

Link-based Ranking of Skyline Result Sets

Akrivi Vlachou^{1 *}

¹ Department of Informatics
Athens University of Economics and Business
Athens, Greece
avlachou@aueb.gr

Michalis Vazirgiannis^{1,2 †}

² GEMO Team
INRIA/FUTURS
Paris, France
mvazirg@aueb.gr

ABSTRACT

Skyline query processing has received considerable attention in the recent past. Mainly, the skyline query is used to find a set of non dominated data points in a multi-dimensional dataset. One of the major drawbacks of the skyline operator is the high cardinality of the result set. Providing the most interesting points of the skyline set (top- k) inherently involves the ranking of the skyline points. In this paper, we propose a method for ranking the skyline points and therefore for answering top- k skyline queries. First, we introduce the notion of the *skyline graph* which relies on the dominance relationship of the skyline points in all possible subspaces of the original data space. Using the aforementioned mapping, we can apply well-known link-based ranking algorithms on the skyline graph. Unlike most previously proposed ranking approaches we do not rely on user defined functions and do not impose arbitrary preferences on some dimensions. An experimental evaluation of the proposed method is presented illustrating the ranking ability of our framework.

1. INTRODUCTION

The skyline operator and its computation have attracted much attention recently. Skyline queries help users make intelligent decisions over complex data, where different and often conflicting criteria are considered. Such queries return a set of interesting data points that are not dominated by any other point on all dimensions [5]. Consider for example a database containing information about hotels. Each tuple of the database is represented as a point in a data space consisting of numerous dimensions. In our example, the y -dimension represent the price of a room, whereas the x -dimension captures the distance of the hotel to a point of interest such as the beach (Figure 1). According to the dominance definition, a hotel dominates another hotels because it is cheaper and closer to the beach.

Skyline analysis applications usually provide numerous candi-

*This research project is co-financed by E.U.-European Social Fund (75%) and the Greek Ministry of Development-GSRT (25%).

†Supported by the Marie Curie Intra-European Fellowship NG-WeMiS: Next Generation Web Mining and Searching (MEIF-CT-2005-011549).

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

M-PREF '07 Vienna, Austria

Copyright 200X ACM X-XXXXX-XX-X/XX/XX ...\$5.00.

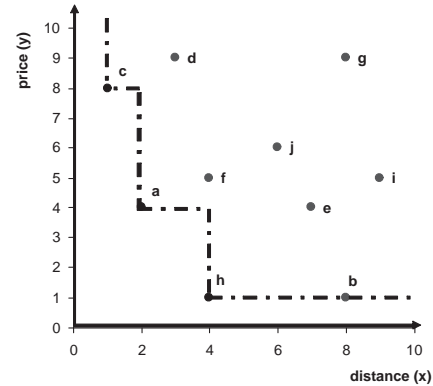


Figure 1: Skyline Example

date attributes, and skyline queries generally refer to different subsets of the attributes, henceforth called subspace of the original space, depending on the users' interests. In our running example, the hotel database could contain many other attributes, such as price, distance, age and number of rooms. Figure 1 depicts the skyline points of a query posed by a user who is sensitive in the price and the distance attributes.

One of the major drawbacks of the skyline operator is the high cardinality of the result set [14, 8]. Especially in the case where the data set is high dimensional or anti-correlated the size of the result set can be huge. The huge size of the skyline result set hinders decision-making by the user, and motivates the need for ranking of skyline points. The user prefers to view the top- k skyline points instead of the whole skyline set in a random or algorithmic specified order containing possibly thousands data points. By definition, the skyline operator returns a set of data points that are equivalent in importance (all skyline points are equally good based on the users preferences). An automated ranking of the skyline points is essential to be able to return the top- k skyline points to the user.

The dominance relationship specifies which data points belong to the skyline set. The importance of the dominance relationship in subspaces was studied in [22] in order to investigate the semantics of skyline points. [22] motivates us to rank the skyline points according to their dominance relationship in all possible non-empty subspaces. In [7] the authors propose to rank the skyline points based on the skyline frequency, which indicates how often they are returned in the skyline when a subspace of attributes is considered. In contrast to their counting approach we focus on the dominance relationships of skyline points in subspaces. We aim to rank the skyline points by investigating the dominance relationships

between pairs of skyline points in subspaces. Therefore, we map the dominance relationship to a graph, called *skyline graph*, which allows us to apply well-known link-based ranking algorithms such as PageRank [6] or HITS [18].

Result ranking is a cornerstone process in web search applications enabling users to effectively retrieve relevant and important information easily. Similar to the problem of skyline ranking, in web search a set of relevant web pages that are retrieved have to be ordered before returned to the user. Link-based algorithms have been proposed to deal with equally relevant web documents. For example PageRank [6] is a well-known algorithm used for ranking web search results and has received significant attention in the related research literature. Furthermore, link-based ranking of relational data was studied in [13]. ObjectRank is presented in [1] that applies authority-based ranking to keyword search in databases modelled as labelled graphs.

