b-\beta) + F(n, \beta)\big)$; where $S(1, b) = F(1, b), \forall b = 1, \ldots, B$, and $S(n, 0) = 0, \forall n = 1, \ldots, N$;
9: $\beta^{max} = \arg\max_{\beta \in \mathcal{B}} \big(S(n-1, b-\beta) + F(n, \beta)\big)$;
10: $I_n^b = \boldsymbol{x}_{K,n}^{\beta^{max}}$; $Prev(n, b) = b - \beta^{max}$;
11: **end for**
12: **end for**

range $[0, B/gcd]$ with provider costs $c_{n,k}/gcd$.

IV. Accelerating the Provider Selection

Algorithm 2 in the previous section requires the run of Algorithm 1 for N times, one for each time slot and then some further calculations to reach the overall optimal solution over the time horizon $\mathcal{T}$. In some cases, it might be necessary to get fast decisions about the providers that must be selected at expense of optimality. Three faster but suboptimal alternative methods will be described and discussed in this section. Numerical results on their performance and time requirements will be presented later in Section VI.

A. Independent per Slot Optimization

Suppose that the city agency has an overall weekly operating budget B along with a daily expense constraint b_n, such that $\sum b_n = B$ for the week. Clearly, in this case, the budget allocations at the decision points are outside the purview of the general optimization problem $\mathbf{\Pi}^{\mathcal{T}}$. Therefore, we can avoid the extra calculations of Algorithm 2, and instead solve a sequence of independent optimization problems for each time n. In other words, one could formulate problem $\mathbf{\Pi}_{\mathbf{b_n}}^{\mathcal{T}}$ as:

Problem $\mathbf{\Pi}_{\mathbf{b_n}}^{\mathcal{T}}$: Given a collection of budget allocations b_n, $n \in \mathcal{N}$, with $\sum_{n \in \mathcal{N}} b_n = B$, find the provider selection matrix $\mathbf{I}$ that maximizes $r_{st}\big(q_d, q_p^{\mathbf{I}}\big)$, such that:

$$\sum_{k=1}^{K} I(k, n) c_{k,n} \leq b_n, \quad n = 1, \ldots, N. \tag{4}$$

Problem $\mathbf{\Pi}_{\mathbf{b_n}}^{\mathcal{T}}$ can be split into N independent optimization problems that optimize the spatial relevancy independently at each slot based on a fixed amount of budget b_n. In other words, Problem $\mathbf{\Pi}_{\mathbf{b_n}}^{\mathcal{T}}$ consists of N repetitions of Problem $\mathbf{\Pi_{bg}}$ in [1], one for each time instant n. So, compared with Problem $\mathbf{\Pi}_{\mathbf{B}}^{\mathcal{T}}$ this approximation runs only Algorithm 1 to get the optimal selection mask $\boldsymbol{x}_{K,n}^{b_n}$ at each time n (which will become the n^{th} column of matrix $\boldsymbol{I}$) and saves the time and memory requirements of Algorithm 2 at the expense of some calculation error in the overall spatiotemporal relevancy. Therefore, the time and memory requirements of Problem $\mathbf{\Pi}_{\mathbf{b_n}}^{\mathcal{T}}$ are the same as for Algorithm 1. The acceleration techniques mentioned in the discussion of Algorithm 2 and in [1] can be also used here to reduce the time requirements of the method.

There could be numerous splitting policies for the budget which may be based on actual needs, accounting conveniences, and so on. These policies could depend on n, or not, they could depend on other b_n's (i.e., have "memory"), or not, etc. In the simplest case, the budget B could be split uniformly across the time horizon, and hence $b_n = B/N$ for all n. With such a memoryless budget assignment, any unused portion of the budget at each slot will go wasted. This may not be desirable though and instead one may want to rollover the unused budget to subsequent slots, e.g., at the next slot or along all remaining slots, to reduce the unused portion of the budget. As an example, at time n let the provider selection mask decided be $\mathbf{I}_n = \{I(1,n), \ldots, I(K,n)\}$. Then the budget slack due to this selection will be $b'_n = b_n - \sum_{k=1}^{K} I(k,n)c_{k,n}$. If this slack is to be distributed equally across the remaining slots and taken advantage when provider selection decisions are made at time $(n+1)$, then:

$$b_{n+1} \stackrel{def}{=} b_{n+1}^{new} = b_{n+1}^{old} + \frac{b'_n}{N-n}, \tag{5}$$

where the designations "old" and "new" apply to the old budget available for decision instant $(n+1)$ (from budget slack assignments made prior to time n) and the updated budget at this time instant.

