Minimizing Navigation Cost Through Interactive Data Exploration and Discovery

[1]Srilaxmi Challa, [2]Dr.R.V.Krishnaiah

[1] Department of CSE, JNTU H, DRK College of Engineering and Technology
Hyderabad, Andhra Pradesh, India

[2] Principal, Department of CSE, JNTU H, DRK College of Engineering and Technology
Hyderabad, Andhra Pradesh, India

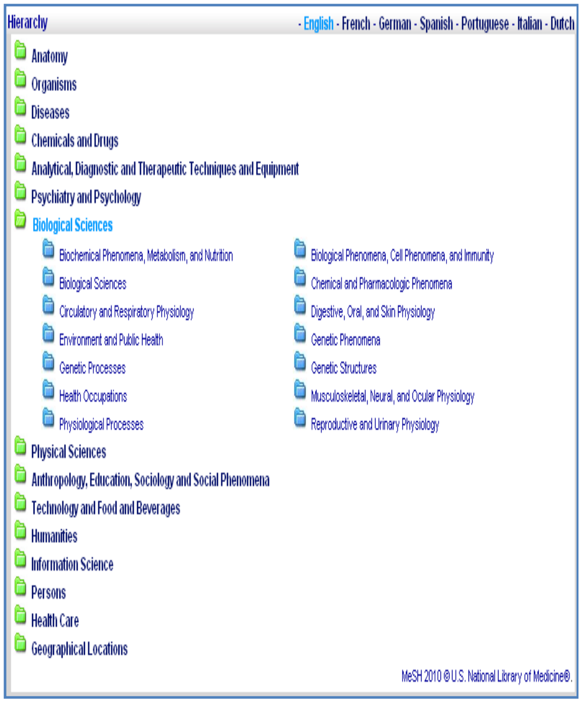
Abstract

Web databases when queried result in huge number of records when users of query need a portion of those results which are real interest to them. This problem can be solved using concept hierarchies. Knowledge representation in the form of concepts and the relationships among them (Ontology) allows effective navigation. This paper presents provisions for categorization and ranking in order to reduce the number of results of query and also ensure that the navigation is effective. User should not spend much time to view the actual subset of records he is interested in from the avalanche of records that have been retrieved. For experiments, PubMed database which is in the public domain is used. The PubMed data is medical in nature and organized as per the annotations provided that is instrumental in making concept hierarchies to represent the whole dataset of PubMed. The proposed technique in this paper provides a new search interface that facilitates end users to have effective navigation of query results that are presented in the form of concept hierarchies. Moreover the query results are presented in such a way that the navigation cost is minimized and thus giving rich user experience in this area. The empirical results revealed that the proposed navigation system is effective and can be adapted to real world systems where huge number of tuples is to be presented.

Keywords- *Web Database , Opt Edge Cut Algorithm*

1. Introduction

The amount of data provided over World Wide Web (WWW) is increasing rapidly every year. In the past decade in started growing drastically. Especially biomedical data and the literature pertaining to it that reviews the aspects of biomedical data across the globe have seen tremendous growth in terms of quantity. Biological data sources such as [1], [2], and [3] are growing in terms of laths of new citations every year. The queries made by people associated with healthcare domain have to search such databases by providing a search keyword. The results are very huge in number and the users are not able to view all the records when they actually need a subset of them. This has led to users to refine query with other keywords and get the desired results after many trials. Here it has to be observed that user time is wasted in refining search criteria and also the navigation of query results which are abundant and bulky. The navigation cost is more as user has to spend lot of time in finding the required subset of rows from the bulk of search results. This problem has been researched in [1], [2], [3] and the problem is identified as information overload. Figure 1 shows static navigation of MeSh hierarchy of biomedical data.



Hierarchy - English - French - German - Spanish - Portuguese - Italian - Dutch

- Anatomy
- Organisms
- Diseases
- Chemicals and Drugs
- Analytical, Diagnostic and Therapeutic Techniques and Equipment
- Psychiatry and Psychology
- Biological Sciences
  - Biochemical Phenomena, Metabolism, and Nutrition
  - Biological Sciences
  - Circulatory and Respiratory Physiology
  - Environment and Public Health
  - Genetic Processes
  - Health Occupations
  - Physiological Processes
  - Biological Phenomena, Cell Phenomena, and Immunity
  - Chemical and Pharmacologic Phenomena
  - Digestive, Oral, and Skin Physiology
  - Genetic Phenomena
  - Genetic Structures
  - Musculoskeletal, Neural, and Ocular Physiology
  - Reproductive and Urinary Physiology
- Physical Sciences
- Anthropology, Education, Sociology and Social Phenomena
- Technology and Food and Beverages
- Humanities
- Information Science
- Persons
- Health Care
- Geographical Locations

MeSH 2010 © U.S. National Library of Medicine®.

