A Comprehensive Review on Query Optimization for Distributed Databases

Ravneet Kaur[1], Er. Ajay Sharma[2]

[1]Ravneet Kaur, Computer Science Engineering Department, Amritsar College of Engineering and Technology, Amritsar
[2]Er.Ajay Sharma, Computer Science Engineering Department, Amritsar College of Engineering and Technology, Amritsar

---

Abstract - *With an advancement in multimedia technology, databases are turn out be primary source of information. Since the size of database is large, so, we generally use distributed database to provide information to users. However, query optimization in distributed database is still a challenging issue. Many techniques have been proposed so far to optimize the distributed queries. However, the majority of existing techniques suffer from the effect of query cost and communication overheads are ignored in most of existing research on distributed databases. The use of multi-objective optimization is ignored by most of existing researchers. The use of multi-objective non-dominated sorting genetic algorithm to reduce query cost is also neglected in existing literature. Thus, there is a need to design a novel distributed query optimization technique.*

Key Words**: Distributed, Databases, Query optimization, Genetic algorithms.**

1. INTRODUCTION

Due to increased network traffic along with decreased efficiency, partitioning of information at various locations has become significant which usually lead to the development of a group of related databases that are known as the distributed database, where each location has their own local processing and storage abilities. This enhances effectiveness, stability, availability, and also modularity compared to that in traditional centralized databases system. The data in distributed database may be replicated at different locations in accordance to distribution allocation plan. Therefore to provide answers to a query, association among different sites is required. Thus optimizing query has been a crucial subject for the Distributed Database Management System (DBMS). The progress in computer hardware, software, networks, storage and protocols have altered the view of business demands by making the processing of distributed database, feasible and useful option.

1.1 Distributed database system architecture

Distributed database can be considered as a set of many databases related logically that are dispersed over a computer network. Distributed Database Management System (DDBMS) belongs to the class of application software that handles distributed database and offers transparent access mechanism to several users across multiple sites by incorporating parallelism as well as modularity. Based on the types of DDBMS software installed on the various sites of distributed databases, it can be categorized as Homogenous DDBMS or Heterogeneous DDBMS.

- **Homogeneous Distributed System**-The data is distributed but all servers run the same Database Management System software in homogeneous distributed database.
- **Heterogeneous Distributed System**–several sites work under the control of different database management systems in heterogeneous distributed database. They are related by some means to allow access to data from several sites.

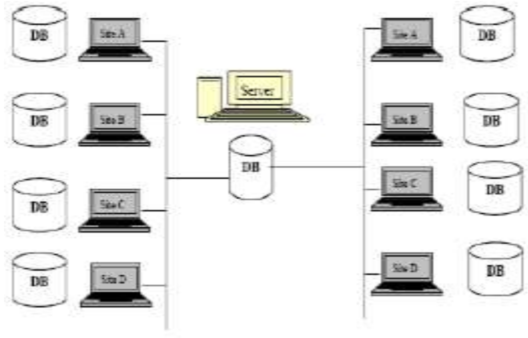


Fig -1: Distributed database systems Architecture

1.1.1 Distributed database storage

Data can be stored in following two ways on different sites:

Replication

In this technique, the whole relation is stored excessively on two or more than two sites. In case whole database is accessible at all the sites, it is a completely repetitive database. Therefore in replication, systems retain copies of data. This enhances the accessibility of data at several sites. Moreover query requests can be processed in side by side.

However, it has certain weaknesses as well. Data is required to be continuously updated. In case of any change done at single site is required to be saved at each and every other site or else it may result in irregularity and lots of overhead. Concurrency control also becomes more complex, since, now concurrent access needs to be examined over multiple sites.

Fragmentation

The relations are partitioned into fragments (smaller parts) and every one of these fragments are saved at multiple sites where ever needed. It needs to be ensured that the smaller units are so that these can be anyway used to rebuild an original table, which means that there must be not any loss of information.

This technique does not produce replicates of data, uniformity or reliability isn't a problem. Fragmentation can be carried out in following three different ways:

- **Vertical fragmentation:** This splits a relation vertically with columns. To add the primary key of the relation in each of the vertical fragment is essential such that the whole relation could be rebuild if required.
- **Horizontal fragmentation:** This approach splits a relation 'horizontally' by simply choosing the appropriate rows and all these smaller units can be allotted to multiple sites within distributed database system.
- **Mixed fragmentation:** The original relation could be rebuild by using operations named natural and union join in correct sequence in this approach.