B. Boolean Relaxation

Problem $\mathbf{\Pi}^{\mathcal{T}}$ is an integer programming problem that is *NP-Hard* as a generalization of the Knapsack problem. However, the problem can be easily solved once the requirement for deriving binary (0-1) values for the selection matrix is relaxed. *Boolean relaxation* [10] is a common technique in optimization problems that contain optimization variables that have boolean values. For an optimization variable x by relaxing the constraint from $x \in \{0,1\}$ to $x \in [0,1]$, we convert the problem into a continuous optimization problem that is in general easier to solve, particularly, if the resulting relaxed problem is convex. In this case, we also relax the concept of provider selection, and instead refer to provider *participation* (to be further explained shortly), and the new optimization problem $\mathbf{\Pi}_{\mathbf{Rel}}^{\mathcal{T}}$ becomes:

Problem $\mathbf{\Pi}_{\mathbf{Rel}}^{\mathcal{T}}$: Find the provider participation matrix $\mathbf{I}$ that maximizes $r_{st}\big(q_d, q_p^{\mathbf{I}}\big)$, such that:

$$(1)\ \sum_{n=1}^{N}\sum_{k=1}^{K} I(k,n)c_{k,n} \leq B;\ \text{ and} \tag{6}$$

$$(2)\ 0 \leq I(k,n) \leq 1,\ \ k \in \mathcal{K} \text{ and } n \in \mathcal{N}$$

Algorithm 3 – Myopic Algorithm

Initialize $I(k,n) = 0$ for $k = 1, \ldots, K$ and $n = 1, \ldots, N$
Calculate $r_{st}\big(q_d, q_p^k\big)$, $k = 1, \ldots, K$, and sort in descending order;
Let $idx(m)$ be the index of ordered provider m.
$b = B$; $m = 1$;
while $(b > 0$ & $m \leq K)$ **do**
  if $b \geq c_{idx(m)}$ **then**
    $I(idx(m), n) = 1$ for $n = 1, \ldots, N$;
  end if
  $b = b - c_{idx(m)}$; $m = m + 1$;
end while

Problem $\mathbf{\Pi}_{\mathbf{Rel}}^{\mathcal{T}}$ is a convex continuous optimization problem since the constraints are linear functions of the optimization variables $I(k,n)$, for $k \in \mathcal{K}$ and $n \in \mathcal{N}$, and the objective function is concave as a summation of linear functions of the optimization variables. Therefore, Problem $\mathbf{\Pi}_{\mathbf{Rel}}^{\mathcal{T}}$ and can be easily solved with gradient based optimization algorithms that require linear time [11].

In the case of Boolean relaxation, there is an appealing interpretation of the derived solution, particularly, when the slots n are of equal duration. An optimal variable $I(k,n) = \alpha$ with $\alpha \in [0,1]$ implies that the consumer decides to bind with (and pay) provider k at time n only for the fraction α of time slot n. This is why we refer to the selection matrix as the participation matrix instead in this case.

C. Myopic Algorithm

In circumstances that a fast decision about selecting an appropriate set of providers is of highest importance, we may resort to myopic algorithms that select providers based on a rather limited view of the problem. We have considered a simple myopic algorithm where the spatiotemporal relevancy $r_{st}\big(q_d, q_p^k\big)$ of each provider k over the time horizon $\mathcal{T}$ is calculated first. Then, the providers are ordered according to their respective relevancy $r_{st}\big(q_d, q_p^k\big)$. Based on this relevancy and the total cost $c^k = \sum_{n=1}^{N} c_{k,n}$ for engaging (whenever possible) with provider k during the period $\mathcal{T}$, the top M providers are chosen so that the total cost is as close to B as possible. In case that $\sum_{m=1}^{M} c^m < B$ and $\sum_{m=1}^{M+1} c^m > B$, it is possible to choose a provider that is further down in the relevancy ranking table but with cost low enough to be accommodated by the remaining budget. Algorithm 3 summarizes the Myopic algorithm. Regarding its complexity, the calculation of the spatiotemporal relevancy of providers requires $O(K)$ time, the ordering of the providers can be calculated in $O(K \log K)$ time and the selection of the providers is $O(KB)$ in the worst case.