Fig. 1 – MeSH Hierarchy [2]

The solutions are of two types namely categorization and ranking. However, these two can be combined to have more desired results. The proposed system is specially meant for presenting results in such a way that the navigation cost is reduced. For this purpose categorization techniques is used and concept hierarchies are built. The categorization techniques are supported by simple ranking techniques. The proposed solution uses citations as described in [4],[8] and effectively constructs a navigation tree that can reduce cost of navigation and user's experience is much better when compared with existing systems that do not use these techniques. These techniques are being used by e-Commerce systems to let their users have smooth navigation to the results returned by such systems.

The proposed system uses a cost model that lets it estimate the cost of navigation and make decisions in providing concept hierarchies. The cost of navigation is directly proportionate to the navigation sub tree[10] instead of the whole results in the tree. Earlier work on dynamic categorization of query results are in [2], [3], [5] and [6]. They made use of query dependent clusters based on the unsupervised technique. However, they neglect the process of navigation of clusters. In this aspect the proposed system is distinct and provides dynamic navigation on a pre-defined concept hierarchy. Another telling difference between existing systems and the proposed one is that the proposed system uses navigation cost model that minimizes navigation cost no matter what the bulky of search results is. Overall, our contributions are development of a framework for effective navigation of query results; a formal model for cost estimation; algorithm to optimize the results' navigation cost; experimental evidence on the effectiveness.

2. Architecture of Proposed Framework

The proposed framework is meant for making navigation of query results as effective as possible. The results of this project enable end users save lot of time as the proposed framework reduces the time taken to reach valuable content in the hierarchy.

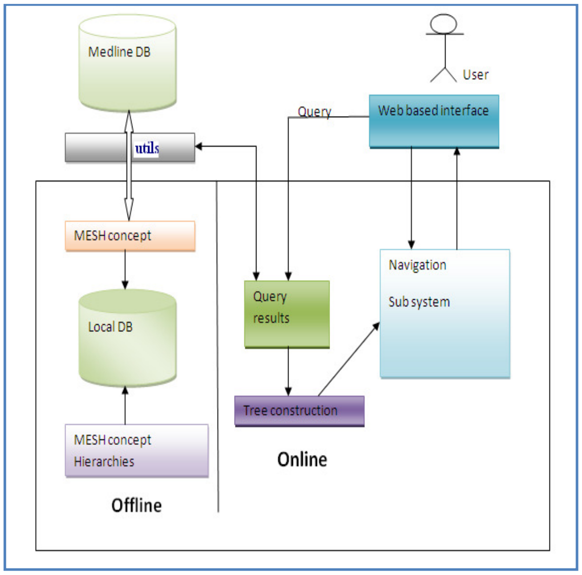
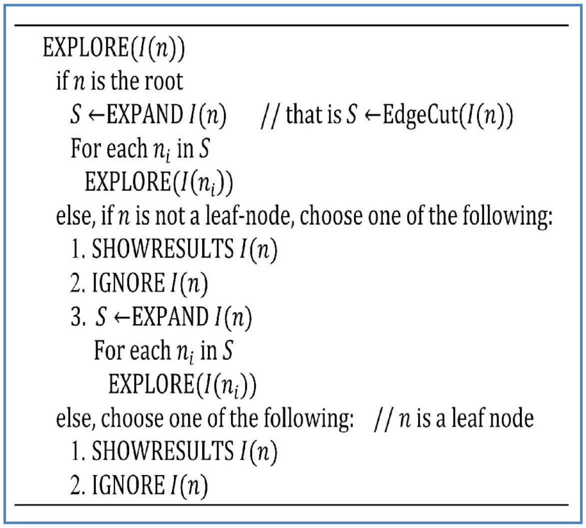


