ISSN: 0975-8585

Research Journal of Pharmaceutical, Biological and Chemical Sciences

Efficient Heterogeneous Grid Data Sharing Using Heuristics Based Dynamic Hash Key Generation.

Ashokkumar K[1], and Chandrasekar C[2]

[1]Sathyabama University, Chennai, Tamil Nadu, India.
[2]Professor, Periyarr University, Salem, Tamil Nadu, India.

ABSTRACT

With huge quantity of data distributed in scientific and industrial world, demand for parallel accessing of information motivates the usage of Grid. Parallel accessing of data grid at different location needs to be handled properly, to improve the computational and resource efficiency (i.e., reliability) of the Grid Data Environment. Star-topology based data grid although keep track of users' file access behaviors, but is not effective in developing a reliable dynamic replication model. While prior works on grid computing though enables the operation of multiple data grid services (i.e., heterogeneous environment), but lacks in the storage and computation overheads while performing parallel data accessing. To develop an efficient multiple and reliable grid data storage, a framework called, Heuristics Based Dynamic Hash Key Generation (HDHKG) is introduced to store the data in virtual grid points. The main objective of the framework HDHKG is to represent multiple data grids in a sequence of dynamic hash keying without any effect on data stored in the grids. Initially, an iterated Local Meta-heuristic Search over dynamic hash key storage on grid interchange is performed to optimize the single sequence in HDHKG. To support efficient handling of storage, Meta-heuristic Search in HDHKG moves to the next data grid only if feasible solution is obtained with better performance in distributed grid environment reducing the computation overhead. In addition to achieve interchanges on data grid and optimize the storage, Dynamic Hash Key is carried out effectively in grid location with parallel accessing with different sequence of data grid and meta-heuristic search operations. Finally, Optimized Meta-heuristic Dynamic Hash Key algorithm is designed to reduce the parallel accessing time for heterogeneous grid data sharing. The proposed framework, HDHKG is tested with multiple data exchange grid data sets from research repositories using grid simulator with performance parameters such as data storage, parallel accessing time, reliability and computation overhead.

Keywords: Grid computing, Parallel accessing, Star-topology, Meta-heuristic Search, Dynamic Hash Key Generation, Grid data sharing

Corresponding author

INTRODUCTION

Grid computing refers to the upcoming infrastructure model involving both computational and networking that is sketched in such a way to provide different domains involving, pervasive and reliable data access over a wide organizational domain. Dynamic Replication in Data Grid using Modified Region (DRDG-MR) (Sashi.k, Thanamani, 2011) with the application of modified Bandwidth Hierarchy Replication (BHR) algorithm was designed with the motive of reducing the data access time. But, the model was not effective in developing a reliable dynamic replication model.

Popular File Replicate First (PERF) (Ming-Chang Lee et al, 2012) ensured reliability through star-topology data grids, however, storage and computation overheads while performing parallel data accessing increased with the increase in the user density. Parallel data accessing with security was addressed in (Manghui Tu et al ,2010) with the application of general and tree graph. Though access performance was increased, but at the cost of energy. Energy efficient techniques was introduced in (Shu Yin et al, 2014) with the objective of improving the reliability.

The application of grid in several domains has been increased with the increase in their applicability. In (Ni Zhang, et al, 2014), centralized data processing and storage scheme was introduced to improve the data storage in grid environment. To improve performance optimization and quality of service in grid environment, autonomic management model was designed in (Jesus Montes ,2012) with the aid of service level system management model. A comparison between different grid models was provided in (Gideon Juve et al, 212) using kepler periodogram workflow. However, security remained unaddressed.

To address security and improve data access performance, a model called optimal resident set (Aruna Kranthi et al, 2012) was designed. However, with the increase in data sharing, security was compromised. To address this security issue, Message Passing Interface (MPI) (Xiaojun Ruan et al 2012) was applied and included with encryption and decryption algorithms. Integration Control Communication and Metering (ICCM) (Reddy.k.s et al, 2014) were designed on the basis of smart grid to address the issues related to energy optimization and efficient data transfer.