This myopic algorithm will operate well when the providers experience inconsequential overlapping in which case the effects of provider aggregation do not manifest themselves strongly. This is because in this case, the increase of the *aggregate* spatiotemporal relevancy by including a provider will be equal to its individual spatiotemporal relevancy. However, in general, the more the providers overlap, the more discrepancy would be in the computed relevancy with regard to the optimal one which could effect the provider selections.

V. Accelerating the Relevancy Assessment

The implementation of DP Algorithm 2 and the three approximation methods presented in Section IV involve multiple evaluations of equations (1) and (2) and therefore multiple numerical integrations of the VoI function. On the other hand, even though significantly faster, the computation of relevancy by merely computing the regions of overlap (e.g., using [12], [13]) as was implied in [14] cannot capture the variability of QoI functions. To bridge that gap between these two extreme cases in spatiotemporal relativity calculation, this section presents two alternative approximation methods that leverage existing polygon intersection tools in such a way that they take into account all the information contained in the QoI functions and lead to significant improvements by avoiding the numerical integration of equations (1) and (2).

A. Polygon Intersection (PolIn)

The most time consuming operation in Algorithm 1 is the calculation of spatial relevancy for the candidate selection masks in line 7. In order to accelerate the calculation of the selection mask at time n, we approximate the computation of the aggregate relevancy $r\big(q_d, q_p^{\boldsymbol{x}}; n\big)$ by splitting the support regions R_p^k and R_d of each provider $k \in \mathcal{K}$ and the consumer, respectively, in a number of polygons and associate a single QoI value with each one. Then, we calculate the aggregate spatial relevancy using polygon operations.

More specifically, let the coverage region $R_p^{k,n}$ of provider k at slot n be partitioned into $L_{n,k}$ polygons[4] $m_{n,k}^l$, with $l = 1, 2, \ldots, L_{n,k}$, and a single QoI value $q_{n,k}^l$ be associated with each polygon to represent the QoI function values within it. In the same way, let the desired support region R_d^n be partitioned into L_n^d polygons and a QoI value $q_{n,d}^l$ be associated with polygon $m_{n,d}^l$, with $l = 1, 2, \ldots, L_n^d$. The QoI values $q_{n,k}^l$ for each provider polygon and $q_{n,d}^l$ for each desired support polygon could represent the mean of the actual QoI values within those polygons. Then, the aggregation of the polygons based on the candidate selection mask $\boldsymbol{x}$ at each iteration of Algorithm 1 is conducted by calculating the intersections and subtractions for all the combinations of the selected polygons. The result of the aggregation operation will be the set of polygons $m_{n,\boldsymbol{x}}^l$, where $l = 1, 2, \ldots, L_{n,\boldsymbol{x}}$. Each one of these polygons will be associated with a QoI value $q_{n,\boldsymbol{x}}^l$ which will be the result of the QoI provider aggregation operator h of all the providers involved in the creation of polygon $m_{n,\boldsymbol{x}}^l$; as mentioned earlier, for the numerical results h will be the "$\max$" operator. These aggregate polygons of the selected providers are then intersected with the desired polygons (i.e., derived from R_d) and the result is a set of polygons $m_{n,d\cap\boldsymbol{x}}^s$, with $s = 1, 2, \ldots, S_{n,d\cap\boldsymbol{x}}$ along with their corresponding QoI values $q_{\tau,d\cap\boldsymbol{x}}^s$. The QoI values are calculated by applying the VoI function v at the desired $q_{n,d}^l$ and aggregate $q_{n,\boldsymbol{x}}^l$ QoI values of the provider and desired polygons involved in the creation of polygon $m_{n,d\cap\boldsymbol{x}}^s$; $d \cap \boldsymbol{x}$ identifies the regions of overlap between the desired polygons and the polygons of providers in $\boldsymbol{x}$. As mentioned earlier, for the numerical results, v will be represented by the "$\min$" operator. Figure 2 illustrates the described procedure for $K = 2$, $L_{n,k} = 2$, $k \in \{1, 2\}$, and $L_n^d = 2$. Observe that the aggregation operation of the 4 provider polygons results in the creation of $L_{n,\boldsymbol{x}} = 8$ polygons, whereas the aggregate provider and desired polygons' intersection leads to the creation of $S_{n,d\cap\boldsymbol{x}} = 7$ polygons.