1.1.2 Advantages

- **Modular Development** –Local information and new computers are added to the new locations or sites in distributed databases. Then finally these sites are connected to the distributed system in a way that it doesn't interrupt existing functioning.
- **Better Response** – User requests can be fulfilled from local data available there, in case the information is distributed in a relevant manner, which result in quick response.
- **More Reliable** – In case any part or component doesn't work, then also the system continues to work whether at a low performance in distributed systems, which makes it more consistent.
- **Lower Communication Cost** – Where data is frequently used, if it is stored locally over there, then the costs required for handling of data can be reduced.

In distributed systems it is easier to keep errors local rather than the whole organization being affected.

1.1.3 Disadvantages

- **Results in Data integrity** – Necessity for data to de updated at several sites may result in the problem of data integrity.

- **Requirement of expensive and complex software** – In order to maintain co-operation and transparency of data over multiple sites, expensive software is required by distributed systems.

- **Processing overhead** – Even basic operations may need a Huge number of communications and also extra computations may be required even to perform simple operations in order to maintain regularity in data across the several sites.

- **Leads to Overheads for not proper distribution of data** – If data is not distributed properly then it may result into very slow respond to all the requests of user.

1.2 QUERY PLAN OPTIMIZER IN DISTRIBUTED DATABASE MANAGEMENT SYSTEM

The factors defined for optimizer, largely affects it. Some of these factors are reliable on the structural framework of DDBMS and some of them are dependent on its procedural behaviour. These factors are: (i) Allocation of data at multiple sites impacting the actual Data Transmission Cost (ii) Shape of Join Query Graph (iii) Order of Join Operation (v) Arrangement of Query Operators Among all these factors, the most important factor researched upon by applying appropriate search strategies is the Order of Join Operation of the relations involved in the query. Many search strategies have been researched upon and the search is still going on for finding a appropriate Search Strategy for efficiently optimizing huge join queries. The review of literature focuses on the Search Strategies implemented till now as query optimizer in Distributed Database Management System.

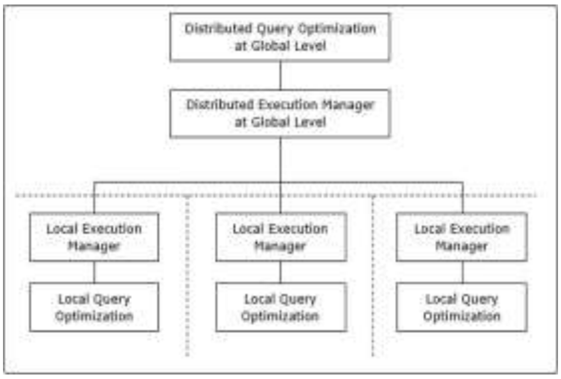


Fig -2: Architecture of Distributed Query Optimizer

1.2.1 Query Processing in Distributed Database

Standard method of processing query in distributed system is parsing a particular Structured query language statement and creating a query having representation resembling a relational calculus-like logical form, and then to create the query optimizer, so as to produce a query plan. This plan is then provided to an execution engine where it is executed directly, with very little or sometime no decision – making of execution time.

Distributed query optimizer first maps the distributed database with the SQL query defined on the global relations and then converts them into a string of relational operations that are defined on fragmented relations. Query processing in distributed database consists of four generic phases where the input is relational calculus of the distributed query expressed on global conceptual schema.

- **Decomposition of query**: Distributed query is decomposed in the form of algebraic query. Actual task is to re-write query in normalized form, analyze it semantically (based on meanings), simplify it by removing the redundant predicates and then restructure it into an algebraic query.
- **Data Localization**: Here the input is algebraic query on global relations where its data is localized according to the information available on distributed relations. Two steps are involved in creating fragmented query in this phase. In the first step a construction program is used to map the input query with the fragmented data and replaces each relation by its fragments to create a fragmented query. The second step involves simplifying this query and restructuring into another appropriate query.
- **Global Query Optimization**: The main task of this phase is to generate an optimal query execution plan with proper sequencing of relational operators in the fragment query which greatly minimizes the defined objective cost function.
- **Local Query Optimization**: Here the optimal Join Order received which is output of above phase is executed locally at multiple sites using the Local Conceptual Schema of the Distributed Database.

