Dariusz Bernatowicz
Department of Electronics and Computer Science

Anna Bernatowicz
Department of Civil, Environmental Engineering and Survey

Technical University of Koszalin
2 Śniadeckich St.
75-343 Koszalin
Poland

Application of correlation in the vertical fragmentation based on statistic of queries

Keywords: vertical fragmentation, distributed databases, graphical partitioning algorithm, matrix and graph of correlation

1. Introduction

The design of distributed databases usually applies to the problems of data fragmentation, allocation and replication. The main goal of distribution is to improve efficiency and to increase the reliability of the system. The first goal is especially important because the distribution of data is a consequence of the nature of distributed organization and the need for local searching and data processing as well. In practice those problems are complex (NP-hard) because of that they are mostly considered separately and they could be solved using heuristics which fulfill chosen **criteria.**

The first stage of distributed data design is fragmentation defined as a partition of single set of attributes for a given relation into two or more parts. However joining back all of the parts allows to achieve original set of data without losing any piece of information [9].

The aim of such a partitioning is to minimize the cost of processing for a given set of attributes. The set of attributes is defined by the objective function and estimated quality of obtained distribution [1]. Criterion of data distribution in the fragmentation determines its type: vertical, horizontal or mixed that is a hybrid of mentioned two. Vertical partitioning is most commonly investigated because it is characterized by much higher degree of complexity. The existing solutions for vertical fragmentation are based on a model that describes the statistic of queries that arrive to centralized system. This statistic define empirical data related with the type and frequency of the queries.

An approach based on statistic of queries utilize affinity matrix as an input for partitioning algorithms. Partitioning algorithms based upon graph theory are mainly used. For these kind of graphs attributes and values of affinity matrix define vertexes and edges of the graph. Simplification of graph structure (reducing the number of edges) could be achieved by the application of modified affinity matrix or alternative approaches based on statistic of queries.

This paper proposes an approach based on an application of correlation which is a measure of the statistical relationship between inputs. It allows direction and strength of relation between attributes and it remains an alternative technique for reduction of the number of edges in the graph.

The reminder of the paper is structured as follows. Section 2 includes the characteristic of vertical fragmentation databases. Section 3 describes the concept and features of the approach based on affinity matrix and graphical algorithm. An alternative approach based on correlation matrix is described in section 4 whereas development of graphical algorithm that solves the problem of disconnected graph is presented in section 5. Section 4 proposes alternative approach in a later section there is a. Conclusions and anticipated directions for a future work are provided in Section 6.

2. Characteristics of vertical partitioning

Vertical partitioning is the problem of clustering the attributes of a relation into fragments named as a partitioning scheme. This scheme represents the input for the allocation process in the next phase of designing data distribution. It should minimize the execution time of user's application that uses the obtained fragments. Large number of possible solutions called Bell's number makes the vertical partitioning too expensive while using traditional methods. For example, the set of ten attributes produces B10 = 115 975 of possible partitions whiles the set of 15-attributes gives B15 = 1 382 958 545. Such a rapid growth of search space results in the need for using heuristic techniques with determined objective function. This problem should be considered in the context of the optimization process. Heuristic techniques are incomplete algorithms which enable finding an approximate solution of the problem within acceptable execution time. Efficiency of these algorithms can be determined within two categories: the quality of partitioning scheme and the computational complexity of the algorithm.

Chakravathy [1] proposes quality evaluation of the obtained partition scheme. It specifies the minimum value of processing cost for set of transactions. It depends on the frequency of occurrence and fragmentation of elements to which they access.

The quality evaluation enables to determine and compare the ”goodness” of obtained partition schema considering the same input data. It also allows to balance the costs of the local and remote access to certain attributes by particular transactions.

The best results (similar to optimal) are obtained by metaheuristics [3],[5] but they are not used commonly due to high complexity (for example population management). For the same reason they are less popular than classical algorithms. Considering the criterion of computational complexity classical algorithms and their modifications are the most efficient. Such an algorithm include the following: RBPA [8], GPA [10] i CBPA[4]. All of above algorithms use Affinity Matrix as an input. The matrix can be entered directly or after normalization.

