International Journal of Engineering Research & Management Technology

ISSN: 2348-4039

September-2014 Volume-1, Issue-5

Email: editor@ijermt.org www.ijermt.org

A Review of Static and Dynamic Data Allocation Techniques for Distributed Database Design

Arjan Singh

Punjabi University, Patiala, Punjab, India

Abstract: Distributed database technology is one of the major technological developments during last 30 years in the field of computer science. Moreover, the need as well as the importance of distributed database systems has increased a lot with the invention of cloud computing. The design of a distributed DBMS plays a key role in its performance and data allocation is one of the major design issues. This paper is an attempt to review static as well as dynamic data allocation techniques available for distributed database design.

Keywords: Distributed Databases, Data Allocation, Static Allocation, Dynamic Allocation, Replicated Allocation, Non-Replicated Allocation.

1. Introduction

Design of distributed database involves two major design issues: fragmentation of the global database i.e. design of fragmentation schema and allocation of fragments over the communication networks i.e. design of allocation schema. Both fragmentation and allocation have an important role to play in the development of a cost effective distributed database system [39]. This paper concentrates on the problem of fragments allocation. The problem of fragments allocation can be categorized as: static allocation and dynamic allocation [39].

- *Static Allocation*: The access likelihood of the fragment required at different sites never change i.e. the access pattern are static.
- *Dynamic Allocation*: The access likelihood of the fragment required at different sites change over time i.e. the access pattern are dynamic.

2. Static Allocation

Allocation problem is first studied from file allocation point of view. Chu [18] was first to investigate it. He formulated the file allocation problem into a non-linear 0-1 programming problem which be solved with linear integer programming techniques. He has given an optimization model to minimize storage and transmission cost under certain constraints. Casey [13] examines the Chu's work and has given more stress on the cost of updating the data as well as cost of retrieving data. Eswaran [22] demonstrate that formulation given by Casey [13] is NP-Complete. Therefore, optimal solution can't be found in a reasonable amount of time. Therefore, he suggested heuristics rather than deterministic techniques.

Ceri et al. [14] investigated the data allocation problem for distributed database. They developed an optimization model for a non-replicated allocation of data in a linear 0-1 programming problem form. Horizontal fragmentation is used as an input for the allocation model. The main objective of the model is to minimize the total transaction processing cost. Decomposition heuristics are developed due to the complexity of the problem.

Wong and Katz [50] suggested local sufficiency as a measure of parallelism in a distributed database. They have suggested three different approaches for replication of fragments, each having different blend of cost and benefits.

Apers [6] has shown that the problem of fragment allocation and of file allocation are completely distinct problems in distributed database. Further, he proves that allocation of fragments is NP hard problem. Heuristics and optimization algorithms are proposed for non-replicated data allocation to minimize the total data transmission cost during the execution of a set of transactions. He has proved that the performance of optimal algorithm is superior to heuristic algorithm.

Chiu and Raghavenda [17] presented an allocation model to enhance the reliability of the system with the use of triple module redundancy (TMR) scheme. Triple module redundancy scheme ensures that each query request must be executed by at least three database servers on the communication network. The allocation problem is converted into a 0-1 integer programming problem and the Lagrangian relaxation technique has been applied to solve it. The goal of the allocation model is to discover the number of database servers and their respective location so that a distributed database system's operational cost can be minimized.

Lin et al. [31] proposed a simple polynomial time heuristic algorithm for data allocation to minimize the overall communication cost. The proposed algorithm considers physical network and transaction processing strategies for the allocation of data.

Corcoran and Hale [19] presented a genetic algorithm (GA) to allocate fragments in a distributed database system. Objective function for allocation is the total transmission cost of executing a set of queries over all the sites of the network. The transmission cost considers only the retrieval frequency of the different sites to all the fragments and moreover replication of fragments is not taken into consideration. Performance of the proposed GA is compared with the greedy heuristic and GA is found to have superior performance than that of greedy heuristics. They observed that the performance of GA does not degrade as the size of search space increases as compare to greedy approach.