[4] We have implicitly assumed that any region $R_p^{k,n}$ or R_d^n can be approximated sufficiently well by a set of polygons of arbitrary number of edges, a widely used technique in computer graphics called *Polygonal Modeling* [15].

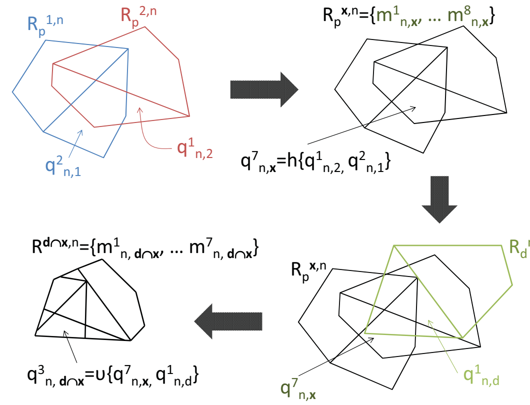

Fig. 2. Example of the PolIn method

Finally, the spatial relevancy $r\big(q_d, q_p^{\boldsymbol{x}}; n\big)$ will be calculated according to equation:

$$r_s\big(q_d, q_p^{\boldsymbol{x}}, n\big) = \frac{\sum_{s=1}^{S_{n,d\cap\boldsymbol{x}}} q_{n,d\cap\boldsymbol{x}}^s \cdot A_{n,d\cap\boldsymbol{x}}^s}{\sum_{l=1}^{L_n^d} q_{n,d}^l \cdot A_{n,d}^l} \tag{7}$$

where $A_{n,d\cap\boldsymbol{x}}^s$ is the area of the provider polygon $m_{n,d\cap\boldsymbol{x}}^s$ for $s = 1, 2, \ldots, S_{n,\boldsymbol{x}}$, and $A_{n,d}^l$ is the area of the desired polygons $m_{n,d}^l$, for $l = 1, 2, \ldots, L_n^d$.

Apart from this approximation of the spatial relevancy metric the rest of the procedure for solving Problem $\boldsymbol{\Pi}^{\mathcal{T}}$ using algorithms 1 and 2 remains identical.

B. Mean Value Polygon Intersection (mvPolIn)

The PolIn method can lead to the creation of a very large number of polygons. The calculation of the aggregation and intersection operations for all these polygons could still consume a significant amount of time. Therefore, to accelerate the process even further, we also consider the following variation, which we refer to as the Mean Value Polygon Intersection (mvPolIn).

The main idea behind mvPolIn is that the spatial relevancy calculation process can be accelerated significantly if the aggregation of providers leads to a relatively small number of polygons with a single QoI value. Therefore, let $m_{n,\boldsymbol{x}}^j$, with $j = 1, 2, \ldots, J_n$ be the polygons resulting from the union of