In this paper, we propose a novel method for ranking the skyline points of a data set. We map the dominance relationship of the skyline points into a weighted directed graph, called *skyline graph*. More specifically, if a skyline point is dominated by another skyline point in a subspace of the data space an edge is added to the skyline graph. Intuitively, a skyline point that is dominated from another skyline point in some subspace gives some of its importance to the dominating point. Thus, link-based ranking techniques applied to the skyline graph are suitable to rank the skyline points. Finally, we present a bottom-up algorithm that relies on [27] and computes the dominance relationships by sharing skyline results.

To summarize, the key contributions of this paper are:

- We propose a framework for ranking the skyline result set that respects the skyline definition, in the sense that it does not impose arbitrary preferences on certain dimensions, and does not rely on user defined functions.
- We introduce the notion of the skyline graph which explores the dominance relationship of skyline points in subspaces of the original data space.
- An efficient bottom-up algorithm is presented in order to compute the dominance relationships in subspaces which are required for the skyline graph construction.
- In our experimental evaluation on real-life data we use PageRank as the ranking algorithm, and we evaluate the effectiveness of our approach and the ability of the dominance relationship to provide a meaningful ranking.

The rest of this paper is organized as follows: We present related work in Section 2, while in Section 3 we provide a brief overview of the relevant techniques. In Section 4, we introduce the notion of the skyline graph. Our bottom-up algorithm for the skyline graph construction is presented in Section 5. In Section 6, we present the experimental evaluation. Finally, we conclude in Section 7.

2. RELATED WORK

Skyline computation has recently received considerable attention in the database research community. Two algorithms, BNL and DC are proposed in [5], while SFS [10], is based on the same principle as BNL, but improves performance by first sorting the data according to a monotone function. Tan *et al.* [23] propose the first progressive techniques, namely Bitmap and Index method. In [19], an algorithm based on nearest neighbor search on the indexed dataset is presented. Then, Papadias *et al.* [21] propose a branch and bound algorithm to progressively output skyline points on a dataset indexed by an R-Tree, with guaranteed minimum I/O cost.

Recently papers focus on algorithms to support subspace skyline retrieval. In [24] SUBSKY, a non-incremental algorithm, is presented, which transforms the multi-dimensional data to one-dimensional values, and then indexes the dataset with a B-Tree. In [11] the problem of supporting constrained subspace skylines was posed. [25] addresses the computation of subspace skyline queries in large-scale peer-to-peer networks.

Pei *et al.* [22] discussed subspace skylines primarily from the view of the query semantics. They solved the skyline membership query, why and in which subspaces an object belongs to the subspace skyline, by using the notion of skyline group. The authors in [27] present a pre-processing approach, called SKYCUBE, which is defined as the union of all skyline points of all possible non-empty subspaces. For this purpose, they explore sharing strategies for answering multiple skyline queries by identifying computational dependencies among skyline queries. Recently, Xia and Zhang [26] address the issue of supporting updates in SKYCUBE.

Ranking of skyline points was discussed in [7]. The authors introduce a new metric called skyline frequency, to compare and rank the interestingness of data points based on how often they are returned in the skyline when different subspaces are considered. To avoid the $2^d - 1$ skyline computations that are required, they propose an approximate algorithm for estimating the skyline frequency. Our approach differs to their approach because we do not focus on the number of subspaces but on the dominance relationship between two skyline points in all possible subspaces. Intuitively, a skyline point that is dominated from another skyline point in some subspace gives some of its importance to the dominating point.

Our work differs to the traditional top-k query [12, 9] that requires the user to provide a preference function over all dimensions. Moreover we do not rely on user defined functions and do not impose arbitrary preferences on some dimensions [16, 2]. In [16], the authors present the Telescope algorithm that ranks the skyline points by user specified preferences on the available dimensions. In [2] a ranking approach based on user defined regions that dominate all other regions is proposed.

Ranking of query results in a web search-engine is an important problem and has attracted significant attention in the web research community. Link-based ranking techniques like PageRank [6] or HITS [18] assess the importance of web pages based on the Web's structure. These two approaches have been extended [17, 15, 20] and their properties have been studied intensively [3, 4]. Furthermore, link-based ranking of relational data was studied in [13].

3. PRELIMINARIES

In this section we provide a brief overview of the relevant techniques to our approach, namely skyline queries and the PageRank algorithm. For a complete reference to the symbols used in this paper see Table 1.

3.1 Skyline Queries

Given a data space D defined by a set of n dimensions $\{d_1, \dots, d_n\}$ and a data set S on D , a point $p \in S$ can be represented as $p = \{p_1, \dots, p_n\}$ where every p_i is a value on dimension d_i .

Definition 1 (Skyline): A point $p \in S$ is said to *dominate* another point $q \in S$ if (1) on every dimension $d_i \in D$, $p_i \leq q_i$; and (2) on at least one dimension $d_j \in D$, $p_j < q_j$ denoted as $p \prec q$. The *skyline* is a set of points $SKY \subseteq S$ which are not dominated by any other point. The points in SKY are called skyline points.

The notion of skyline can be extended to subspaces. Each non-empty subset U of D ($U \subseteq D$) is referred to as a *subspace* of D .