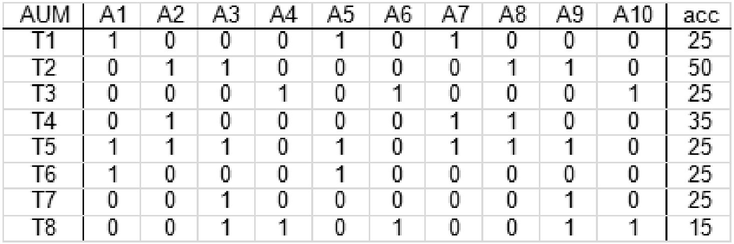
3. Approach based on affinity matrix

Vertical partitioning algorithms based upon statistic of queries utilize the Attribute Usage Matrix (AUM) as an input. This matrix determines both transaction attribute reference times as well as transaction access frequencies within particular timeframe. Values of the matrix are defined with the following function:

$$AUM(T_i, A_j) = \begin{cases} 1 & if\ attribute\ A_j\ is\ referred\ by\ transaction\ T_i \\ 0 & otherwise \end{cases} \tag{1}$$

Values of the matrix define the access by transactions (rows of the matrix) to attributes (columns of the matrix) with certain frequency. An example of representative AUM matrix considered in [4],[8] and [10] is shown in table 1. It contains 8 transactions referring to 10 attributes with the frequency determined in column acc.

Table 1. The AUM for example 1



AUM	A1	A2	A3	A4	A5	A6	A7	A8	A9	A10	acc
T1	1	0	0	0	1	0	1	0	0	0	25
T2	0	1	1	0	0	0	0	1	1	0	50
T3	0	0	0	1	0	1	0	0	0	1	25
T4	0	1	0	0	0	0	1	1	0	0	35
T5	1	1	1	0	1	0	1	1	1	0	25
T6	1	0	0	0	1	0	0	0	0	0	25
T7	0	0	1	0	0	0	0	0	1	0	25
T8	0	0	1	1	0	1	0	0	1	1	15

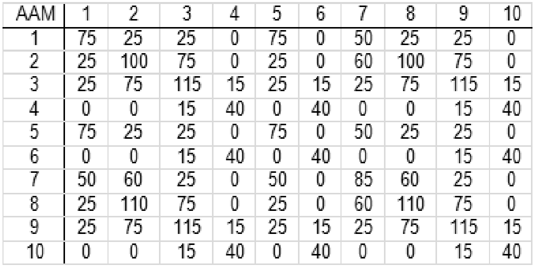
An Attribute Affinity Matrix (AAM) proposed in [6] is derived from the AUM and frequency vector acc. It specifies affinity values between two attributes A_i and A_j of a relation $R(A_1,A_2,...,A_n)$ as the sum of the concurrent access frequency of any two attributes for each transaction. Values of the matrix are defined by the following equation.

$$AAM_{ij} = \sum_{k|AUM(T_k,A_i)=1,\ AUM(T_k,A_j)=1} acc(T_k) \tag{2}$$

An example of the AAM obtained from AUM is shown in table 2. It Has symmetrical structure and diagonal elements *are* always of the highest value found in the corresponding columns and rows. Both, rows and columns, represent attributes of the matrix and the values belong to the set of positive integers ($N_{+} \cup \{0\}$). High affinity value indicates that the same transactions access to the same pair of attributes more often. Hence that the possibility of occurrence of the same pair of attributes in the same fragment increases. For a zero value the pair of attributes have no transactions in common or the transactions are characterized by low frequency so the attributes are placed in different fragments.

The figure 1 shows an undirected affinity graph based on AAM where the numbers of edges define the attributes and edges of graph contain affinity value between given vertexes. The number of graph's edges represents the difference between edges of full graph defined as n(n-1)/2 and the number of zero values edges. For the graph in Fig. 1 the number of edges is 45-15 = 30.

Table 2. The AAM for Example 1



AAM	1	2	3	4	5	6	7	8	9	10
1	75	25	25	0	75	0	50	25	25	0
2	25	100	75	0	25	0	60	100	75	0
3	25	75	115	15	25	15	25	75	115	15
4	0	0	15	40	0	40	0	0	15	40
5	75	25	25	0	75	0	50	25	25	0
6	0	0	15	40	0	40	0	0	15	40
7	50	60	25	0	50	0	85	60	25	0
8	25	110	75	0	25	0	60	110	75	0
9	25	75	115	15	25	15	25	75	115	15
10	0	0	15	40	0	40	0	0	15	40