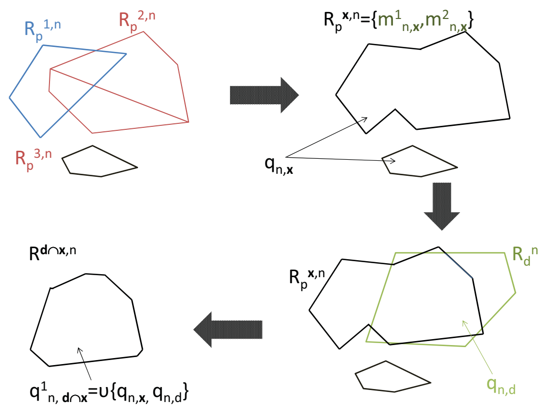

Fig. 3. Example of the mvPolIn method

all providers in the selection mask $\boldsymbol{x}^5$ and there is a single QoI value $q_{n,\boldsymbol{x}}$ associated with these polygons $m^j_{n,\boldsymbol{x}}$, calculated as the weighted average of the QoI values of all the polygons participating in their creation, according to:

$$q_{n,\boldsymbol{x}} = \frac{\sum_{j=1}^{J_n} A^j_{n,\boldsymbol{x}} \cdot m^j_{n,\boldsymbol{x}}}{\sum_{j=1}^{J_n} A^j_{n,\boldsymbol{x}}} \tag{8}$$

where $A^j_{n,\boldsymbol{x}}$ is the area of polygon $m^j_{n,\boldsymbol{x}}$. Then, to calculate the spatial relevancy of the candidate selection mask $\boldsymbol{x}$, the J_n provider polygons are intersected with the L^d_n desired polygons whose QoI value $q_{n,d}$ is also a weighted average calculated in a way similar to (8). The resulting polygons $m^s_{n,d\cap\boldsymbol{x}}$ for $s = 1, 2, \ldots, S_{n,d\cap\boldsymbol{x}}$ along with their respective QoI values $q^s_{n,d\cap\boldsymbol{x}} = v\big(q_{n,\boldsymbol{x}}, q_{n,d}\big)$ are used to calculate the spatial relevancy of the candidate selection mask $\boldsymbol{x}$ using:

$$r\big(q_d, q_p^{\boldsymbol{x}}; n\big) = \frac{\sum_{s=1}^{S_{n,d\cap\boldsymbol{x}}} q^s_{n,d\cap\boldsymbol{x}} \cdot A^s_{n,d\cap\boldsymbol{x}}}{\sum_{l=1}^{L^d_n} q^l_{n,d} \cdot A^l_{n,d}} \tag{9}$$

where $A^s_{n,d\cap\boldsymbol{x}}$ is the area of polygon $m^s_{n,d\cap\boldsymbol{x}}$. Figure 3 shows an example of the above described procedure for $K = 3$, $L_{n,k} = 1$, $k \in \{1, 3\}$ and $L_{n,2} = 2$. The spatial relevancy value calculated by (9) is then used at the next steps of Algorithm 1 to determine the optimal selection mask at time n.

VI. Simulation Results

The DP Algorithm 2 and the other alternative methods presented in the previous sections were simulated in a MATLAB environment to study their performance vs. time-complexity trade-offs.

The various methods were evaluated in multiple randomly generated experiments for $N = 8$ time instants. At every instant of these experiments a number of Gaussian mixtures, representing the desired (q_d) and provided (q_p^k) QoI functions, where randomly scaled and placed on the plane. The QoI shapes for all N instants where then fed to the various optimization methods where each of them determined the most appropriate set of providers according to the budget constraints, while measuring the necessary optimization time of each method. The above experiments were repeated multiple times and for values of K within the range $[4, 11]$; budgets $B = 75$, 85, 98, 110, 115, 130, 138, 150; and provider costs: 2, 3, 4, 6, 3, 4, 2, 5, 4, 2, 1, 4 (fixed per provider). Figure 4 shows an example time instant where the "city agency" has to determine the spatiotemporal relevancy of $K = 8$ providers.

Figure 5 shows the spatiotemporal relevancy value of each method with respect to the number of providers K. As expected, Algorithm 2 and the Boolean Relaxation method lead to almost identical values of spatiotemporal relevancy, which is also higher than all other mothods. Then, the PolIn method results to values that are relatively close to the optimal (as calculated by Algorithm 2). More specifically, the spatiotemporal relavancy metric value calculated by PolIn was on average 5% below the optimal solution. On the other hand, the performance of mvPolIn and the Myopic methods were significantly suboptimal, with the former being on average 20% suboptimal and the latter achieving around 26% worse performance. This high percentage of error is observed due to the several approximations each of the methods makes to accelerate their execution (see discussion of Figure 6).