March and Rho [36] extend the work of Cornell and Yu [20] to include data replication and concurrency control mechanism. They developed a comprehensive mathematical model for allocation of data as well as operations to nodes during the design process of a distributed database. Network communication information, local processing power of each site and data storage cost is considered which the development of the model. The mathematical formulation is solved using an iterative genetic.

March and Rho [37] analyze the performance of March and Rho [36] over various control parameters of genetic algorithm. Quality of the solutions provided by March and Rho [69] are analyzed over different pool size and crossover operators.

Lin and Orlowska [32] transformed the data allocation problem into an integer linear program. They investigate the data allocation problem to minimize the total communication cost of executing a set of transactions. Data replication and both read and update transaction are considered while allocating data over the communication network. Lin and Orlowska [32] have suggested that once the data allocation problem is converted into an integer linear program than the probability of finding a polynomial time bounded solution is quite high.

Daudpota [21] proposed a method to develop a model of data allocation for distributed database system. The method consists of five steps: collection of global relations; analysis of frequently asked queries; fixing objectives of data allocation; transformation of global relations into fragments; and allocation of fragmented relations to various sites of network. Based on these five steps, a model of data allocation is constructed. A heuristic algorithm named TGTF was derived for the transformation of global relation into fragments and their allocation in replicated manner.

Tamhankar and Ram [46] developed an integrated methodology for fragmentation and replicated allocation of data featuring concurrency control mechanism. The methodology is divided into four steps: primary distribution and three secondary distributions. First step, i.e. primary distribution, deals with the design objectives at application level. Output of the first step is a distribution matrix which designates the data distribution for each site. Second, third and fourth steps, i.e. secondary distribution, of the methodology deals with the response time, availability of the data and optimization of storage space respectively. A case study of manpower distribution in a distributed organization has been taken to show the performance of the proposed methodology. Barney and Low [8] extended the work of Tamhankar and Ram [46] for object-oriented databases by including the process workload estimation.

Barker and Bhar [7] proposed a graphical optimization technique for non-replicated allocation of fragments in distributed objectbase systems. A non-redundant cost model is developed for the allocation. To achieve a "near optimal" allocation of fragments, a heuristic algorithm is proposed by exchanging and/or moving fragments between every pair of site of the network.

Huang and Chen [28] proposed a comprehensive model for replicated allocation of fragments using the behavior of transactions in distributed databases. The cost model is developed based on the size of the fragment, retrieval and update behavior of all the transaction, and data transfer cost over the communication network. Two heuristic algorithms are proposed to minimize the total communication cost for the execution of a given set of transactions. Initially, all the fragments are allocated to each site that needs them. First, heuristic algorithm removes the replicas of fragments from the initial fragment allocation to minimize the communication cost. The second heuristic algorithm removes the replicas of fragments from the initial allocation based on the updates of each site during the execution of set of transactions. It is an attempt to find out "near-optimal" solutions. Results of the proposed heuristic algorithms are compared with Lin et al. [31].

So et al. [43] developed a probabilistic navigational model for data allocation in distributed hypermedia databases under the average response time constraint. A Hill-climbing heuristic algorithm is proposed for allocation of multimedia data object. It was observed that the proposed algorithm is a good choice for small problem size.

Ahmad et al. [5] proposed evolutionary algorithms for non-redundant data allocation in distributed database systems. Three different evolutionary algorithms and search based heuristic are introduced to solve the problem data allocation problem. The data transfer cost model has been developed using site-independent fragment dependency graph representation. The main objective is to obtain high quality solution (i.e. minimum data transfer cost) with fast turnaround time. Evolutionary algorithms are genetic algorithm (GA), simulated evolution algorithm and mean field annealing algorithm where as neighborhood search algorithm is based on search based heuristic. All the algorithms are compared on the quality of solution provided for data allocation and execution time. They have suggested genetic algorithm performs better than other approaches. Karlapalem et al. [51] also empirically evaluated the performance of these four algorithms for allocation of data in distributed multimedia databases.