1.3 ARTIFICIAL BEE COLONY OPTIMIZATION

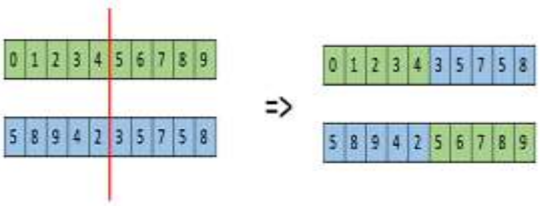
ABC algorithm is latest evolutionary approach based on honey bee swarm's intelligent behaviour. The Artificial Bee Colony technique is inspired through the natural food searching behaviour of the honey bee. ACO is regarded as the most effective strands of Swarm Intelligence because of its highly effective as well as appropriate decision making ability that is dependent on the cooperative behaviour of its agents i.e., ants. ABC algorithm still has few weaknesses, which are it is good at searching or locating but it is poor at manipulating. Its convergencing speed is also a problem in some cases. For these insufficiencies, a Novel Artificial Bee Colony algorithm (NABC) is proposed to optimize problems numerically so as to increase its ability of manipulating a problem by including the existing best solution into the process of searching. Basic task of this thesis is to create query optimizer for distributed database which make use of the positive characteristics of ACO Algorithm combined with another algorithm to optimize big queries in distributed systems. ACO is entirely combined with another algorithm, in order to overcome the insufficiencies of ACO and improving its processing time.

1.4 GENETIC OPERATORS

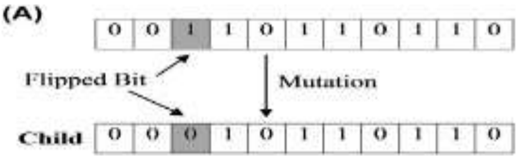
Genetic operators are widely used to generate and maintain genetic-diversity, merge existing solutions into completely new solutions and select between solutions. The Genetic Algorithm (GA) is used to execute large queries with lesser joins. Initially Chromosomes population is created by making use of Genetic Algorithm, where each of the chromosomes is used to present a query plan. Every chromosome is made up of genes where the site of relation is represented by every chromosome's gene. Gene's value depicts the location in which the particular relation is situated is depicted by a gene's value. Genetic operators are as follows:

Selection: Individuals are selected for reproduction depending on their value of fitness by using this operator.

Crossover: Two solutions are combined to produce the one best solution by using crossover operator.



Mutation: A new solution is created by altering existing solution's gene values.



2. LITERATURE REVIEW

Aponso et al. (2017) [20] proposed a Genetic algorithm based method to optimize join query in distributed database. Initial population is created arbitrarily by applying GA. The genetic operation is performed on every phase, and this changes the population. This basically decreases the average cost of each individual. Then the query is executed, at the moment the best or the most optimal plan is achieved, by forwarding it to the database engine. For optimizing the execution plans of join query as well as to reduce execution time and cost, this approach can be used. This also has weaknesses such as problem of local optima and low-speed convergence to obtain an optimal solution.

Dermino et al. (2016) [21] It involves ideal assignment of generation sources in distributed system as well as reclose in instantaneous manner and produces IHS (Improved Harmony Search) algorithm in order to find solution. To achieve it, 2 primary control parameters are attuned so as to obtain best answer from a simple Harmony Search algorithm. There are two parts of proposed multi objective function: one is enhancing consistency values and other is decreasing loss of power. Depending on user requirements and their reaction to temporary as well as long-lasting problems, consistency indices are designated. Next it uses four consistency indices; they are: cost of unsupplied energy, system average interruption duration index (SAIDI), system average interruption frequency index (SAIFI) and momentary average interruption frequency index.

Kumar et al. (2015) [22] proposed ACO algorithm reliable on technique named stochastic search. Movements of ant are modified from one relation to other. Movement of ant in the following iterations towards the optimization parameters from one particular relation will still be in different clockwise manner so as to show all the relations. The above procedure repeats for a pre-defined quantity of duplications until the best query plans are produced as outcome. For relations in large numbers, ACO algorithm creates best query plan. But it has weaknesses such as high respond time and also optimization overheads are higher.