Notation	Description
S	Dataset
n	Cardinality of S
D	Data space of S
U, V	Subspace of D ($U, V \subseteq D$)
d	Data dimensionality
d_i	One dimension ($1 \leq i \leq d$)
p, q	Data points
p_i	Value of p on dimension d_i
$p \prec q$	p dominates q
$p \prec_U q$	p dominates q on subspace U
SKY	Set of the skyline points of D
SKY_U	Skyline of subspace U
G_{SKY}	Skyline graph
V_{SKY}	Vertices of G_{SKY}
E_{SKY}	Edges of G_{SKY}
w_{SKY}	Weight function of G_{SKY}

Table 1: Overview of notation.

Definition 2 (Subspace Skyline): A point $p \in S$ is said to *dominate* another point $q \in S$ on subspace $U \subseteq D$ if (1) on every dimension $d_i \in U$, $p_i \leq q_i$; and (2) on at least one dimension $d_j \in U$, $p_j < q_j$ denoted as $p \prec_U q$. The *skyline* of a subspace $U \subseteq D$ is a set of points $SKY_U \subseteq S$ which are not dominated by any other point on subspace U .

Consider for example the dataset depicted in Figure 1. The skyline points are $SKY = \{c, a, h\}$, while for the (non-empty) subspace $U = \{y\}$ the skyline points on U are $SKY_U = \{h, b\}$. Notice that the point b is a skyline point on the subspace $\{y\}$ but it is dominated by the point h in the space $\{x, y\}$. Generally, a skyline point $p \in SKY_U$ on a subspace $U \subseteq D$ is either a skyline point on D , or is dominated on D by another skyline point $q \in SKY_U$ on a subspace U for which $p_i = q_i, \forall i : d_i \in U$.

In [5] the authors extended SQL’s SELECT statement by an optional SKYLINE OF, such that the users can specify the dimensions as well as the function (MIN, MAX, DIFF) used for the skyline query. For example, the query corresponding to Figure 1 is expressed in SQL as: SELECT * FROM Hotels SKYLINE OF distance MIN, price MIN. Without loss of generality, we assume that all dimensions are evaluated according to the min function.

[27] further extends SQL by introducing the SKYCUBE BY keyword defined as the set of all possible skyline results for any non-empty subspace of the data space D . For example, for the hotel dataset shown in Figure 1 the SKYCUBE query can be written as: SELECT * FROM Hotels SKYCUBE BY distance MIN, price MIN. The benefit of SKYCUBE is that the skyline points of different subspaces can be computed in a coincide and semantic-clear query which enables sharing strategies among different skyline queries.

3.2 PageRank Overview

Brin and Page [6] proposed a link analysis algorithm, called PageRank, in order to rank web pages, which is based on the random surfer model. The algorithm models the behavior of a random surfer, which either chooses an outgoing link of the current page, or jumps to a random page from the entire Web.

Consider the Web as a directed graph $G(V, E)$, where the vertices V represent the web pages and the edges E the links between them. A surfer on a given page $p \in V$ with probability $1 - \epsilon$

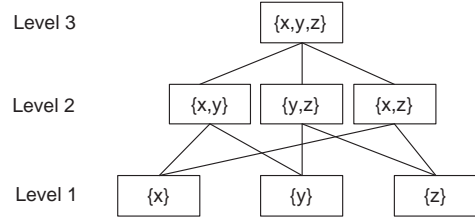


Figure 2: Lattice structure

chooses to select uniformly one of its outlinks $Out(p)$ and with probability ϵ to jump to a random page from the entire Web (ϵ is a small real value, usually 0.15). The PageRank score for page p is defined as the stationary probability of finding the random surfer at page p :

$$PR(p) = (1 - \epsilon) \cdot \sum_{p_i \rightarrow p} \frac{PR(p_i)}{Out(p_i)} + \epsilon \cdot \frac{1}{|V|} \quad (1)$$

The definition of PageRank is recursive and must be iteratively evaluated until convergence. This probability is correlated with the importance of the web page, as it is defined based on the number and the importance of the pages pointing to it. Intuitively, the basic idea of PageRank is that if page p has a link to page p' , then the author of p is implicitly transferring some importance to page p' .

4. SKYLINE GRAPH

The dominance relationship specifies which data points belong to the skyline set. The importance of the dominance relationship in the subspaces was studied in [22] in order to investigate the semantics of the skyline points. In more detail, the authors try to answer the questions why and in which subspaces an object belongs to the skyline, by defining the subspace that decides that an object belongs to the skyline.

A skyline point $p \in SKY$ on the data space D either belongs to the skyline result set on subspace $U \subset D$ or is dominated by at least one skyline point $q \in SKY$ of the data space D . In order to define the skyline graph we focus on subspace dominance relationships of the skyline points $p \in SKY$, and we map them into a graph. Each skyline point of the space D is represented as a vertex v in the skyline graph. If a point q dominates a point p in a subspace U then we add a edge e_{pq} in the skyline graph.