Silva and Dissanayake [42] proposed an integrated solution of fragmentation and allocation problem in distributed database design. Two different algorithms are proposed for optimal design of distributed database. First algorithm fragments the global relation into vertical fragments based on the read and update requirements of different sites. The second algorithm allocates vertical fragments to various sites by using mimic cultural evolution technique.

Ma et al. [35] proposed a generalized framework for distribution design of higher-order data models. Fragmentation, allocation and replication are incorporated in proposed framework. An integrated heuristic approach is used for fragmentation and allocation. A cost model for query processing cost is constructed using query tree. Impact of horizontal as well vertical fragmentation on the query cost has been studied. The work is divided into three parts. The first part finds out the initial allocation of fragments. Initially, fragments are allocated either to the site, where they are most needed or to the sites that will have minimum transmission cost. The second part further evaluates the initial allocation and verifies that, whether further fragmentation improves that performance of data allocation or not. The third part puts a restriction on the query frequency selection predicates.

Hababeh et al. [26] designed an integrated methodology for clustering the sites of communication network and replicated data allocation in distributed database system for high performance computing. Sites of the communication network are grouped into different clusters based on some clustering decision value. Once the site of communication network is divided into different clusters then fragments are allocated to these clusters and their respective sites. The decision of allocating a fragment to a cluster is determined by the cost of allocating or the cost of not allocating it to that cluster. If the cost of allocating a fragment is more than the cost of not allocating then that fragment is allocated to the respective cluster otherwise not. The main idea is to reduce the overall communication and increase the availability as well reliability of the system.

Rahmani et al. [40] have attempted to further enhance the performance of Hababeh et al. [26] work by using GA. Roulette wheel method is used to select two individual from the population and single point crossover method is used for crossover operation. Objective function that has to be minimized is the overall communication cost which includes the cost of local retrievals, local updates, remote retrieval, remote updates and storage.

Adl and Rankoohi [4] proposed an algorithm named ACO-DAP which use ant colony optimization (ACO) for non-replicated allocation of data. The cost function that has to be minimize is the total data transmission cost under the storage capacity constrain. Three different versions of ACO-DAP are used to explore the performance of the proposed heuristic.

3. Dynamic Allocation

Apers [6] and Lin et al. [31] have also investigated the problem for dynamic environment. Apers [6] proposed heuristic algorithm for non-replicated data allocation for dynamic environment. Schedules of all the transactions are recomputed for the initial allocation to deal with the dynamic environment. The current allocation is change according to the cost of recomputed schedules. Schedules of all the transactions are again computed for the new allocation to check the performance.

Lin et al. [31] proposed a heuristic algorithm for dynamic data allocation to minimize the overall communication cost. The heuristic algorithm iteratively refines the initial allocation through local modification to minimize the overall communication. The local modification is the process of adding a copy of fragment to a site or by removing a copy of fragment from a site. Iteration process of the algorithm will stop if overall communication cost does not decrease. But in practice Lin et al. [31] suggested a fixed number of iteration.

Loukopoulos and Ahmad [34] proposed an adaptive genetic replication algorithm (AGRA) based on genetic algorithm for dynamic environment. The proposed AGRA adjusts itself to the changing environment and data allocation schema is modified to get better performance in new environment.

Rivera-Vega et al. [41] suggested a heuristic strategy to solve the problem of redistribution of data in distributed databases. They investigated the problem of redistribution of data from three different prospective: due to physical change, due to logical change and combination of both. An approximation strategy is proposed to solve the problem in linear polynomial time. The objective of the approximation strategy is to minimize the transfer time during redistribution of data.

Chaturvadi et al. [15] developed an adaptive method for allocation data in distributed database environment using machine learning approach. A method named MLTIF is proposed to minimize the data communication and update synchronization cost. MLTIF method automatically acquires information about the data usage of each site from the query history of the database. The acquired information is used to reschedule the allocation process.