Fig. 2 – Architecture of Proposed Framework

As can be seen in fig. 2, the proposed framework has two phases such as Online and Offline. The offline phase performs operations in which user's active presence is not required. The Online phase is responsible to perform operations that are direct responses to user queries and also navigation operations made by user. The Medline DB has Mesh [9]concepts that can be loaded into local database using utility programs which are provided by the DB vendors. The Mesh concepts thus downloaded are stored in local database. In online phase, user enters a query. The query gets processed and results are obtained from database. The results then are used to construct concept hierarchies. The navigation sub system is responsible to take care of fine-tuning navigation tree so as to reduce the time for viewing desired results only. User is provided with a web based interface though which users can determine giving queries and the results get presented.

3. Algorithms

Navigation model is described in fig. 3. It makes use of the following to calculate the navigational cost. Number of EXPAND actions, Number of concept nodes shown by a single EXPAN action and Number of citations presented for a single SHOWRESULTS action.



```
EXPLORE($I(n)$)
  if $n$ is the root
    $S \leftarrow$EXPAND $I(n)$    // that is $S \leftarrow$EdgeCut($I(n)$)
    For each $n_i$ in $S$
      EXPLORE($I(n_i)$)
  else, if $n$ is not a leaf-node, choose one of the following:
    1. SHOWRESULTS $I(n)$
    2. IGNORE $I(n)$
    3. $S \leftarrow$EXPAND $I(n)$
       For each $n_i$ in $S$
         EXPLORE($I(n_i)$)
  else, choose one of the following:    // $n$ is a leaf node
    1. SHOWRESULTS $I(n)$
    2. IGNORE $I(n)$
```

Fig. 3 - Navigation model in TOPDOWN fashion [1]

The EXPAND operations shows set of related nodes. SHOWRESULTS shows results to end user. IGNORE is used to ignore a node and move on to the other desired results. BACKTRACK occurs when undo is performed by end user.

3.1 Opt Edge Cut Algorithm

The Opt-EdgeCut algorithm shown in fig. 4 which is responsible to calculate the minimum expected navigational cost.

```
Algorithm. Opt-EdgeCut
Input: The navigation tree $T$
Output: The best EdgeCut
Traversing $T$ in postorder, let $n$ be the current node
while $n \neq root$ do
  if $n$ is a leaf node then
    $mincost(n, \emptyset) \leftarrow P_E(n) * L(n)$
    $optcut(n, \emptyset) \leftarrow \{\emptyset\}$
  else
    $\mathbb{C}(n) \leftarrow$ enumerate all possible EdgeCuts
                 for the tree rooted at $n$
    $\mathbb{I}(n) \leftarrow$ enumerate all possible subtrees
                 for the tree rooted at $n$
    foreach $I(n) \in (n)$ do
      compute $P_E(I(n))$ and $P_c(I(n))$
      foreach $C \in (n)$ do
        if $C$ is a valid EdgeCut for $I(n)$ then
          $cost(I(n), C) \leftarrow$
$P_E(I(n)) \cdot \left( \begin{smallmatrix} (1 - P_c(I(n))) \cdot L(I(n)) \\ + P_c(I(n)) \cdot (B + |S| + \sum_{s \in S} mincost(I_C(s))) \end{smallmatrix} \right)$
        else
          $\cos t(I(n), C) = \infty$
      $mincost(n, I(n)) \leftarrow \min_{c_i \in \mathbf{C}(n)} cost(I(n), C_i)$
      $optcut(n, I(n)) \leftarrow C_i$
return $optcut(root, E)$    //$E$ is the set of all tree edges
```

Fig. 4 – Opt-EdgeCut Algorithm [1]

The algorithm proposed in fig. 4 is more expensive in terms of computational cost. To overcome this drawback, heuristic reduced opt algorithm is proposed. According to this algorithm, which makes use of k-partition algorithm [10] and also pruning concepts to improve performance of navigation.

```
Algorithm. Heuristic-ReducedOpt
Input: Component subtree $I(n)$, number $z$ of partitions
Output: The best EdgeCut
$z' \leftarrow z$
repeat
  $k \leftarrow \sum_{n \in T} L(n) \cdot P_E(n) / z'$
  $Partitions \leftarrow k\text{-partition}(I(n), k)$
      // call $k$-partition algorithm [14]
  $z' \leftarrow z' - 1$
until $|Partitions| \leq z$
construct reduced subtree $I'(n)$ from $Partitions$
$EdgeCut' \leftarrow Opt\text{-}EdgeCut(I'(n))$
$EdgeCut \leftarrow$ corresponding of $EdgeCut'$ for $I(n)$
return $EdgeCut$
```