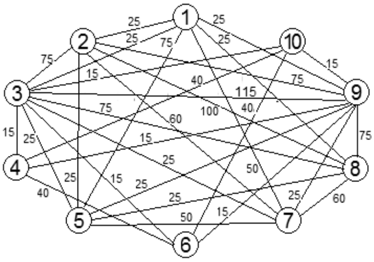


Fig. 1. Affinity graph obtained from AAM for example 1

In case of graphical partitioning algorithm GPA (based on affinity graph) the graph linearization is performed by creating the tree of spread and next the cycles of fragments identification are searched. Although the reduction of complexity O(n^2) is identified as the main advantage of the algorithm but the quality of partitioning stays low especially for large number of attributes [10]. Because of low scalability there are many alternative approaches based on modifications of the graph. They refer to reduction of edges obtained by normalization of the affinity matrix and introduction of additional limiting parameters. An example of such algorithm is CBPA. It enables to transform AAM into connection matrix for which particular values determine the strength of connection between the pair of attributes. Next the connection graph is created by limiting the number of edges and adopting an arbitrary minimum threshold of connection strength of a pair of attributes that is called acceptance threshold. Shortage of univocal rules to define the minimum acceptance threshold values as well as choosing the additional parameters creates significant problem and affects the quality of partitioning.

Presented approach allows to reduce the edges without limiting parameter by applying correlation as a measure of relationship between the attributes.

4. Matrix and graph of attributes correlation

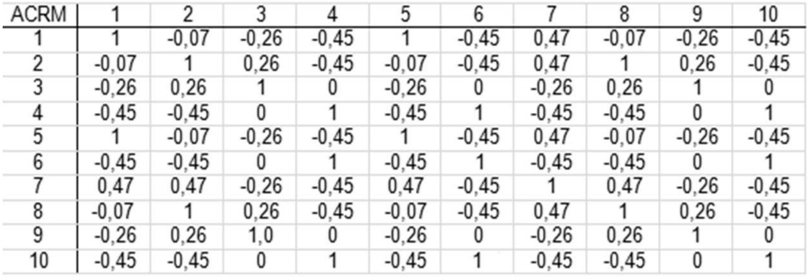
Affinity Matrix specifies the dependence between pairs of attributes as a sum of frequency of simultaneous access to these attributes by particular transactions. Values of the matrix are positive integers so the occurrence of a single transaction instance that refers to the pair of attributes will produce the edge in a graph.

Occurrence of graph's edge with a low affinity value increases complexity of the graph but does not affect partition scheme. In order to avoid irrelevant edges and to reduce complexity of the graph Pearson correlation coefficient was used. The Pearson correlation coefficient is a measure of strength and direction of linear dependence between two variables [11]. A basic property of Pearson's ρ coefficient is that the direction can be positive or negative and its values fit the range of $-1 \leq \rho \leq 1$. Using this relation the symmetric matrix called Attribute Correlation Matrix (ACRM) can be generated. The ACRM defines a connection between two attributes Ai and A j of the AUM matrix. Matrix values are defined as:

$$ACRM_{ij} = \begin{cases} 0 < \rho_{A_iA_j} \leq 1, & if\ positive\ relationship \\ -1 \leq \rho_{A_iA_j} < 0, & if\ negative\ relationship \\ 0, & if\ no\ correlation \end{cases} \tag{3}$$

A characteristic feature of the ACRM is the possibility of negative relationship between a pair of attributes. It is also the main difference comparing to AAM. It means that most of the transactions refer to one attribute only and low frequency of occurrence can be noted for common transactions. In case of lack of correlation none of the transactions refers to both attributes at the same time. For all cases above particular pair of attributes is not present in the same fragment so the values of ACRM matrix can be omitted and will not be considered further for the analysis. With such an assumption it is allowed to reduce the number of values in the matrix without the necessity of determining an empirical acceptance threshold value similarly as in the CBPA algorithm. The correlation matrix obtained from the AUM for example 1 is shown in Table 3.

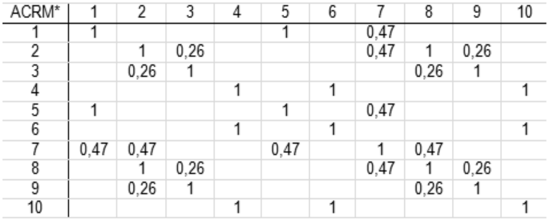
Table 3. The ACRM for example 1



ACRM	1	2	3	4	5	6	7	8	9	10
1	1	-0,07	-0,26	-0,45	1	-0,45	0,47	-0,07	-0,26	-0,45
2	-0,07	1	0,26	-0,45	-0,07	-0,45	0,47	1	0,26	-0,45
3	-0,26	0,26	1	0	-0,26	0	-0,26	0,26	1	0
4	-0,45	-0,45	0	1	-0,45	1	-0,45	-0,45	0	1
5	1	-0,07	-0,26	-0,45	1	-0,45	0,47	-0,07	-0,26	-0,45
6	-0,45	-0,45	0	1	-0,45	1	-0,45	-0,45	0	1
7	0,47	0,47	-0,26	-0,45	0,47	-0,45	1	0,47	-0,26	-0,45
8	-0,07	1	0,26	-0,45	-0,07	-0,45	0,47	1	0,26	-0,45
9	-0,26	0,26	1,0	0	-0,26	0	-0,26	0,26	1	0
10	-0,45	-0,45	0	1	-0,45	1	-0,45	-0,45	0	1