Ban et al. (2015) [23] has combined GA and Min-Max Ant System to propose query optimization algorithm so as to enhance the efficiency of query. Initially, number of query processing plans is generated that are highly optimal, by using high speed convergence behaviour of Genetic Algorithm. They are then converted into the initial pheromone of Min-Max Ant System which helps the ants to converge faster in finding the relatively optimal query execution plan by increasing their convergence-speed. The superiority of parallelism is highly shown by Corresponding GA and Max-Min Ant System in case of large number of relations. In comparison to other algorithms, this execution plan has less search time and also time of query execution is reduced in optimal plan generated. This has weaknesses that computation time is increased due to parallel processing of two algorithms.

Mishra et al (2014) [24] for evaluating QEP (Query Execution Plans), technique named particle swarm optimization was developed. In order to arrange a swarm that is migrating across the search space for finding the best answer, a set of particles or social agents are used. Particles are denoted by the appropriate query plans based on relation schemes. The entire query processing plan may be created by optimizing entire query at the time of compilation. By assuming the arbitrary population of random speed and variables, function of evaluating particles that is indicated by the query plans in the relation, can de estimated. By doing this less search space is used and also the particle boundaries are towards the best place globally. But it has problem of local optima.

Golshanara et al. (2014) [25] Distributed database systems offer completely new data storage and processing technology for today's decentralized organizations. Because of large search space of corresponding plans acquired by its distribution, query optimization (producing optimal execution plan) has been more challenging in these systems. Use of stochastic-based algorithms has attracted many researchers due to computational intractability in generating an optimal execution plan. Golshanara proposed multi-colony ant algorithm, for the first time, so as to optimize join queries in distributed systems in which tables can be duplicated but they cannot be partitioned or fragmented. In this planned algorithm, 4 kinds of ants cooperate to generate an execution plan. Therefore each of the iteration has four ant colonies. In order to find the optimal plan, each ant performs decision-making. Two types of cost prototypes centred on total time as well as response time are utilized for the evaluation of the quality of the generated plan.

Adimi et al.(2014)[26] proposed Particle Swarm Optimization (PSO) fuzzy multiple objective approach to generate optimum localizing and value arrangement of Unified Power Flow Controller (UPFC) within a control system for a large time. Placing Unified Power Flow Controller in transmission area has resulted in decreasing the total cost of generation, because of its capability to alter network's power flow pattern. Unified power flow controller can be utilised to lessen or even remove congestion in the transmission network. Another problem is violation of voltage in power system which may lead the problem of solving optimum power flow. This destruction of voltage can also be lessened by properly applying Unified Power Flow Controller in a transmission system. These objectives are taken into account concurrently in integrated objective function for the proposed optimization algorithm.

Xiang H et al. (2013) [27] describes about the DDMS and processing of query in distributed system. It also mention requirement distributing huge scientific datasets distribution such as the Sloan Digital Sky Survey. Basic aim of the paper is to completely examine cross-joins query in distributed database through a scientific database that is distributed heterogeneously.

Dokeroglu et al. (2012) [28] proposed a technique focused on collaborative behaviour of agents which results when corresponding individuals interact locally with each other. This has high speed of convergence towards obtaining optimal solution and also execution time. This suffers from problem of low variety population as well as local optima.

Chen et al. (2011) [29] In order to optimize huge query, Chen developed Simulated Annealing method that is based On Graph-Based Approach. Hence the problems which have great solution space are optimized by using this probabilistic algorithm. Every solution depicts a state in the search space, and cost is associated with every state. The aim is to search minimum cost space. Linear recursion technique represents Simulated Annealing used for optimizing difficult queries. Relational operator's algebraic structure defines the state space. Various algebraic expressions for solving query are represented by a particular state.