However, the accuracy of spatiotemporal relevancy calculations are only one side of the coin. The execution time of each of the methods is also important to determine their efficiency. Figure 6 shows the execution times of the 6 methods. According to this, Algorithm 2 consumes the largest amount of time. This is a result of the numerical integrations necessary to calculate the spatial relevancy of every candidate selection mask at each time instant. The Boolean Relaxation method converges to the optimal solution in relatively short time as it solves a continuous optimization problem for which there are efficient linear gradient based solution algorithms [11]. The Independent per Slot optimization algorithm is faster

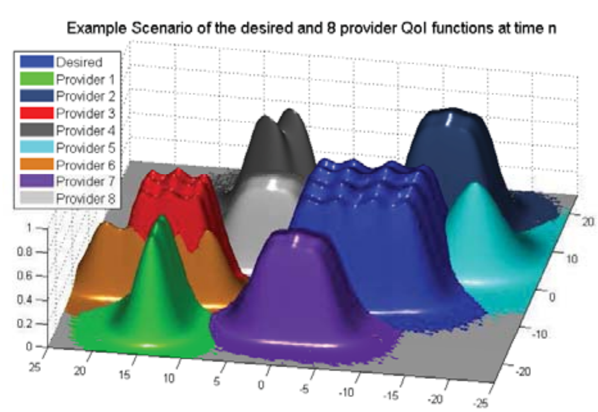

Fig. 4. An example of the QoI functions a consumer (blue) and 8 providers

[5] In general, the union of a set of polygons will never increase its cardinality, hence, $\sum_{k|x(k)=1} L_{n,k} \geq J_n$ will hold true.

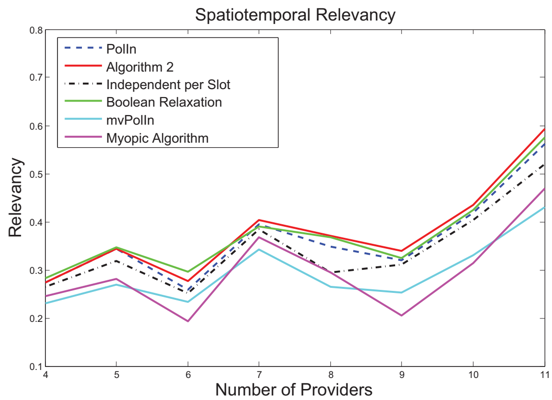

Fig. 5. Aggregate spatiotemporal relevancy of various methods

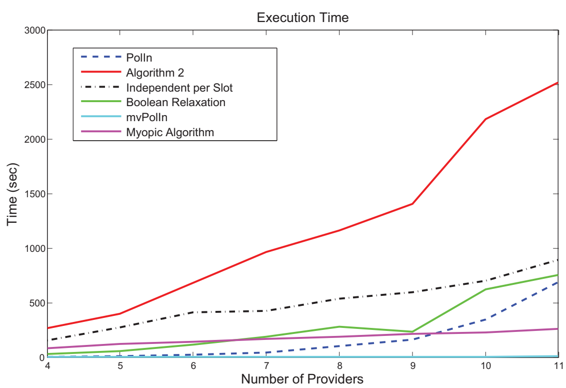

Fig. 6. Execution Time of Algorithm 2 and Approximation Methods using a 2.4 GHz dual core PC with 4 GB of RAM.

than Algorithm 2 since it only executes Algorithm 1 N times. Moreover, it makes more efficient use of parallel processing capabilities of current computer systems and the acceleration method using the *greater common divisor* of the provider cost and the available budget (see Section III). Then, the Myopic algorithm needs almost linear execution time with respect to the number of providers, and, finally, mvPolIn is the fastest method due to the significant simplifications done in the calculation of the spatial relevancy.

It is therefore evident that each algorithm offers a different trade-off between accuracy and time efficiency that might make some algorithms more suitable than others depending on the requirements (time and accuracy) of an application.