An undirected correlation graph is based on the ACRM. It is similar by its concept to the affinity graph presented in Figure 1. Vertexes define particular arguments once the edges define elements of the matrix with positive dependence. Table 4 presents all values of the ACRM and the corresponding graph is shown in Fig. 2. Considering vertexes with positive correlation coefficient only allowed to reduce the number of edges in correlation graph from 30 to14.

Table 4. Positive elements in the ACRM



ACRM*	1	2	3	4	5	6	7	8	9	10
1	1				1		0,47			
2		1	0,26				0,47	1	0,26	
3		0,26	1					0,26	1	
4				1		1				1
5	1				1		0,47			
6				1		1				1
7	0,47	0,47			0,47		1	0,47		
8		1	0,26				0,47	1	0,26	
9		0,26	1					0,26	1	
10				1		1				1

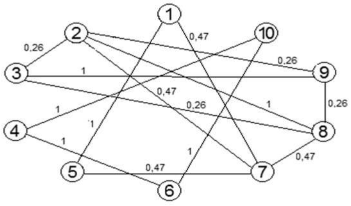


Fig. 2. Correlation graph obtained from ACRM for example 1

It can be noted that the number of edges in the graph depends directly on the characteristics of the transaction in the AUM. Table 5 shows the results of experiment during which the author determines the average number of edges depending on the degree of the AUM fulfillment assuming different approaches: the AAM, the ACRM and the ACM. Fulfillment ratio of a matrix (FRM) is defined as a ratio of positive elements number to the total number of all elements of a matrix. The range of FRM values is based on literature studies and it varies between 25% and 40%. The minimum values of acceptance threshold for the ACM is assumed as

0.2 and 0,4. Average number of graph's edges (NGE) is determined basing upon the number of 10000 different AUM matrixes generated for given dimensions and fulfillment ratio. Additionally, average number of disconnected graphs (NDG) obtained from n-sample has been determined in relation to the AUM fulfillment ratio. The average values has been obtained by multiple repetition of measurements.

Table 5. The average number of edges depending on the fulfillment ratio of the matrix in example 1

	AAM		ACRM		ACM (0,2)		ACM (0,4)	
FRM [%]	NGE	NDG [%]	NGE	NDG [%]	NGE	NDG [%]	NGE	NDG [%]
50	41,4	0,1	19,3	10,3	38,7	0,7	28,2	7,5
45	38,6	0,2	19,1	11,7	35,6	1,4	24,4	12,3
40	34,7	0,4	18,7	12,8	31,9	2,8	20,8	20,0
35	29,7	2,8	18,2	18,2	27,3	7,0	17,5	35,1
30	23,5	11,3	17,2	29,6	22,1	17,2	14,4	57,5
25	17,2	39,7	14,8	66,4	16,3	46,4	11,4	83,8
20	10,6	100,0	10,3	100,0	10,4	100,0	8,0	100,0

The results presented in table 5 show that the approach based on correlation is not influenced by decreasing number of graph's edges nor relative decrease of AUM fulfillment ratio. For the low value of fulfillment ratio (25%) the number of graph's edges for particular cases reaches similar values due to characteristic of transaction which appears in low number of accesses to attributes. For the higher values of fulfillment ratio (45-50%) of AAM and ACM matrixes with acceptance threshold set to 0,2 the number of graph's edges equals 45 (full graph). Further increasing of acceptance threshold to 0,4 and simultaneous decreasing the number of graph's edges to 25-30 still remains about 60% of full graph. The use of the Pearson correlation coefficient and considering only its positive values allows to reduce the number of edges in the graph to 20 and that gives about 50% reduction of graph's edges comparing to AAM and ACM. For the fulfillment ratio on the level of 30 to 40% the reduction of the edges is not so significant but still varies in the range of 20-40%.