Over a set S of d -dimensional points (on space D) there are $2^d - 1$ possible skyline queries on different subspaces. The different subspaces can be visualized in a lattice structure, see for example Figure 2 where all possible non-empty subspaces of a three dimensional space are shown. In addition the lattice structure shows which subspaces share common dimensions. From the bottom to the top of the lattice, we number each level of the subspaces increasingly. For two subspaces U and V , if V is a subset of U ($V \subset U$) and the level of U is equal to the level of subspace V increased by one, then we call the subspace U a parent subspace of V , i.e. $U = parent(V)$. Analogously, we call subspace V a child subspace of subspace U . Consider the lattice structure of the subspaces shown in Figure 2. The subspace $\{x, y\}$ is a parent of $\{x\}$, while the subspace $\{x, y, z\}$ is not a parent of $\{x\}$.

Definition 3 (Skyline Graph G_{SKY}): Given a d -dimensional dataset S on space D , the skyline graph is the weighted directed graph $G_{SKY} = \{V_{SKY}, E_{SKY}, w_{SKY}\}$ where:

- V_{SKY} is the set of vertices. Each vertex corresponds to a

point	x	y	z
a	10	7	10
b	2	18	28
c	9	5	41
d	49	15	3
e	25	3	52
f	5	16	58
g	23	70	7
h	0	6	79
i	34	73	6
j	89	1	5
k	54	1	82
l	72	90	2

Table 2: Skyline points of 3d dataset

skyline point $p \in SKY$. Thus, $V_{SKY} = \{p \mid p \in SKY\}$

- E_{SKY} is the set of edges. E_{SKY} is a set of ordered pairs $e_{pq} = (p, q)$, where the point p, q are skyline points on space D and there exist two subspaces V, U such that:
 - U is a parent of V
 - $p, q \in SKY_U$ and $q \in SKY_V$
 - q dominates p on subspace V

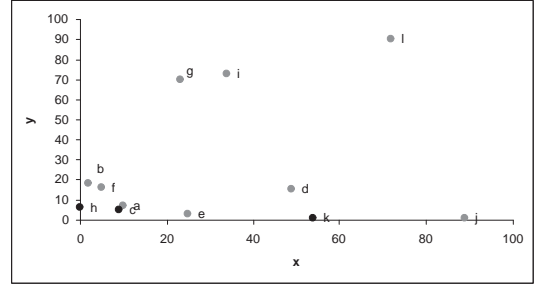
Thus, the set of edges is defined as:

$$E_{SKY} = \{e_{pq} = (p, q) \mid \exists V, U : U = \text{parent}(V), q, p \in SKY_U, q \in SKY_V, q, p \in SKY_U, q \prec_V p\}$$

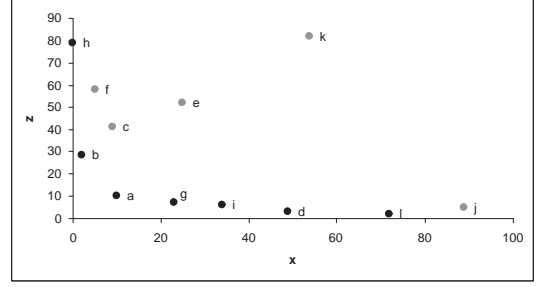
- w_{SKY} is a weight function defined as:
$$w_{SKY}(e_{pq}) = \frac{\text{times}(e_{pq})}{\sum_{e_{ij} \in E_{SKY}} \text{times}(e_{ij})}$$
where $\text{times}(e_{pq})$ the number of U, V such that: $U = \text{parent}(V), q, p \in SKY_U, q \in SKY_V, q, p \in SKY_U, q \prec_V p$.

Intuitively, a skyline point p of subspace U which is dominated by another skyline point q in a child subspace V of subspace U transfers some of its importance to the skyline point q , since skyline point q is at least partial responsible that the point p loses its importance considering subspace V , since it does not belong to the skyline of subspace V . Thus, an edge of the skyline graph means that a skyline point p which is dominated by a skyline point q in some subspace U gives some of its importance/authority to the point q , in a similar way as a web page than links to another web page transfers some of its importance to the linked web page. Therefore, we claim that the propose skyline graph is suitable for ranking the skyline points by applying link-based ranking algorithms such as PageRank.

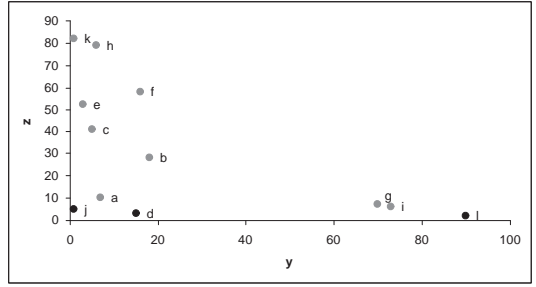
Let $D = \{x, z, y\}$ be the data space of the dataset S . In Table 2 we show the skyline points SKY of the dataset S . In Figure 3 we show the two dimensional projection of the skyline points SKY of the dataset S on space D . Notice that some skyline points are dominated in the $2d$ subspaces, for example the skyline point k is dominated in the subspace $U = \{x, z\}$ from the skyline point a . To illustrate how complex the dominance relationships in all subspaces between the skyline points are, consider the skyline points j and k . The skyline point j dominates the skyline point k in subspace $\{x, y\}$ (Figure 3(a)), while in subspace $\{y, z\}$ the skyline point k dominates the skyline point j (Figure 3(c)). Furthermore, in subspace $\{x, z\}$ (Figure 3(b)) neither the skyline point j nor the skyline point k is a subspace skyline point. In general, as higher the dataset dimensionality is, the more complex the dominance relationships between skyline points are. Figure 4 depicts the part of



(a) subspace $\{x, y\}$



(b) subspace $\{x, z\}$



(c) subspace $\{y, z\}$

Figure 3: Two dimensional subspaces.

the skyline graph that represents the dominance relationships of the subspace $\{x, z\}$. In Figure 4 the shadowed vertices are the skyline points of the subspace $\{x, z\}$.