Brunstroml et al. [11] proposed two heuristic algorithms (Simple Counter Algorithm and Load Sensitive Counter Algorithm) for dynamic data allocation in distributed database. Simple Counter Algorithm is proposed for reallocation of fragments due to change access pattern in distributed database environment. Load Sensitive Counter Algorithm is proposed to handle overloading of a site. Performance of the developed algorithms is studied on a local area network and a wide area network as well. It was observed that the Simple Counter Algorithm is performing 30 percent more than the static allocation.

Wolfson et al. [49] developed adaptive data replication algorithm (ADR). ADR changes the replicated allocation scheme on basis of current access behavior of the transaction in a distributed network system. ADR algorithm incorporates the storage capacity constraint i.e. the storage capacity of each site is taken care of while changing the existing allocation schema. Issues of failure and recovery of the distributed database system are addressed during the adaptive replication process.

Kazerouni and Karlapalem [30] presented a stepwise redesign of distributed relational databases. Redesign of distributed database is divided into two different phases: split phase and merge phase. The split phase of redesign approach breaks a fragment into smaller fragments on the basis of transactions access behavior towards the fragment. A merge phase of redesign approach merged a set of fragments together as they are accessed by a set of transactions collectively. The goal of the redesign is to reduce the irrelevant data access and to decrease the overall data transfer cost. The stepwise redesign is instigated either after fixed number of transactions or fixed time interval. Kazerouni and Karlapalem [30] approach work for non-replicated situations.

Chin [16] proposed an incremental growth framework. This framework is invoked in a situation when the performance of the fall below than a suitable threshold. The new servers are added incrementally in the system to enhance the performance at acceptable level. The data is reallocated in the system after the addition of each new server. Two heuristic algorithms are proposed for reallocation of data. Both the heuristic algorithms reallocate the data on the basis of objective cost function by using greedy, hill-climbing approach.

Mei et al. [38] proposed a model for dynamic file allocation schemes in a large-scale distributed file system to measure the assurance. The proposed model takes up the fragmentation, replication and security issues together during reallocation. An optimal dynamic distributed allocation algorithm is presented to make sure that reallocation of the fragments is transparent and automatic. The algorithm converges to optimal solution according to global access pattern. The aim of Maximizing file assurance defines optimality. The dynamic algorithm is developed to analyze high assurance, availability, performance, and scalability of distributed system.

Ulus and Uysal [47] proposed a heuristic algorithm named threshold algorithm for dynamic data allocation for non-replicated distributed database systems. The threshold algorithm is an extension of Simple Counter Algorithm given by Brunstroml et al. [11] to improve the performance of distributed database system. The threshold algorithm reallocates the fragments according to changing access pattern of distributed database queries. The threshold algorithm changes the ownership of fragment from one site to another site according to frequency of remote accesses.

Abiteboul et al. [3] provided a framework for distribution and replication of dynamic XML documents for distributed computing. A comprehensive cost model is developed for query evaluation to determine the cost of queries. A dynamic replicated algorithm is proposed to reduce the global costs of query execution as many sites can be collaborated to execute a query.

Buchholz and Buchholz [12] studied the problem of replica placement in adaptive content distribution network. A model is introduced for cost-quality-optimized content networking. The replica placement problem is converted into the knapsack problem. Plan ranking and greedy ranking heuristic algorithms are proposed to solve the problem. But no experimental work is done to prove the efficiency of the proposed algorithm.

Gog and Grebla [23] and Grebla and Gog [24] proposed an evolutionary fragmentation and allocation (EFA) algorithm for re-fragmentation and re-allocation in dynamic distributed database systems. The evolutionary fragmentation and allocation (EFA) algorithm re-fragment and re-allocate the fragments for redesign phase using genetic algorithm. The main objective the proposed allocation is to minimize the total data transmission cost.

Lin and Veeravalli [33] designed a dynamic object allocation and replication algorithm for distributed systems with centralized control. The new designed algorithm named as dynamic window mechanism (DWM). A mathematical cost model is developed to determine the costs involved in servicing a request. The dynamic window mechanism (DWM) algorithm transforms the existing allocation scheme to new allocation schema to adjust with the changing access behavior. The objective of the DWM is to minimize the total data transmission cost.