Clustering, an unsupervised learning process is also used in grid in order to split multidimensional data object into unique cells. In (Raghavi Chouhan, Abhishek Chauhan, 2014) grid-based clustering was performed using extended K-medoid algorithm improving the accuracy. Though, accuracy was improved, security aspects were not covered. Improving security and privacy in smart grid was addressed in (Rajiv .K. Bhatia, Varsha Bodade, 2014) with the aid of classification model.

In this work, an efficient multiple and reliable grid data storage using the framework, Heuristics Based Dynamic Hash Key Generation (HDHKG) is introduced to store the data in virtual grid points. The major contributions of this work are summarized as follows:

1. Identify the resource availability and optimal solution for the user requested task using Local Meta-heuristic PSO based Search.
2. Investigate a radically different approach through Dynamic Hash Key Generation to optimize the storage and ensure efficient mapping through Dynamic Hash Table.
3. Multiple and reliable grid data storage using virtual data points to reduce the parallel accessing time.
4. Evaluate the proposed reliable grid data storage framework and proof-of-concept test bed with a comprehensive set of performance measures.

The remainder of this paper is organized as follows. Section 2 presents the motivation for the development of a reliable and multiple grid data storage framework with multiple data exchange grid datasets and the related work of this study. Section 3 outlines the design and implementation of the Heuristic Dynamic Hash Key Generation (HDHKG) framework. In Section 4, we leverage HDHKG with experimental setup to build the reliable and multiple grid data storage framework. Section 5 presents the discussion on HDHKG framework and comparison made with existing state-of-the-art methods. Section 6 concludes the paper.

RELATED WORK

Data Grid involves a Grid Computing environment that is designed in such a way that significantly ensures geographically distributed data and therefore evaluating enormous amounts of data. Efficient data management has to be ensured to reduce the data accessing time. In (Sheida Dayyani, Mohammad Reza Khayyambashi, 2014), a novel replication strategy called, Dynamic Hierarchical Replication with Threshold (DHR-T) was designed with the objective of minimizing the data access time. Efficient Dynamic Replication Algorithm (EDRA) (Priyanka V et al, 2014) was constructed using data grid simulator to reduce the mean job execution time and improve the network usage. Though mean job execution time was reduced, parallel accessing was not performed.

In (Mais Nijim et al, 2012) energy conserving strategy in grid was designed using data partitioning mechanism with the objective of improving the bandwidth. To improve reliability and security in smart grid environment, mobile virtualization was introduced in (Su-Wan Park et al, 2015) using access control and key management schemes. However, scalability remained unsolved. High reliability and scalability was solved in (Kawanabe Masazumi et al, 2012) using accelerator and storage nodes for efficient data storage.

The data grid environment has been used extensively for accessing grid and providing storages in order to deal with the increasing needs of applications subject to voluminous data and throughput. In (Alberto Sanchez et al, 2011) a framework was designed to improve the quality of service in grid environment using decision making process. However, scalability was not focused. A scalable model called, Map Reduce platform using big data classification was introduced in (Galip Aydin et al, 215) via high performance server. In (Xiong Fu et al, 2015) an algorithm called, Heuristic Optimal Virtual Machine (HOVM) was introduced to solve the issues related to scheduling of subtasks.

Based on the above mentioned methods, a framework for an efficient multiple and reliable grid data storage called Meta-heuristic Dynamic Hash Key Storage is designed and explained in detail in the forthcoming sections.

Design Of Heuristic Dynamic Hashkey Generation

Grid computing is one of the emerging trends that makes easy access to multiple and reliable storage and has therefore evolved as an important factor. Grid computing on the other hand involves distributed nature and heterogeneous system to achieve increased computational performance and reliability. The main objective includes addressing issues related to computationally significant problems that cannot be addressed with the help of single CPU.