Notice that based on the skyline frequency considering the two subspaces $\{x, z\}, \{x, y, z\}$ all skyline points of subspace $\{x, z\}$ have the same frequency namely two, whereas in our approach for example point a gains more importance than h which dominates fewer points in the subspace $\{x, z\}$. Moreover, skyline point k is dominated by many points in the subspace $\{x, z\}$, while the point f is dominated only by the point b . Thus, the skyline point f transfers all its importance to the point b , indicating that if the point b was removed from the dataset S , the point f would become a skyline point in the subspace $\{x, z\}$. Furthermore, the importance of skyline point k is shared through the skyline graph edges to many skyline points, because the point k cannot a skyline point in the subspace $\{x, z\}$ by removing only one point from the dataset S . This example indicates the expressiveness of the proposed skyline graph.

In our future work we aim to study the ability of the skyline graph to determine which points should be removed so that a particular point becomes a skyline point in a given subspace. In addition we aim to answer with the skyline graph queries about the decisive subspace [22], i.e. the subspaces which decide that a point

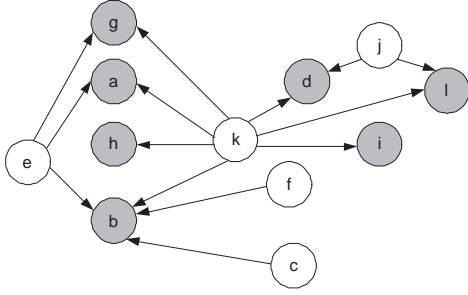


Figure 4: Part of the skyline graph.

is a skyline point. Thus, we consider enhancing edges in the skyline graph with a label indicating the subspaces for which this edge occurs.

5. SKYLINE GRAPH CONSTRUCTION

We present a bottom-up algorithm that relies on [27] that allows us to compute the dominance relationships by sharing skyline results. Since our algorithm for computing the dominance relationship among the skyline points in different subspaces relies on SKYCUBE, our approach can easily be extended in order to provide different ranking for each subspace. However, we focus on providing a ranking of the skyline points in the original data space where the cardinality of the result set is much higher. Notice that our approach is not applicable for ranking skyline points of one-dimensional data spaces but this case is trivial since there are only few (duplicate) values.

Our algorithm for the skyline graph computation relies on BNL [5], SFS [10], BUS [27], thus we first present a short overview of the existing algorithms and then we discuss how to include the skyline graph computation.

5.1 BNL and SFS Algorithms

The authors in [5] introduce the BNL (Block Nested Loop) algorithm. BNL scans the dataset once and keeps a list of candidate skyline points. Initially, the candidate list is empty. Each data point p is compared to every point q in the candidate list. If the data point p is dominated by at least one candidate skyline point q , then the data point p is discarded. Otherwise the data point p is added to the candidate list, and all candidate skyline points that are dominated by the data point p are removed. After examining all data points, BNL outputs all points in the candidate list as skyline points. In case that the candidate list becomes too large to fit in main-memory, a temporary file is used.

SFS (Sort Filter Skyline) [10] improves the performance of BNL by reducing the number of pairwise comparisons between the data points. SFS presorts the data points according to the entropy value of a data point p ($E(p) = \sum_{1 \leq i \leq d} (\ln p(a_i) + 1)$). The gain of sorting is twofold. First due to the monotonicity of the entropy function, a data point p can be dominated only by points that have a smaller entropy value. Secondly, the smaller an entropy value is, the less likely the data point is dominated by others. Therefore, the number of comparisons between data points and candidate skyline points is reduced. The SFS algorithm examines the data points ordered according to the entropy value in a similar way as BNL.

5.2 BUS Algorithm

BUS (Bottom Up Skycube) algorithm [27] computes the SKYCUBE level by level in a bottom-up fashion, while applying sharing

Algorithm 1 Skyline graph construction

```

1: Input:  $U$  denotes the subspace,  $p$  the new skyline point
2:  $SKY_U \leftarrow SKY_U \cup \{p\}$ 
3: if  $!(p \in V_{SKY})$  then
4:    $V_{SKY} \leftarrow V_{SKY} \cup \{p\}$ 
5: end if
6: for (each child subspace  $V$  of  $U$ ) do
7:   if  $!(p \in SKY_V)$  then
8:     for (each point  $q$  such that  $q \prec_V p$ ) do
9:       if  $!(e_{pq} \in E_{SKY})$  then
10:         $E_{SKY} \leftarrow E_{SKY} \cup \{e_{pq}\}$ 
11:       else
12:         $w_{SKY}(e_{pq}) = w_{SKY}(e_{pq}) + 1$ 
13:       end if
14:     end for
15:   end if
16: end for

```

strategies to reduce the computation cost. Throughout the description of BUS we assume that the distinct value condition [27] holds, i.e. no pair of points share the same value on any dimension. As shown in [27], BUS can easily be extended to handle duplicate values.