Gu et al. [25] investigated the problem of replicated data allocation for distributed databases with buffer constraints. The dynamic window mechanism (DWM) algorithm of Lin and Veeravalli [33] was used for dynamic replicated allocation of data. DMW algorithm makes a decision of saving a read request or not saving a read request depends on whether it can minimize the total cost of a request sequence or not. Two models have been developed for homogeneous and heterogeneous data objects. Three different strategies (No Replacement, Least Recently Used and Least Frequently Used) are employed on both developed models to check the performance of the proposed DWM algorithm.

Haddad and Charrada [27] proposed a dynamic replicated database placement on large scale system such as grid environment. A fragment reallocation algorithm is developed to change the allocation schema. The developed algorithm reallocates the fragments according to query patterns and the grid sites storage capabilities. The objective of this reallocation algorithm is to decrease the data transmission cost and to increase the resource utilization.

Basseda et al. [9] recommend a novel dynamic data allocation algorithm for non-replicated distributed database systems. Whenever, there is a change in the communication network due to a change in the pattern of data access, this novel algorithm, called Near Neighborhood Allocation (NNA), reallocated the date fragments. The network topology and routing for specifying destination is considered in the proposed methodology. The NNA algorithm moved the data fragment to a site which is the neighborhood of current owner of the fragment and on the path to the site having maximum access counter. The objective the proposed algorithm is to minimize the total data transfer cost during the execution of a set of transactions.

Abdul-Wahid et al. [2] proposed a bio-inspired replication approach for adaptive distributed databases. The new developed approach is based on swarm intelligence. A system of mobile agents, called Pogo ants, is designed for the management of partial replicated database system. This system creates the replicas of data fragments dynamically on the sites which are having frequent access over these fragments. These replicas of data fragments are removed after being ideal for a fixed period of time from the respective sites. The main goal of the proposed approach is to minimize the inter-site communication and response time.

Uysal and Ulus [48] formulated a Markov chain model of threshold algorithm [47] for non-replicated allocation of data in dynamic distributed database systems. Finite-state Markov chain is used for the threshold algorithm modeling. Change in access probability and change in threshold value are used to analyze the behavior of a fragment in the system. In this study, it was observed that the fragment tends to reside more at the site with higher access probability as the threshold value of the algorithm increases.

Tâmbulea and Horvat [44] proposed a model for dynamic redistribution of data fragments in distributed databases. The statistical information related to queries and their read /write requests for different fragments is gathered for a specific time period. Tâmbulea and Horvat-Petrescu [45] also developed a model to evaluate the queries in dynamic distributed database environment. A heuristic algorithm is proposed to redistribute data fragments using statistical information related to read /write requests of different queries. The objective of the redistribute process of data fragments is to minimize the size of data transfer.

Basseda and Rahgozar [10] proposed a fuzzy based approach (FNA) to improve the performance of Near Neighborhood Allocation (NNA) algorithm given by Basseda et al. [9]. FNA uses differentiation of access pattern for dynamic allocation of data. Basseda and Rahgozar [10] observed that the fuzzy based approach (FNA) is performed better NNA for larger networks.

Hauglid et al. [27] presented a decentralized dynamic fragmentation and replication management (DYFRAM) approach for dynamic distributed databases. DYFRAM is an integrated approach for fragmentation, replication and reallocation. DYFRAM uses recent access history of different transaction in the communication network. The objective of DYFRAM approach is to maximize the number of local accesses as compare to the number of accesses from remote sites.

Abdalla [1] presented a data reallocation model for replicated as well as non-replicated distributed database systems. A heuristic algorithm is proposed for reallocation process. The algorithm reallocates the fragments under storage capacity and fragment limit constraints. The objective is to minimize the communication cost and response time.