The data files are very large and the grid storages are limited. As a result, efficient virtual storage points are required to serve multiple user requests from different location. The above scenario motivates to develop a Heuristic Dynamic Hash Key Generation (HDHKG) Framework in the lines of improving the efficient handling of storage computation and reduce the parallel accessing time for heterogeneous grid data sharing in Grid Environment.

With the motive of minimizing the computation overhead, Local Meta-heuristic Particle Swarm Optimization (PSO) based Search is applied in the proposed framework, Heuristic Dynamic Hash Key Generation. This is achieved by obtaining the resource availability and identifying the optimal solution for user requests through PSO based Search.

Figure 1 shows the Architecture diagram of Heuristic Dynamic Hash Key Generation Framework. Based upon the user request, the framework, Heuristic Dynamic Hash Key Generation is divided into three parts, namely, (i) Design of Local Meta-heuristic PSO based Search, (ii) Construction of Dynamic Hash Key Storage and (iii) Generating an Optimized Meta-heuristic Dynamic Hash Key algorithm.

Then, Dynamic Hash Key Storage is applied to the PSO based Search for optimizing and providing reliable grid storage in grid environment. This is performed using virtual grid points to improve the reliability and efficient multiple grid data storage. Finally, by introducing an Optimized Meta-heuristic Dynamic Hash Key algorithm, parallel accessing time is reduced during dynamic hash key storage interchanges on the data grid.

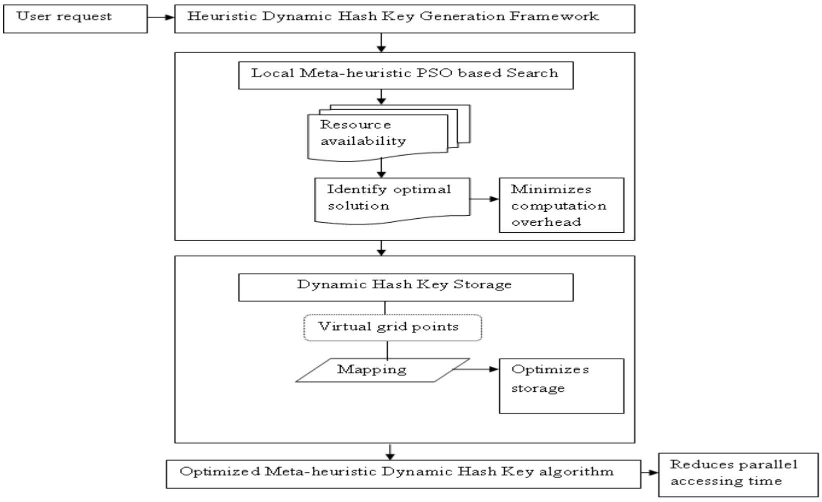

Figure 1: Architecture diagram of Heuristic Dynamic Hash Key Generation Framework

The elaborate description about the proposed framework, Heuristic Dynamic Hash Key Generation is explained in the forthcoming sections.

Design of Local Meta-heuristic PSO based Search (minimizes the computation overhead)

In this section, the Local Meta-heuristic PSO based Search is described which are aimed at fulfilling the objective of minimizing the computation overhead for reliable grid data storage in grid environment. Local Meta-heuristic PSO based Search involves an iterative process that begins with an initial solution and generates sequence of solutions with the efficient application of trivial local changes. The objective of using Local Meta-heuristic PSO based Search in Heuristic Dynamic Hash Key Generation framework is the solution to local optimality.

Grid data storage using meta-heuristic search in HDHKG framework is performed extensively based on the resource availability and identifying the optimal solution rapidly. But, due to higher amount of tasks and resources, data storage involves a tedious task. Therefore, the framework, HDHKG uses Local Meta-heuristic PSO based Search for efficient grid resource management and therefore reliable grid data storage.