Fig. 5 – Heuristic-ReducedOpt Algorithm [1]

4. Experiments

4.1 Environment

The environment used for experiments include a PC with 2 GB RAM, 2.93GHz processor with Windows XP OS. The software used include JDK 1.6 (also known as Java Standard Edition 6.0), Net Beans IDE (for rapid application development), browser. The technologies used include Servlets and JSP. The home page of the application is as shown in fig. 6.

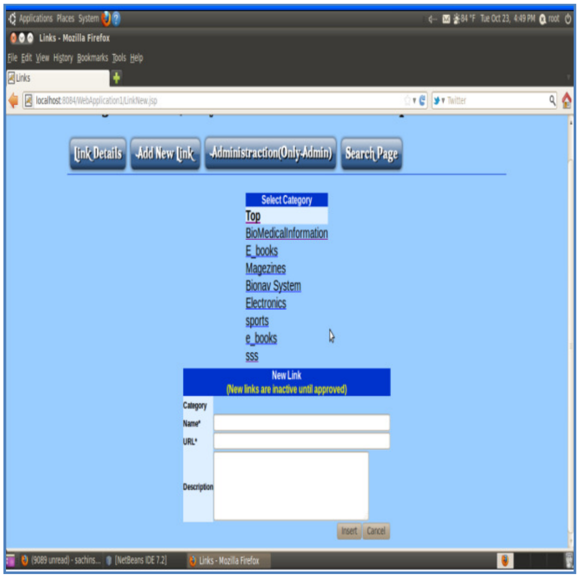


Fig. 6 – Home page of the application

As can be seen in fig. 6, the home page facilitates the search operations besides other admin operations. The search results and the navigation tree are presented in fig. 7.

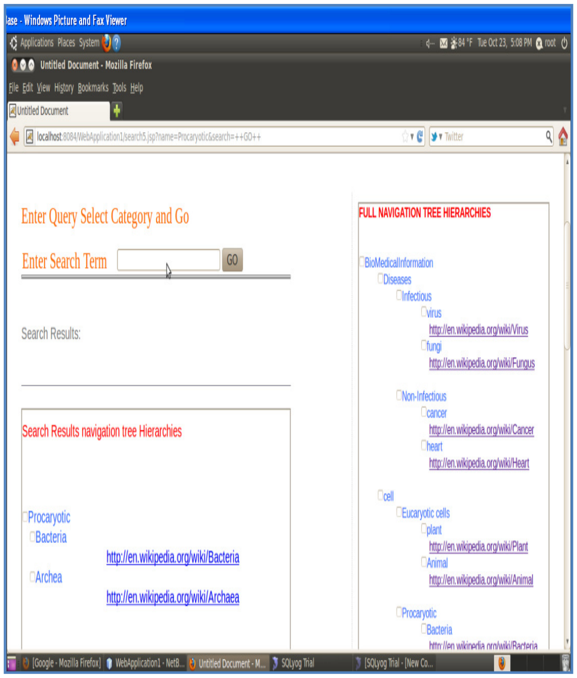

ase - Windows Picture and Fax Viewer
Applications Places System
84 °F Tue Oct 23, 5:08 PM root
Untitled Document - Mozilla Firefox
File Edit View History Bookmarks Tools Help
Untitled Document
localhost:8084/WebApplication1/search5.jsp?name=Procaryotic&search=++GO++
Twitter
Enter Query Select Category and Go
Enter Search Term
GO
Search Results:
Search Results navigation tree Hierarchies
Procaryotic
Bacteria
http://en.wikipedia.org/wiki/Bacteria
Archea
http://en.wikipedia.org/wiki/Archaea
FULL NAVIGATION TREE HIERARCHIES
BioMedicalInformation
Diseases
Infectious
virus
http://en.wikipedia.org/wiki/Virus
fungi
http://en.wikipedia.org/wiki/Fungus
Non-Infectious
cancer
http://en.wikipedia.org/wiki/Cancer
heart
http://en.wikipedia.org/wiki/Heart
cell
Eucaryotic cells
plant
http://en.wikipedia.org/wiki/Plant
Animal
http://en.wikipedia.org/wiki/Animal
Procaryotic
Bacteria
http://en.wikipedia.org/wiki/Bacteria
[Google - Mozilla Firefox]
WebApplication1 - NetB...
Untitled Document - M...
SQLyog Trial
[SQLyog Trial - [New Co...