4. Conclusion

Data allocation problem in distributed database design is NP-hard. There are two methods to handle NP-hard problems: heuristic approach and optimization algorithm. A heuristic approach may generate only a good approximation solution to the problem. Heuristic approaches are generally polynomial time bounded with respect to the input size. An optimization algorithm may always generate the optimal solution to the problem. But optimization algorithms are exponential time bounded with respect to the input size. Most of the researchers have proposed either heuristic algorithms or optimization algorithms to solve the data allocation problem in distributed database system.

References

1. H.I. Abdalla, "*A New Data Re-Allocation Model for Distributed Database Systems*", International Journal of Database Theory & Application, Vol. 5, No. 2, pp. 45-60, 2012.
2. S. Abdul-Wahid, R. Andonie, J. Lemley, J. Schwing, and J. Widger, "*Adaptive Distributed Database Replication Through Colonies of Pogo Ants*", In Proceedings of IEEE International Parallel and Distributed Processing Symposium (IPDPS 2007), pp. 1-8, 26-30 March 2007.
3. S. Abiteboul, A. Bonifati, G. Cobena, I. Manolescu, and T. Milo, "*Dynamic XML Documents with Distribution and Replication*," In Proceedings of ACM SIGMOD International Conference on Management of Data, pp. 527-538, 9-12 June 2003.
4. R.K. Adl and S.M.T.R. Rankoohi, "*A New Ant Colony Optimization Based Algorithm for Data Allocation Problem in Distributed Databases*", Knowledge and Information Systems, Vol. 20, No. 3, pp 349-373, 2009.
5. I. Ahmad, K. Karlapalem, Y.K. Kwok and S.K. So, "*Evolutionary Algorithms for Allocating Data in Distributed Database Systems*", Distributed Parallel Databases, Vol. 11, No. 1, pp. 5–32, 2002.
6. P.M.G. Apers, "*Data Allocation in Distributed Database Systems*", ACM Transaction on Database Systems, Vol. 13, No. 3, pp. 263-304, 1988.
7. K. Barker and S. Bhar , "*A Graphical Approach to Allocating Class Fragments in Distributed Objectbase Systems*", Distributed and Parallel Databases, Vol. 10, No. 3, pp 207-239, 2001.
8. H.T. Barney and G.C. Low, "*Object Allocation with Replication in Distributed Systems*", World Academy of Science, Engineering and Technology, Vol. 24, pp. 496-504, 2008.
9. R. Basseda, S. Tasharofi and M. Rahgozar, "*Near Neighborhood Allocation (NNA): A Novel Dynamic Data Allocation Algorithm in DDB*", In Proceedings of 11th Computer Society of Iran Computer Conference (CSICC2006), 2006.
10. R. Basseda and M. Rahgozar, "*A novel fuzzy approach to improve near neighborhood allocation algorithm in DDB*", In Proceedings of IEEE/ACS International Conference on Computer Systems and Applications, pp. 571-578, 10-13 May 2009.
11. A. Brunstrom, S.T. Leutenegger and R. Simha, "*Experimental Evaluation of Dynamic Data Allocation Strategies in a Distributed Database with changing Workloads*", In Proceedings of the fourth international conference on Information and knowledge management, pp. 395-402, 1995.
12. S. Buchholz and T. Buchholz, "*Replica Placement in Adaptive Content Distribution Networks*", In Proceedings of ACM Symp. Applied Computing (SAC'04), pp. 1705-1710, 14-17 March 2004.
13. R. G. Casey, "*Allocation of Copies of a File in an Information Network*", In Proceedings of AFIPC 1972 SJCC, Vol. 40, pp. 617-625, 16-18 May 1972.
14. S. Ceri, S. Navathe, and G. Weiderhold, "*Distribution Design of Logical Database Schemas*," IEEE Transactions on Software Engineering, Vol. 9, pp. 487-563, 1983.
15. A. Chaturvedi, A. Choubey, and J. Roan, "*Scheduling the Allocation of Data Fragments in a Distributed Database Environment: A Machine Learning Approach*", IEEE Transactions on Engineering and Management, Vol. 41, No. 2, pp. 194-207, 1994.
16. A.G. Chin, "*Incremental Data Allocation and Reallocation in Distributed Database Systems*", Journal of Database Management, Vol. 12, No. 1, pp. 35-45, 2001.