The Local Meta-heuristic PSO based search efficiently identifies the desired solution according to the user query and therefore matches the resource required by the user to the task. Figure 2 show the flow model of Local Meta-heuristic PSO based Search.

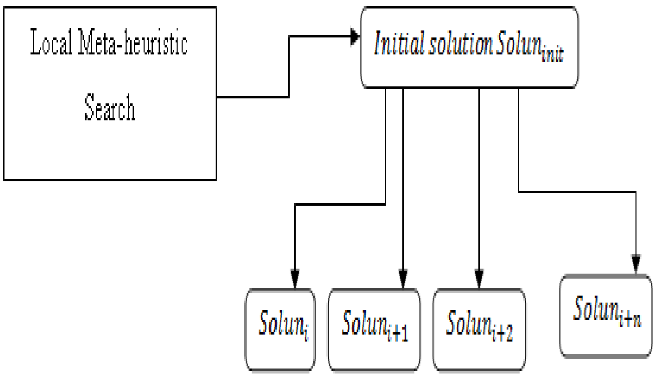

Figure 2: Flow model of Local Meta-heuristic PSO based Search

As shown in the figure, the Local PSO based search in Heuristic Dynamic Hash Key Generation Framework first obtains the initial solution '$Solun_{init}$'. The meta-heuristic search operation in the HDHKG form using PSO based search moves to the next data grid only if the feasible solution is obtained with better performance in the distributed grid environment.

If a new solution rates (i.e., feasible solution) is better when compared with the best current solution, it replaces the current best solution and moves to the next data grid i.e., '$Solun_i, Solun_{i+1}, \ldots, Solun_{i+n}$' and so on. The entire process of meta-heuristic search operation stops upon identification of feasible solution. The solution to the problem is the best solution found so far.

The PSO based Search represents a swarm of individuals (called resources) fly through the search space. Each particle represents a candidate solution (i.e., best possible resources) to the optimization problem. The position of a particle is decided upon several factors like, best resource (Res_{best}) it has visited.

The identification of best resource is obtained either through its own movement or by observing neighboring particles. This PSO based Search in HDHKG framework stops iteration when the global best resource ($Global_{best}$) of the particle, is obtained. The mathematical formulation for optimized iterations is given as below

$$Iterations_{opt} = Vel_i + Const_1(Res_{best} - A_i) + Const_2(Global_{best} - A_i) \quad (1)$$

From (1), the position and velocity of the 'ith' particle is represented by 'A_i' and 'Vel_i' respectively, whereas '$Const_1$' and '$Const_2$' denotes the constants for best resource and global best resources respectively. Local Meta-heuristic PSO based search is performed for reliable data storage and therefore minimizing the computation overhead using equation (1).

Construction of Dynamic Hash Key Generation (optimizes the storage)

In this section, the PSO based search for efficient and optimized storage in grid environment is presented. The main objective of the HDHKG form is to represent the multiple data grids in a sequence of dynamic hash keying without any effect on data stored in the grids. The dynamic hash key generation interchanges on the data grid carried out effectively in grid location with parallel accessing and therefore optimizing the storage during heterogeneous grid data sharing.

The important issue in key representations for applying PSO based search to perform parallel accessing and therefore optimizing the storage is to identify a worthy mapping between user specified task and resource assignment. Let us assume a evaluated grid resource management comprising of '$UserD$' user domains, user given task '$UserT$' with 'Res_N' types of resources and $UserD_k$' represents the user domain set of keys. The types of resources include different attributes namely '$\{CPU = 'P5', Operating\ System = 'Linux', Memory = '5G'\}$'.