In order to compute the skyline points on a subspace U , the SFS algorithm is applied. The main difference to SFS is that data points are not sorted based on their entropy on subspace U , but one of the dimensions $d_i \in U$ is used for sorting. Thus, the data points are examined in the order computed so far, reducing the number of sorting operations from $2^d - 1$ to d .

For each skyline computation on subspace U the data points are accessed and compared against the candidate list as in SFS. Since the complexity of the dominance test is $O(d)$, which might be expensive when d increases, BUS aims to reduce the number of dominance tests. When the next data point p belongs to the skyline set SKY_V of any subspace $V \subset U$ then the point p is added to the candidate list without any dominance test.

BUS also applies a filter function ($f_U(p) = \sum_{v_i: d_i \in U} p_i$) in order to reduce even more the number of dominance tests. The monotonicity of the filter function f_U ensures that for two data point p and q , if $f_U(p) \leq f_U(q)$ then q does not dominate p on subspace U . Therefore, BUS keeps the candidate list sorted in a non-decreasing order of their filter values. When evaluating a data point q against a skyline point p , BUS first compares their filter values. If the filter value $f_U(q)$ of q is smaller than p 's filter value $f_U(p)$, p and all the skyline points after p cannot dominate q . Therefore, it is immediately known that q is a new skyline on subspace U and the skyline points are computed incrementally.

5.3 Bottom-up Skyline Graph Algorithm

In this subsection we present a bottom-up algorithm to compute the skyline graph of a dataset S on the data space D . Our algorithm aims to determine the dominance relationships of the skyline points among the different subspaces, while minimizing the required comparisons and dominance tests. Thus, we rely on BUS and we compute the dominance relationships during the subspace skyline computation.

In a similar way to BUS the skyline results are computed according to a bottom-up traversal of the subspace skyline lattice. For each subspace U the data points are retrieved and compared against the already found skyline points. As in BUS for each data point p first we examine if there exists a child subspace V of U on which the point p belongs to the skyline SKY_V . Otherwise the filter values are examined and if this test fails a dominance test occurs. If

a new skyline point p is found, then the process differs from BUS, since the skyline graph has to be updated. Before inserting the new skyline point p into the skyline list, we examine the dominance relationship between the new skyline point p and the skyline points q on every child subspace $V \subset U$ of subspace U . For each child subspace V where p is a skyline, no edge is added to the skyline graph. For each child subspace V where p is not a skyline we retrieve all skyline points q that dominate p in subspace V and add an edge e_{pq} to the skyline graph. If the edge e_{pq} already exists we just increase the number of occurrences of the edge e_{pq} . After all skyline points of the data space D are retrieved, we calculate the weight function w_{SKY} by normalizing the edges' occurrences.

Algorithm 1 sketches the procedure followed when a new skyline point p in the subspace U is found.

6. EXPERIMENTAL EVALUATION

In this section we evaluate the effectiveness of the proposed ranking approach using real-life data. We conducted experiments on the NBA dataset (available from www.nba.com) which consists of 17-dimensional statistics about all players who have played in the NBA from 1946 to 2003. Each dimension represents a certain characteristic of the player such as game points, number of fouls, rebounds, steals and so on. There are over 19,000 tuples, each of them describing the characteristic of a player for a certain year. Therefore, it is possible to have several tuples for the same player, each tuple referring to a different year.

All the experiments were carried out on low-end commodity hardware (3-GHz Pentium PC with 1 GB of memory and local IDE disk, under Windows XP).

6.1 Ranking Results on NBA Dataset

In this section we validate the ranking quality of our approach that relies on the skyline graph and the dominance relationships among skyline points on different subspaces. We construct the skyline graph by using the proposed bottom-up algorithm. Thereafter, we apply PageRank as the ranking algorithm to evaluate the ability of the dominance relationship to provide a meaningful ranking. We compare our ranking results to the skyline frequency metric proposed in [7].

In Table 3(a) and in Table 3(b), we present the top-10 NBA players as resulting by our approach and the skyline frequency metric. The results in Table 3 rely on the 10 out of 17 dimension of the NBA dataset. We similarly report in Table 4 our results for the top-10 NBA players in the case where the 5 out of 17 dimensions of the NBA dataset are considered. Again we rank the NBA players using both our proposed ranking approach (Table 4(a)) and the skyline frequency metric (Table 4(b)).

6.2 Scalability Study

In the next series of experiments we evaluate the scalability of the skyline graph, by means of the number of skylines and the dominance relationships among the skyline points in different subspaces. Recall that the number of skylines is equal to the number of vertices of the skyline graph. The number of dominance relationships among the skyline points relates to the number of edges and the weight function of the skyline graph, i.e. the number of dominance relationships is equal to $\sum_{\forall e_{ij} \in E_{SKY}} times(e_{ij})$. Figure 5(a) depicts the number of skylines, i.e. graph vertices, for the NBA dataset if different number of dimensions are considered, leading to datasets of different dimensionality. We vary the dimensionality from 3 to 10.