Fig. 7 – Search Process and Results

After making experiments with the proposed framework using a web based application the results are shown in the form of graphs. Fig. 6 shows comparison of number of expand operations.

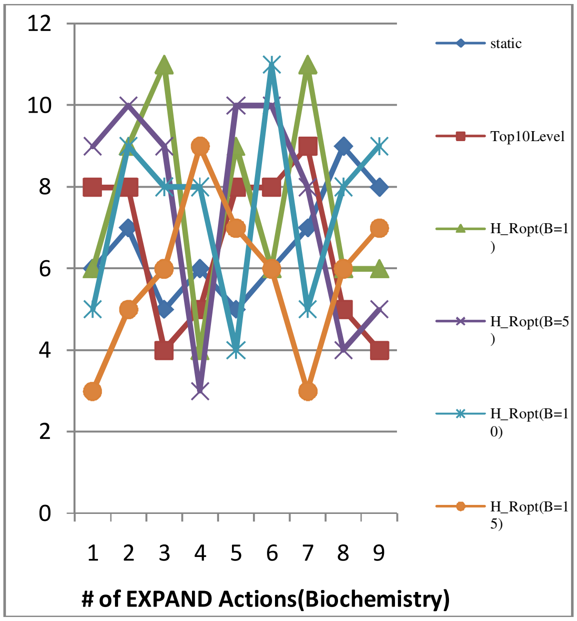


Fig. 6 – Comparison of number of expand operations

The results of EXPAND operations for various approaches are visualized in fig. 6. In X axis query numbers are presented while the Y axis reflects the count of EXPAND operations.

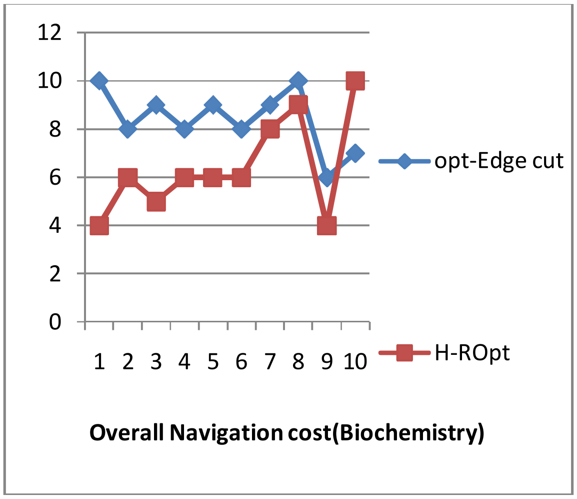


Fig. 7 – Comparison of number of concepts revealed

For the biochemistry database, the operaal number of concepts revealed are presented in fig. 7. The graph compares overall navigation cost of the algorithms such as Opt-EdutCut and Heuristic-ReducedOpt algorithms. As is evident in the figure, Heuristic – ReducedOpt algorithm performance is much better than that of Opt-EdgeCut.

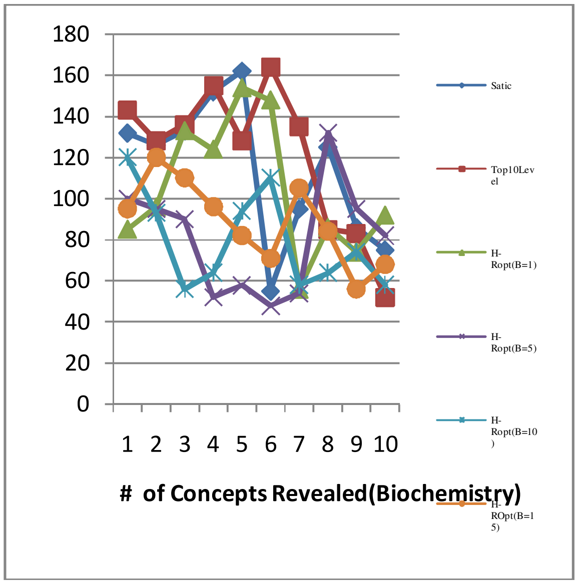


Fig. 8 – Comparison of overall navigation cost

The results of number of concepts revealed when OptEdgeCut and Heuristic-ReducedOpt are compared. The results reveal that the Heuristic-ReducedOpt is far better than Opt-EdgeCut algorithm in terms of overall navigational cost incurred by those algorithms when implemented for effective query navigation of results.