VII. Concluding Remarks

Motivated by evolutionary trends in smart planet applications, this paper studied the spatiotemporal relevancy between information providers and an information consumer by considering the temporal evolution of their spatial relationships. This evolution can be the result of either a spatial movement of providers and/or the changing interests of the consumer, or temporal changes in the QoI functions. We have defined the *spatiotemporal relevancy* metric and then formulated an integer programming optimization formulation to describe the problem of selecting the most appropriate set of providers over a period of time and by assuming knowledge of the QoI functions throughout that period. Then, we presented a two-level Dynamic Programming algorithm to solve the problem optimally in pseudo-polynomial time. Consequently, we discussed alternative optimization formulations and relevancy approximating techniques that improve execution time at the expense of accuracy. Finally, the optimal algorithm along with the various approximations were evaluated and compared in MATLAB.

Future work will consider operational aspects of the system, such as QoI advertisements, as well as more generalized motion models, consideration of additional provider aggregation operators $h(\cdot)$ and VoI functions $v(\cdot)$.

References

[1] G. Tychogiorgos and C. Bisdikian, “Selecting relevant sensor providers for meeting “your” quality information needs,” in *IEEE Mobile Data Management*, Lulea, Sweden, June 2011.

[2] J. Wright, C. Gibson, F. Bergamaschi, K. Marcus, R. Pressley, G. Verma, and G. Whipps, “A dynamic infrastructure for interconnecting disparate ISR/ISTAR assets (the ITA sensor fabric),” in *12th Intl Conf. on Information Fusion (FUSION’09)*, Seattle, WA, USA, July 6–9, 2009.

[3] S. Meguerdichian, F. Koushanfar, M. Potkonjak, and M. B. Srivastava, “Coverage problems in wireless ad-hoc sensor networks,” in *in IEEE INFOCOM*, 2001.

[4] A. Ghosh and S. K. Das, “Coverage and connectivity issues in wireless sensor networks: A survey,” *Pervasive and Mobile Comp.*, June 2008.

[5] N. Pelekis, B. Theodoulidis, I. Kopanakis, and Y. Theodoridis, “Literature review of spatio-temporal database models,” *The Knowledge Engineering Review*, September 2004.

[6] J. Hershbergery, N. Shrivastava, and S. Suriz, “Summarizing spatial data streams using clusterhulls,” in *8th Wksp on Algorithm Engineering and Experiments (ALENEX’06)*, Miami, FL, USA, Jan. 2006.

[7] H. Prautzsch, W. Boehm, and M. Paluszny, *Bezier and B-Spline Techniques*. Springer-Verlag, 2002.

[8] C. Bisdikian, L. M. Kaplan, M. B. Srivastava, D. J. Thornley, D. Verma, and R. I. Young, “Building principles for a quality of information specification for sensor information,” in *12th Intl Conf. on Information Fusion (FUSION’09)*, Seattle, WA, USA, July 2009.

[9] S. Martello and P. Toth, *Knapsack Problems: Algorithms and Computer Implementations*. John Wiley & Sons, 1990.

[10] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge University Press, 2004.

[11] D. P. Bertsekas, *Nonlinear Programming*. Athena Scientific, 1999.

[12] R. Bulbul and A. U. Frank, “Intersection of nonconvex polygons using the alternate hierarchical decomposition,” in *Geospatial Thinking*, ser. Lecture Notes in Geoinformation and Cartography, M. Painho, M. Y. Santos, and H. Pundt, Eds. Springer Berlin Heidelberg, 2010.

[13] J. O’Rourke, C.-B. Chien, T. Olson, and D. Naddor, “A new linear algorithm for intersecting convex polygons,” *Computer Graphics and Image Processing*, pp. 384–391, 1982.

[14] C. Bisdikian, J. Branch, K. K. Leung, and R. I. Young, “A letter soup for the quality of information in sensor networks,” in *IEEE IQ2S PerCom Workshop*, Mar. 2009.

[15] M. Russo, *Polygonal Modeling: Basic and Advanced Techniques*. Worldware Publishing Inc., 2006.