Table -1: Comparison Table

Ref. no.	Year	Technique	Advantages	Disadvantages
Aponso et al[20]	2017	Using certain kind of evolutionary algorithm that uses biological techniques.	•Lesser execution time •Processor cost	• more delay •Suffering from the problem of local optimum •Less convergence speed towards an optimum solution
Kumar et al[22]	2015	Proposing an ant colony algorithm for query optimization.	•Improving the average optimization •Quality of query plan	•High overhead •High response time
Ban et al[23]	2015	Proposing a max-min ant system and	•High efficiency •High	•Both algorithms are required to be computed in

		genetic algorithm in parallel.	effectiveness	parallel •Enhances the cost of computation
Mishra et al[24]	2014	Evaluating query execution plans are evaluated by making use of particle swarm optimization and join operation.	•Reduces the computation al complexity •improves execution plan	•High response time •High execution cost
Golshanara et al[25]	2014	For optimizing join queries, a multi-colony ant algorithm is proposed.	•Decreasing total time •Increasing a convergence speed	•Worse performance for a smaller query •increased the time of execution •Suffer from Local optimum
Dokeroglu et al[28]	2012	Focusing on the cooperative behaviour which results when individuals interact locally with one another.	•High convergence Speed •Execution time	•Low variety population •Drop to local optimal
Chen et al[29]	2011	Proposing a technique in which research is done arbitrarily that exploits an similarity between the way in which metal first cools down and then freeze into energy crystalline structure and the exploration for a minimum is searched in a more general system required for computation.	•Low cost •Better quality query •Avoiding getting stuck at the local minimal	•Dependent on the cost function and the neighbour function used in it •Deficiency resulted by the transformation of the state •limiting the search

3. CONCLUSIONS

Extensive review has been considered in this paper on query optimization techniques. It has been found that the data in distributed databases may be repeated at many sites according to a distributed allocation plan. As a result, an association between various sites is needed in response to the user's query. Therefore, query optimization plays a significant role to handle distributed queries. Due to the increasing number of joins and the number of query execution plans, the query optimization problem is an NP-hard problem. Therefore, many heuristic and metaheuristic approaches have been proposed to solve this problem. The comparative analysis has demonstrated that the development of optimistic query optimizer is still an open area of research.

In this paper, no technique have been designed, therefore, in near future, to overcome the issues associated with the existing techniques, a new multi-objective non-dominated sorting genetic algorithm based query optimization technique will be proposed. The effect of query cost and communication overheads will also be considered. The use of non-dominated sorting genetic algorithm can find optimistic query in order to reduce the query cost and the proposed technique will be compared to existing techniques based upon certain performance metrics.

REFERENCES

[1] Shiwu Y, Peng Y. An Optimization for Distributed Database Multi-join Query Based on Improved Genetic Algorithm. DEStech Trans Comput Sci Eng. Iceiti 2017https://doi:10.12783/dtcse/iceiti2017/18832

[2] Tewari P, Chande SV. Query optimization strategies in distributed databases. Int JAdv Eng Sci. 2013; 3(3):23-29.

[3] Azarbad M, Ebrahimzadeh A, Babajani-Feremi A. Brain tissue segmentation using an unsupervised clustering technique based on PSO algorithm. Paper presented at: 2010 17th Iranian Conference of Biomedical Engineering (ICBME); 2011; Isfahan, Iran.

[4] Butey P, Meshram S, Sonolikar R. Query optimization by genetic algorithm. J In Techno Eng. 2012; 3(1):44-51.

[5] Manafi H, Ghadimi N, Ojaroudi M, Farhadi P. Optimal placement of distributed generations in radial distribution systems using various PSO and DE algorithms. Elektronika IrElektrotechnika. 2013; 19(10):53-57.

[6] Padia S, Khulge S, Gupta A, KhadilikarP. Query optimization strategies in distributed databases. Int JComput SciInf Technol. 2015; 6(5):4228-4234.

[7] Matysiak M. Efficient optimization of large join queries using tabu search. Inform Sci. 1995; 83(1-2):77-88.

[8] Morsali R, Ghadimi N, Karimi M, Mohajeryami S. Solving a novel multi objective placement problem of recloser and distributed generation sources in simultaneous

mode by improved harmony search algorithm. Complexity. 2015; 21(1):328-339.

[9] Shailesh Pandey and Sandeep Kumar, “Enhanced Artificial Bee Colony Algorithm and Its Application to Travelling Salesman Problem,” HCTL Open International Journal of Technology Innovations and Research, Vol 2, March 2013, Pages 137-146, ISSN: 2321-1814, ISBN: 978-1-62776-111-6.

[10] Ren K, Thomson A, Abadi DJ. VLL: a lock manager redesigns for main memory database systems. VLDB J. 2015; 24(5):681-705.

[11] Kumar TV, Kumar L, Arun L. Distributed query plan generation using BCO. Int J Swarm Intell. 2015; 1(4):358-377.