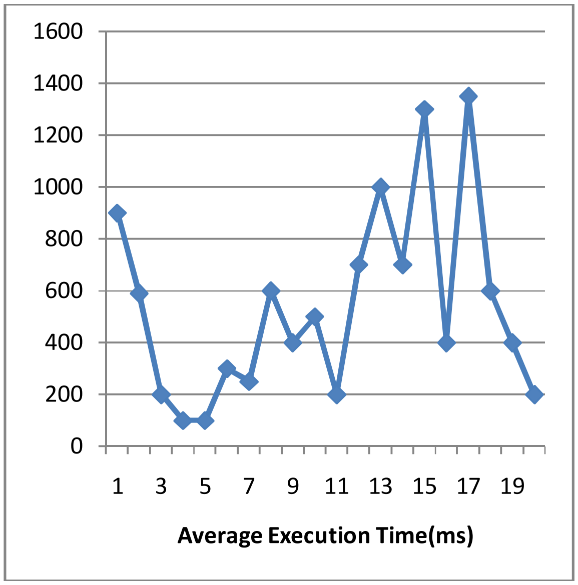


Fig. 9 - Heuristic-ReducedOpt EXPAND performance.

As seen in fig. 9, the average time of Heuristic-ReducedOpt to execute and EXPAND action with respect to each query of table 1. The average values are taken from the number of EXPAND action provided in fig. 6.

5. Conclusion

This paper presents a framework for effective navigation of results of query given to biomedical databases such as PubMed. The problem with query results is that biomedical database returns millions of records and users have to spend some time to navigate to the desired records in the results. This is known as navigation cost. Such problem is also known as information overload problem. The aim of the proposed framework is to address the problem by reducing navigation cost. We achieve this by organizing the results based on the associated MeSH (Medical Subject Headings) hierarchy by proposing a method that works on the resulting navigation tree. The method is known as dynamic navigation method. After applying this method, every node when expanded reveals a subset of required rows thus reducing navigation cost. We have described the underlying cost models and also evaluated them. We developed a prototype application to test the framework's functionality. The empirical results revealed that the proposed framework is effective and can be used in the real time applications.

References

[1] Abhijith Kashyap, Vagelis Hristidis, Michalis Petropoulos, and Sotiria Tavoulari. Effective Navigation of Query Results Based on Concept Hierarchies. IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, VOL. 23, NO. 4, APRIL 2011.

[2] HONSelect (2012). Available online at <http://www.hon.ch/cgi-bin/HONselect?cat+G#MeSH> [viewed: 10 September 2012]

[3]. Z. Chen and T. Li: *Addressing Diverse User Preferences in SQLQuery- Result Navigation*. SIGMOD Conference 2007: 641-652.

[4]. Medical Subject Headings (MeSH®). http://www.nlm.nih.gov/mesh/

[5]. (2008) Vivísimo, Inc. –Clusty. [Online].Available: http://clusty.com/

[6]. A. Kashyap, V. Hristidis, M. Petropoulos, and S. Tavoulari: *BioNav: Effective Navigation on Query Results of Biomedical Databases*. (Short Paper), ICDE 2009, to appear. Available at http://www.cs.fiu.edu/~vagelis/publications/BioNavICDE09.pdf

[7] Medical Subject Headings (MeSH), http: //www.nlm.nih.gov/ mesh/, 2010.

[8] Abhijith Kashyap, Vagelis Hristidis, Michalis Petropoulos, and Sotiria Tavoulari (2011), "Effective Navigation of Query Results Based on Concept Hierarchies". IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, VOL. 23, NO. 4.

[9] S. Kundu and J. Misra, "A Linear Tree Partitioning Algorithm," SIAM J. Computing, vol. 6, no. 1, pp. 151-154, 1977.