As Figure 5(a) illustrates, the cardinality of the skyline increases rapidly, but some dimensions do not influence much the number of

(a) Skyline graph approach

	year	name
1	1989	Hakeem Olajuwon
2	1975	Kareem Abdul-jabbar
3	1978	Moses Malone
4	1973	Julius Erving
5	1975	Julius Erving
6	1988	Michael Jordan
7	1992	Hakeem Olajuwon
8	1988	Hakeem Olajuwon
9	1987	Michael Jordan
10	1979	Michealray Richardson

(b) Skyline frequency approach

	year	name
1	1976	Tom Henderson
2	1968	Walt Bellamy
3	1975	Garfield Heard
4	1977	Kevin Porter
5	2003	Theo Ratliff
6	1996	Mark Jackson
7	1961	Wilt Chamberlain
8	1973	Artis Gilmore
9	1975	Kareem Abdul-jabbar
10	1974	George Mcginnis

Table 3: Top-10 NBA players based on 10 dimensions

skylines since the NBA is a real dataset and there are correlations between some dimensions.

Figure 5(b) shows the number of the dominance relationships while varying the dimensionality from 3 to 10. The number of of the dominance relationships is plotted in a logarithmic scale. As expected, the number of dominance relationships among different subspaces increases even more rapidly than the number of skyline points. This verifies our intuition that the complexity of the dominance relationships increases as the dimensionality of the dataset increases.

7. CONCLUSIONS

Skyline has been proposed as an important operator for many applications. The main drawback of the skyline operator is the high cardinality of the result set. Providing the most interested points of the skyline set (top- k) inherently involves the ranking of the skyline points. In this paper, we propose a method for ranking the skyline points of a data set that allows us to return to the user the top- k skyline points. We map the dominance relationship in different subspaces among skyline points into a weighted directed graph, called *skyline graph*. Using the aforementioned method, we are able to apply link-based techniques to rank the skyline points. We present a bottom-up algorithm that relies on [27] that allows us to compute efficiently the dominance relationships by sharing skyline results. Using PageRank as the ranking algorithm, in the experiments on real-life data, we evaluate the ability of the dominance relationship to provide a meaningful ranking.

8. REFERENCES

- [1] A. Balmin, V. Hristidis, and Y. Papakonstantinou. ObjectRank: Authority-based keyword search in databases. In *Proceedings of International Conference of Very Large Data Bases (VLDB)*, pages 564–575, Toronto, Canada, 2004.

(a) Skyline graph approach			(b) Skyline frequency approach		
	year	name		year	name
1	1961	Wilt Chamberlain	1	1968	Walt Bellamy
2	1974	Bob Mcadoo	2	1961	Wilt Chamberlain
3	1978	Moses Malone	3	1975	Garfield Heard
4	1969	Spencer Haywood	4	1976	Tom Henderson
5	1973	Artis Gilmore	5	1973	Artis Gilmore
6	1973	Elvin Hayes	6	1973	Chuck Williams
7	1977	Truck Robinson	7	1973	Elvin Hayes
8	1981	Moses Malone	8	1978	Moses Malone
9	1986	Michael Jordan	9	1974	Bob Mcadoo
10	1975	Kareem Abdul-jabbar	10	1981	Mike Mitchell

Table 4: Top-10 NBA players based on 5 dimensions

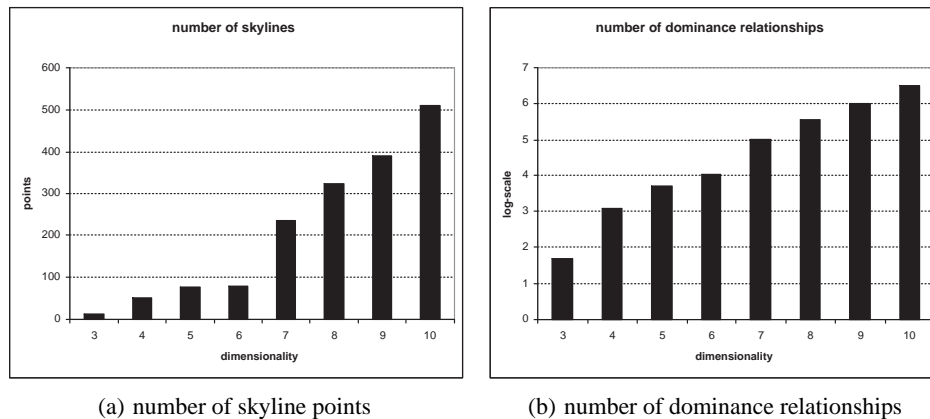


Figure 5: Scalability study over NBA dataset.