17. G. Chiu and C.S. Raghavendra, "*A Model for Optimal Database Allocation in Distributed Computing Systems*", In Proceedings of IEEE INFOCOM 1990, Vol. 3, pp. 827-833, 3-7 June 1990.
18. W.W. Chu, "*Optimal File Allocation in Multiple Computer Systems*", IEEE Transactions on Computers, Vol. C-18, No. 10, pp. 885-889, 1969.
19. A. Corcoran and J. Hale, "*A Genetic Algorithm for Fragment Allocation in a Distributed Database System*", In Proceedings of ACM Symp. Applied Computing, pp. 247-250, 1994.
20. D.W. Cornell and P.S. Yu, "*On Optimal Site Assignment for Relations in the Distributed Database Environment*", IEEE Transactions on Software Engineering, Vol. 15, No. 8, pp. 1004-1009, 1989.
21. N.H. Daudpota, "*Five Steps to Construct a Model of Data Allocation for Distributed Database Systems*", Journal of Intelligent Information Systems, Vol. 11, No. 2, pp. 153-168, 1998.
22. K.P. Eswaran, "*Placement of Records in a File and File Allocation in a Computer Network*", In Proceedings of the IFIP Congress on Information Processing, pp. 304-307, 1974.
23. A. Gog and H.A Grebla, "*Evolutionary Tuning for Distributed Database Performance*", In Proceedings of 4th International Symposium on Parallel and Distributed Computing (ISPDC 2005), pp. 275-281, 4-6 July 2005.
24. H. Grebla and A. Gog, "*Redesign Based Optimization for Distributed Databases*", Studia Univ. Babes-Bolyai, Informatica, Vol. L , No. 1, pp. 97-104, 2005.
25. X. Gu, W. Lin and B. Veeravalli, "*Practically Realizable Efficient Data Allocation and Replication Strategies for Distributed Databases with Buffer Constraints*", IEEE Transactions on Parallel and Distributed Systems, Vol. 17, No. 9, pp. 1001-1013, 2006.
26. I.O. Hababeh, M. Ramachandran and N. Bowring, "*A high-performance computing method for data allocation in distributed database systems*", The Journal of Supercomputing, Vol. 39, No. 1, pp. 3-18, 2007.
27. C. Haddad and F. B. Charrada, "*Database Placement on Large-scale Systems*", International Journal of Information Technology, Vol. 3, No. 4, pp. 251-256, 2006.
28. Y.-F. Huang and J.-H. Chen, "*Fragment Allocation in Distributed Database Design*", Journal of Information Science and Engineering, Vol. 17, pp. 491- 506, 2001.
29. K. Karlapalem, I. Ahmad, S.-K. So and Y. Kwok, "*Empirical Evaluation of data allocation algorithms for distributed multimedia database systems*", In Proceedings of The Twenty-First Annual International Computer Software and Applications Conference (COMPSAC '97), pp. 296-301, 13-15 August 1997.
30. L. Kazerouni and K. Karlapalem, "*Stepwise Redesign of Distributed Relational Databases*" Technical Report, HKUST-CS97-12, September 1997.
31. X. Lin, M. Orlowska and Y. Zhang, "*On Data Allocation with the Minimum Overall Communication Costs In Distributed Database Design*" In Proceedings of ICCI'93: 5th International Conference on Computing and Information, pp. 539-544, 27-29 May 1993.
32. X. Lin and M. Orlowska, "*An Integer linear Programming Approach to data Allocation with the Minimum Total Communication Cost in Distributed Database Systems*", Information Sciences, Vol. 85, No. 1-3, pp. 1–10, 1995.
33. W. Lin and B. Veeravalli, "*A dynamic Object Allocation and Replication Algorithm for Distributed systems with Centralized Control*," International Journal of Computers and Applications, Vol. 28, No. 1, pp. 26-34, 2006.
34. T. Loukopoulos and I. Ahmad, "*Static and Adaptive Distributed Data Replication using Genetic Algorithms*", Journal of Parallel and Distributed Computing, Vol. 64, No. 11, pp. 1270-1285, 2004.
35. H. Ma, K.-D. Schewe and Q. Wang, "*Distribution Design for Higher-order Data Models*", Data & Knowledge Engineering, Vol. 60, No. 2, pp. 400-434, 2007.
36. S.T. March and S. Rho, "*Allocating Data and Operations to Nodes in Distributed Database Design*", IEEE Transactions on Knowledge and Data Engineering, Vol. 7, No. 2, pp. 305-317, 1995.
37. S.T. March and S. Rho, "*Characterization and Analysis of a Nested Genetic Algorithm for Distributed Database Design*," Seoul Journal of Business, Vol. 2, No. 1, pp. 85-121, 1996.