[12] Mishra SK, Pattnaik S, Patnaik D. Evaluating query execution plans by implementing joins operators using particle swarm optimization. J Comput Sci Appl. 2014; 2(2):31-35. .

[13] Lakshmi SV, Vatsavayi VK. Query optimization using clustering and genetic algorithm for distributed databases. Paper presented at: 2016 International Conference on Computer Communication and Informatics (ICCCI); 2016; Coimbatore, India.

[14] Chiregi M, Navimipour NJ. A new method for trust and reputation evaluation in the cloud environments using the recommendations of opinion leaders' entities and removing the effect of troll entities. Comput Hum Behav. 2016; 60:280-292.

[15] Sheikholeslami F, Navimipour NJ. Service allocation in the cloud environments using multi-objective particle swarm optimization algorithm based on crowding distance. Swarm Evol Comput. 2017; 35:53-64. 60. Gh

[16] Khezr SN, Navimipour NJ. Map Reduce and its applications, challenges, and architecture: a comprehensive review and directions for future research.J Grid Comput. 2017; 15(3):295321.

[17] Guerrero C, Lera I, Juiz C. Genetic algorithm for multi-objective optimization of container allocation in cloud architecture. J Grid Comput.2018; 16(1):113-135.

[18] Keshanchi B, Souri A, Navimipour NJ. An improved genetic algorithm for task scheduling in the cloud environments using the priority queues: formal verification, simulation, and statistical testing. J Syst Softw. 2017; 124:1-21.

[19] Panahi, Vahideh, and Nima Jafari Navimipour. "Join query optimization in the distributed database system using an artificial bee colony algorithm and genetic operators." Concurrency and Computation: Practice and Experience (2019): e5218.

[20] Aponso GC, Tennakon TM, Arampath AM, Kandeepan S, Amaratunga HP. Database optimization using genetic algorithms for distributed databases. Int JComput. 2017; 24(1):23-27.

[21] Dermino, Ferero, and Klawi Fortingo. "What is Data Mining Methods with Different Group of Clustering and Classification?" American Journal of Mobile Systems, Applications and Services 1.2 (2015): 140-151.

[22] KumarTV, Singh R, KumarA. Distributed query plan generation using ant colony optimization. Int JApplMetaheuristic Comput. 2015; 6(1):1-22.

[23] Ban W, Lin J, Tong J, Li S. Query optimization of distributed database based on parallel genetic algorithm and max-min ant system. Paper presented at: 2015 8th International Symposium on Computational Intelligence and Design (ISCID); 2015; Hangzhou, China.

[24] Mishra V, Singh V. Generating optimal query plans for distributed query processing using teacher-learner based optimization. Procedia Comput Sci. 2015; 54:281-290.

[25] Adimi N, Afkousi-Paqaleh A, Emamhosseini A. A PSO-based fuzzy long-term multi objective optimization approach for placement and parameter setting of UPFC. Arab J Sci Eng. 2014; 39(4):2953-2963.

[26] Xiang H. Query optimization over a heterogeneously distributed scientific database. In: Proceedings of the 2013 IEEE International Conference on Big Data; 2013; Silicon Valley, CA.

[27] Dokeroglu T, Tosun U, Cosar A. Particle swarm intelligence as a new heuristic for the optimization of distributed database queries. Paper presented at: 2012 6th International Conference on Application of Information and Communication Technologies (AICT); 2012; Tbilisi, Georgia.

[28] ChenY, ZuoW, HeF, ChenK. Optimizing large query by simulated annealing algorithm based on graph-based approach. JSoftw. 2011; 6(9):1655-1663.

[29] Toroslu I, Cosar A. Dynamic programming solution for multiple query optimization problems. Inf Process Lett. 2004; 92(3):149-155.

[30] Sangeeta Sharma, Pawan Bhambu. Artificial Bee Colony Algorithm: A Survey. International Journal of Computer Applications (0975 – 8887); Volume 149 – No.4, September 2016.

[31] Siew Mooi Lim, Abu Bakar Md. Sultan, Md. Nasir Sulaiman, Aida Mustapha, and K. Y. Leong. “Crossover and Mutation Operators of Genetic Algorithms”. International Journal of Machine Learning and Computing, Vol. 7, No. 1, February 2017.