Introduction of correlation as a measure of attributes dependence allows to avoid the problem of empirical selection of parameters as observed for ACM. Reduction of the number of edges can lead to disconnection of the graph. The possibility of disconnected graphs appearance depends on the fulfillment ratio AUM which has been presented in Fig. 2.

In all cases, the reduction of fulfillment ratio increases the probability of disconnected graphs occurrence. More than a half of all disconnected graph are obtained in the AUM fulfillment ratio in the range of 25-30% and 100%

disconnection is obtained for 20%. In case of disconnected graphs it is not possible to use graphical algorithm for the purpose of fragmentation process. For that reason modification of the GPA algorithm has been proposed.

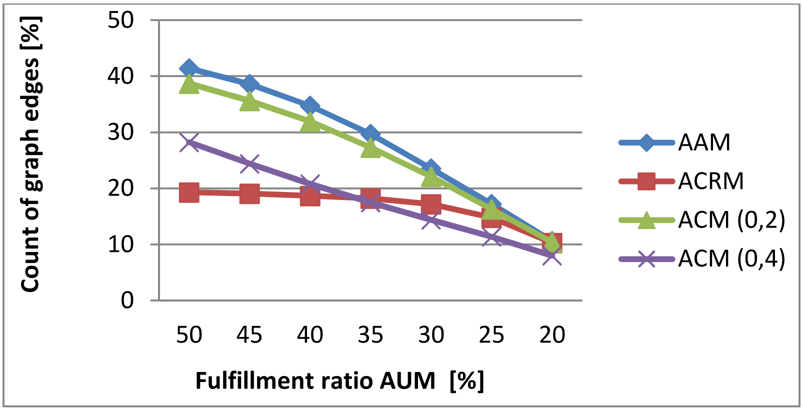


Fig. 3. Number of graph's edges for AAM, ACRM and ACM depending on the fulfillment ratio AUM

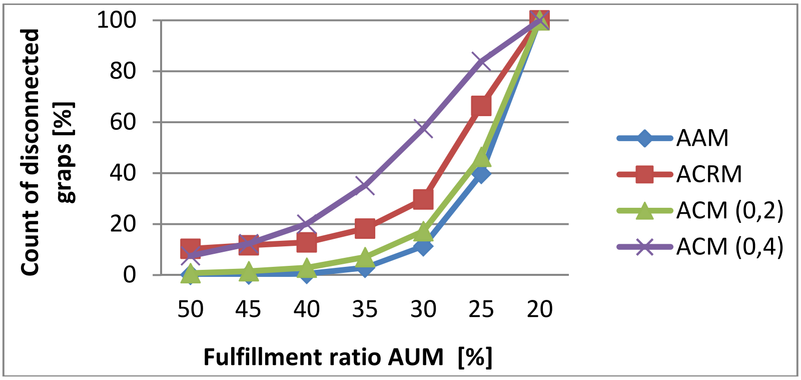


Fig. 4. Number of disconnected graphs depending on the fulfillment ratio AUM for n=10000

5. Two-phase based correlation partitioning algorithm

In order to solve the problem of disconnected correlation graph the author has proposed two-phase partitioning algorithm called Correlation-Based Partitioning Algorithm (CRBPA). The first phase called the initial phase uses disjoined sets that allows to split a correlation graph into independent sub-graphs. This phase is realized iteratively considering all of the graph's edges. Starting from the one argument sets graph's edge is being taken. Later on the sets joining operation (UnionSet) is being performed for attributes of considered edge. In effect the content of the sets is gradually increased and their number is being reduced simultaneously. In case of disconnected graph the end of this phase results in at least two sets and the most numerous of them contains k-attributes for $k < n$.

Computational complexity of the first phase equals $O(m)$ where m represents the number of edges found in the correlation graph. In the second phase independent sub-graphs are generated basing on the obtained sets and next the final partition is being made using classic GPA algorithm for each of mentioned sub-graphs.

Computational complexity of GPA algorithm equals $O(n^2)$ where n denotes number of attributes. Using of such algorithm and assuming its parallel execution allows to reduce computational complexity to $O(k^2)$ where n denotes the number of attributes of the most numerous set. Total complexity of proposed algorithm in case of disconnected graph equals $O(m+k^2)$ or $O(m+n^2)$ otherwise. Considering the example 1 which includes $n=10$ attributes first phases result in two sub-graphs consisted of the following elements (4,6,10) and (1,2,3,5,7,8,9). The number of graph's edges received from ACRM equals $m=14$ and the number of attributes for the most numerous set is $k=7$ so the computational complexity of proposed CRBPA algorithm is lower than in case of GPA algorithm and fulfills condition $O(m+k^2) < O(n^2)$.