38. A. Mei, L. Mancini, and S. Jajodia, "*Secure Dynamic Fragment and Replica Allocation in Large-Scale Distributed File Systems*," IEEE Transactions on Parallel and Distributed Systems, Vol. 14, No. 9, pp. 885-896, 2003.
39. M. Ozsu and P. Valduriez, "*Principles of Distributed Database Systems*", Prentice Hall, 2nd Edition, 2002.
40. S. Rahmani, V. Torkzaban and A.T. Haghighat, "*A New Method of Genetic Algorithm for Data allocation in Distributed Database Systems*", In Proceedings of IEEE 1st International Workshop on Education Technology and Computer Science, pp. 1037-1041, 2009.
41. P.I. Rivera-Vega, R. Varadarajan and S.B. Navathe, "*Scheduling Data Redistribution in Distributed Databases*", In Proceedings of 6th International Conference on Data Engineering, pp.166-173, 5-9 February 1990.
42. D.P.D. Silva and M. Dissanayake, "*Optimal Design of Distributed Databases*", In Proceedings of International Conference on Information and Automation (ICIA 2006), pp. 1-6, 15-17 December 2006.
43. S. So, I. Ahmad and K. Karlapalem, "*A data allocation algorithm for distributed hypermedia documents*", In Proceedings of 17th IEEE Symposium on Reliable Distributed Systems, pp. 473-478, 20-23 October 1998.
44. L. Tâmbulea and M. Horvat, "*Dynamic Distribution Model in Distributed Database*", Int. J. of Computers, Communications & Control, Vol. 3, pp. 512-515, 2008.
45. L. Tâmbulea and M. Horvat-Petrescu, "*Redistributing Fragments into a Distributed Database*", Int. J. of Computers, Communications & Control, Vol. 3, No. 4, pp. 384-394, 2008.
46. A.M. Tamhankar and S. Ram, "*Database Fragmentation and Allocation: An Integrated Methodology and Case Study*", IEEE Transactions on Systems, Man and Cybernetics-Part A: System and Humans, Vol. 28, No. 3, pp. 288-305, 1998.
47. T. Ulus and M. Uysal, "*Heuristic Approach to Dynamic Data Allocation in Distributed Database Systems*", Pakistan Journal of Information and Technology, Vol. 2, No. 3, pp. 231-239, 2003.
48. M. Uysal and T. Ulus, "*A Threshold Based Dynamic Data Allocation Algorithm- A Markove Chain Model Approach*", Journal of Applied Science Vol. 7, No. 2, pp. 165-174, 2007.
49. O. Wolfson, S. Jajodia, and Y. Huang, "*An Adaptive Data Replication Algorithm*", ACM Transactions on Database Systems, Vol. 22, No. 2, pp. 255-314, 1997.
50. E. Wong, R.H. Katz, "*Distributing a Database for Parallelism*" In Proceedings of the 1983 ACM SIGMOD International Conference on Management of Data (SIGMOD '83), pp. 23-29, 1983.