Dynamic Hash Key Generation (DHKG) in HDHKG framework provides a lookup system similar to hash table ($USerD_k$, value) pairs that are generated in a Dynamic Hash Table, and any access is efficiently performed by providing the given key. The advantage of using DHKS is the minimal amount of disruption and storage required for effective maintenance of mapping between keys and values. In grid system, the data grid operation is measured by hashing the name of a node. The formulation is given as below

$$Q = HASH\ (UserD[IP]) \quad (2)$$

From (2), 'Q' represent the unique digital representation of a node, whereas the 'IP' refers to the IP address of the user domain 'UserD' using function 'HASH'. The key in hash represents the unique identifier of a grid data item and is obtained by hashing the name of a grid data item and is given as below

$$USerD_k = HASH\ (UserD) \quad (3)$$

To map user domain set of keys with user domains in Dynamic Hash Key Storage, i.e., '$USerD_k$' ∈ '$UserD$', then '$USerD_k$' is virtualized with '$UserD$'. Figure 3 illustrates the computational grid resource management that performs efficient mapping using Dynamic Hash Key Storage in virtual grid points. In order to perform efficient mapping, user domain, '$UserD$' is virtualized.

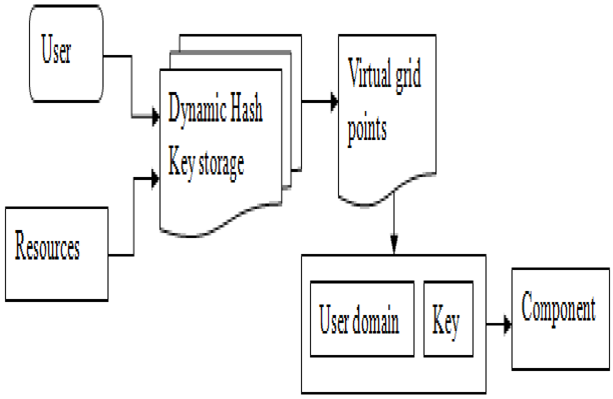

Figure 3: Dynamic Hash Key Storage using virtual grid points

In DHT based grid systems using Dynamic Hash Key Storage, the files are associated to keys whereas each node in the system executes a part of the overall hash space and stores certain range of key. This results in efficient mapping and therefore optimizing the storage.

Optimized Meta-heuristic Dynamic Hash Key algorithm (reduce the parallel accessing time)

In this section, the algorithmic representation of Optimized Meta-heuristic Dynamic Hash Key with the objective of reducing the parallel accessing time in grid environment is provided below. Algorithm 1 given below shows the detailed flow of Optimized Meta-heuristic Dynamic Hash Key (OMDHK) algorithm. The OMDHK algorithm consists of two parts. The first part includes the Local Meta-heuristic PSO based Search and the second part comprises of efficient mapping between user given task and resource assignment with the aid of Dynamic Hash Key Storage using virtual grid points.

Begin

Initialize '$UserD$', 'N', '$USerD_k$',
User submits request for grid service to the Grid Resource Manager (GRM) controlling the user domain
GRM detects group of resource availability and identifies the optimal solution using Local Meta-heuristic PSO based Search

Repeat

Perform *PSO based search*
If *(new solution rates) better than (best current solution)*
Move to the next data grid
Obtain the optimized iterations from (1)

Else
Grid pointer continuous to stay in the recent similar solution
Until *(feasible solution is obtained)*
End if
Perform *mapping between user given task and resource assignment*
For each '$UserD$' *and* 'N' *types of resources*
Perform *data grid operation using ()*

Obtain *key value based on user domain from ()*
For *each* $'USerD_k' \in$ $'UserD'$
Perform *virtual grid points* $'USerD_k'$ *with* $'UserD'$
End for
End for
End

Algorithm 1 - Optimized Meta-heuristic Dynamic Hash Key algorithm

As provided in the above algorithm 1, the initial task of the Grid Resource Manager is to develop an efficient multiple and reliable grid data storage by identifying the optimal solution. The optimal solution is obtained on the basis of resource availability. A Local Meta-heuristic PSO based Search is performed to move to the next data grid by comparing the new solution rate and best current solution rate. This process is performed until feasible or optimal solution is obtained. Once an optimal solution is obtained, significant mapping is performed between the user given task and resource assignment.