6. Conclusions

This paper presents a vertical fragmentation problem in the distributed database design. The approach proposed utilize correlation of input data based on statistic of queries and the frequency of its occurrence. For that purpose new technique of edges reduction in correlation graph. The technique is characterized by low sensitivity to fulfillment ratio of input data (AUM). Potential occurrence of disconnected correlation graph depends mainly on characteristic of input data and it excludes using classical GPA algorithm for partitioning process. The solution of disconnected graph assumes modification of GPA algorithm with additional phase that is based upon disjoined sets structure. Simultaneously, computational complexity of proposed algorithm is decreased due to parallel processing of initially achieved sub-graphs. That way the complexity stays at the level of $O(m+k^2) < O(n^2)$ where $k < n$ in case of disconnected graphs.

In the future work author would like to focus on the application of correlation to estimating an influence of the number of transactions on the partitioning scheme. Another direction is to investigate the relation between significance level and graph's edge reduction.

References

1. Chakravathy S., Muthuraj J., Varadarajan R., Navathe S.: An Objective Function for Vertically Partitioning Relations in Distributed Databases and its Analysis, Distributed and Parallel Databases, Vol. 2, No. 1, pp. 183-207, San Diego, 1993
2. Cormen T.H., Leiserson C. E., Rivest R. L., Stein C.: Introduction in Algorithms, Third Edition, The MIT Press, USA, 2009
3. Du J., Alhajj R., Barker K.: Genetic algorithms based approach to database vertical partitioning, Journal of Intelligent Information Systems, Vol. 26 Issue 2, pp. 167 – 183, 2006
4. Du J., Barker K., Alhajj R.: *Attraction* - Global Affinity Measure for Database Vertical Partitioning, In proc. of ICWI, pp. 538-548, 2003
5. Goli M., Raolnkoohi R., Taghi S. M.: A new vertical fragmentation algorithm based on ant collective behavior in distributed database systems, *Knowledge and Information Systems,* Vol. 30, pp. 435 – 455, 2012
6. Hoffer A., Severance D.: The use of cluster analysis in Physical Database design, In. Proc. First Int. Conf. on very large Database, New York, 1975
7. Muthuray J., Chakravarthy S., Varadarajan R., Navathe S.: A Formal Approach to the Vertical Partitioning Problem in Distributed Database Design, |In Technical Report. CIS Dept, University of Florida, 1993
8. Navathe S.B., Ceri S., Wiederhold G., Dov J.: Vertical Partitioning Algorithms for Database Design, ACM Trans. On Database Systems, Vol. 9, No. 4, pp. 680-710, 1984
9. Ozsu M. T., Valduriez P.: Principles of Distributed Database Systems, Second Edition, Prentice Hall, 1999
10. Ra M., Navathe S. B.: Vertical partitioning and Database Design: A Graphical Algorithm, In Proceedings of the ACM SIGMOD International Conference on Management of Data, pp. 440-450, Portland 1989
11. Sobczyk M:, Statystyka. Aspekty praktyczne i teoretyczne, Wydawnictwo UMCS, Lublin, 2006

Abstract

The main purpose of this paper is to describe an approach of using data input correlation based on both statistic of queries and their frequency of occurrence within distributed databases. This approach is an alternative technique for reducing count of edges in the graph. It also defines a direction and strength of dependence between particular elements and is used for determination of partitioning criterion. This paper also presents a short characteristic of vertical fragmentation process based on statistic of queries and development of a graphical partitioning algorithm which enable to solve the problem of disconnected graph.

Streszczenie

Celem poniższej pracy jest przedstawienie podejścia dotyczącego zastosowania korelacji danych wejściowych opartych na statystyce zapytań i częstości ich wystąpienia w rozproszonych bazach danych. Podejście to jest alternatywną techniką redukcji liczby gałęzi w grafie podziału. Określa także kierunek i siłę zależności pomiędzy poszczególnymi elementami, która jest wykorzystywana przy ustalaniu kryterium podziału. Zawarto również krótką charakterystykę procesu fragmentacji pionowej opartej na statystyce zapytań oraz rozwinięcie algorytmu graficznego umożliwiającego rozwiązanie problemu niespójności grafu.

Słowa kluczowe: fragmentacja pionowa, rozproszone bazy danych, graficzny algorytm podziału, macierz i graf korelacji