- [2] I. Bartolini, P. Ciaccia, V. Oria, and M. T. Ozsü. Flexible integration of multimedia sub-queries with qualitative preferences. *Multimedia Tools Appl.*, 33(3):275–300, 2007.
- [3] M. Bianchini, M. Gori, and F. Scarselli. Inside pagerank. *ACM Transactions on Internet Technology (TOIT)*, 5(1):92–128, 2005.
- [4] A. Borodin, G. O. Roberts, J. S. Rosenthal, and P. Tsaparas. Link analysis ranking: algorithms, theory, and experiments. *ACM Transactions on Internet Technology (TOIT)*, 5(1):231–297, 2005.
- [5] S. Börzsönyi, D. Kossmann, and K. Stocker. The skyline operator. In *Proceedings of IEEE International Conference on Data Engineering (ICDE)*, pages 421–430, 2001.
- [6] S. Brin and L. Page. The anatomy of a large-scale hypertextual Web search engine. *Computer Networks*, 30(1–7):107–117, 1998.
- [7] C. Y. Chan, H. V. Jagadish, K.-L. Tan, A. K. H. Tung, and Z. Zhang. On high dimensional skylines. In *Proceedings of International Conference on Extending Database Technology (EDBT)*, pages 478–495, 2006.
- [8] S. Chaudhuri, N. N. Dalvi, and R. Kaushik. Robust cardinality and cost estimation for skyline operator. In *Proceedings of IEEE International Conference on Data Engineering (ICDE)*, page 64, 2006.
- [9] S. Chaudhuri and L. Gravano. Evaluating top- k selection queries. In *Proceedings of International Conference of Very Large Data Bases (VLDB)*, pages 397–410, 1999.
- [10] J. Chomicki, P. Godfrey, J. Gryz, and D. Liang. Skyline with presorting. In *Proceedings of IEEE International Conference on Data Engineering (ICDE)*, pages 717–719, 2003.
- [11] E. Dellis, A. Vlachou, I. Vladimirskiy, B. Seeger, and Y. Theodoridis. Constrained subspace skyline computation. In *Proceedings of ACM Conference on Information and Knowledge Management (CIKM)*, pages 415–424, 2006.
- [12] R. Fagin, A. Lotem, and M. Naor. Optimal aggregation algorithms for middleware. In *Proceedings of ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems (PODS)*, pages 102–113, 2001.
- [13] F. Geerts, H. Mannila, and E. Terzi. Relational link-based ranking. In *Proceedings of International Conference of Very Large Data Bases (VLDB)*, pages 552–563, Toronto, Canada, 2004.
- [14] P. Godfrey. Skyline cardinality for relational processing. In *Foundations of Information and Knowledge Systems (FoIKS)*, pages 78–97, 2004.
- [15] T. H. Haveliwala. Topic-sensitive PageRank. In *Proceedings of the International Conference on World Wide Web (WWW)*, pages 517–526. ACM Press, 2002.
- [16] G. Y. J. Lee and S. Hwang. Telescope: Zooming to interesting skylines. In *Proceedings of International Conference on Database Systems for Advanced Applications (DASFAA)*, 2007.
- [17] G. Jeh and J. Widom. Scaling Personalized Web Search. In *Proceedings of the International Conference on World Wide Web (WWW)*, pages 271–279. ACM Press, 2003.
- [18] J. M. Kleinberg. Authoritative Sources in a Hyperlinked

- Environment. *Journal of the ACM (JACM)*, 46(5):604–632, 1999.
- [19] D. Kossmann, F. Ramsak, and S. Rost. Shooting stars in the sky: An online algorithm for skyline queries. In *Proceedings of International Conference of Very Large Data Bases (VLDB)*, pages 275–286, 2002.
- [20] Z. Nie, Y. Zhang, J.-R. Wen, and W.-Y. Ma. Object-Level Ranking: Bringing Order to Web Objects. In *Proceedings of the International Conference on World Wide Web (WWW)*, pages 567–574, 2005.
- [21] D. Papadias, Y. Tao, G. Fu, and B. Seeger. Progressive skyline computation in database systems. *ACM Transactions on Database Systems (TODS)*, 30(1):41–82, 2005.
- [22] J. Pei, W. Jin, M. Ester, and Y. Tao. Catching the best views of skyline: A semantic approach based on decisive subspaces. In *Proceedings of International Conference of Very Large Data Bases (VLDB)*, pages 253–264, Trondheim, Norway, 2005.
- [23] K.-L. Tan, P.-K. Eng, and B. C. Ooi. Efficient progressive skyline computation. In *Proceedings of International Conference of Very Large Data Bases (VLDB)*, pages 301–310, 2001.
- [24] Y. Tao, X. Xiao, and J. Pei. SUBSKY: Efficient computation of skylines in subspaces. In *Proceedings of IEEE International Conference on Data Engineering (ICDE)*, page 65, 2006.
- [25] A. Vlachou, C. Doulkeridis, M. Vazirgiannis, and Y. Kotidis. Skypeer: Efficient subspace skyline computation over distributed data. In *Proceedings of IEEE International Conference on Data Engineering (ICDE)*, pages 416–425, 2007.
- [26] T. Xia and D. Zhang. Refreshing the sky: the compressed skycube with efficient support for frequent updates. In *Proceedings of ACM SIGMOD International Conference on Management of Data*, pages 491–502, 2006.
- [27] Y. Yuan, X. Lin, Q. Liu, W. Wang, J. X. Yu, and Q. Zhang. Efficient computation of the skyline cube. In *Proceedings of International Conference of Very Large Data Bases (VLDB)*, pages 241–252, Trondheim, Norway, 2005.