The data grid operation is performed by obtaining the key value based on user domain and to store the data in virtual grid points. The single sequence in HDHKG is optimized using an iterated local meta-heuristic search over dynamic hash key storage on grid interchange. Dynamic hash key storage is carried out with different sequence of data grid and meta-heuristic search operations. The proposed OMDHK algorithm is tested with multiple data exchange grid data sets from research repositories.

EXPERIMENTAL SETUP

To study the performance and reliable grid data storage of HDHKG framework, simulation experiments were conducted based on Gridsim simulator and applied in JAVA using GEOSTAT_Grid_POP_2006_1K European population grid dataset for the year 2006. The dataset integrates data from national grid initiative and European disaggregated dataset obtained from the Austrian Institute for Technology (AIT) into an integrated single population grid dataset. Simulations were performed for 35 user requests at a time interval of 5 seconds and each request consists of a randomly selected data attributes (i.e., resources) and was initiated by the Grid Resource Manager.

The work Heuristic Based Dynamic Hash Key Generation (HDHKG) is compared against the existing Dynamic Replication in Data Grid using Modified Region (DRDG-MR) [1] and Popular File Replicate First (PERF) [2] technique. The experiment is conducted on the factors such as data storage, parallel accessing time, reliability and computation overhead.

The computation overhead in the HDHKG framework evaluates the identification of best resource based on the time taken to obtain the global best and best resource respectively. It is measured in terms of milliseconds (ms). The mathematical formulation of computation overhead using the proposed HDHKG framework is given as below

$$CO = Time\,(Global_{best} - Res_{best}) * Request\ Density \quad (4)$$

Data storage using HDHKG framework measures the rate at which the information is read from or written and is measured in terms of megabytes per second or MB/s. Data storage measures the number of successful user domains store data in virtual grid points. The mathematical formulation is given as below

$$DS = \frac{(SUserD - USUserD)}{Total\ UserD} \quad (5)$$

From (5), $'SUserD'$ and $'USUserD'$ represents the successful and unsuccessful user domains whereas the total user domain for an iteration is given as $'Total\ UserD'$. Parallel accessing time using HDHKG framework refers to the time taken to obtain the user given task and the assignment of resources according to the user request.

$$ParallelAccessing_{time} = Time\ [UserT + Res_N] \quad (6)$$

From (6), '$UserT$' refers to the user given task whereas 'Res_N' represents the number of resources required by the user. Reliability refers to the percentage of grid data storage performed for several user domains with different amount of resources in grid environment. Higher the percentage of reliability, more efficient the method is said to be.

Simulation results

In order to analyze the characteristics and functionality of the HDHKG framework, the performance is quantitatively accessed on the basis of GEOSTAT_Grid_POP_2006_1K European population grid dataset by comparing the outcomes to the results achieved with the Optimized Heuristic Dynamic Hash Key algorithm. The Heuristic Dynamic Hash Key Generation (HDHKG) framework is compared against the existing Dynamic Replication in Data Grid using Modified Region (DRDG-MR) [1] and Popular File Replicate First (PERF) [2] technique. The experimental results using Gridsim in JAVA are compared and analyzed using table values and graphical form as given below.

Measure of computation overhead

To support transient performance, in Table 1 we apply an efficient Optimized Meta-heuristic Dynamic Hash Key (OMDHK) algorithm to obtain the computation overhead and comparison is made with two other existing techniques, DRDG-MR and PERF.

Table 1: Tabulation for computation overhead

Request density	Computation overhead (ms)		
	HDHKG	DRDG-MR	PERF
5	0.10	0.15	0.25
10	0.25	0.37	0.45
15	0.35	0.47	0.53
20	0.43	0.55	0.63
25	0.52	0.67	0.75
30	0.48	0.60	0.68
35	0.55	0.67	0.75

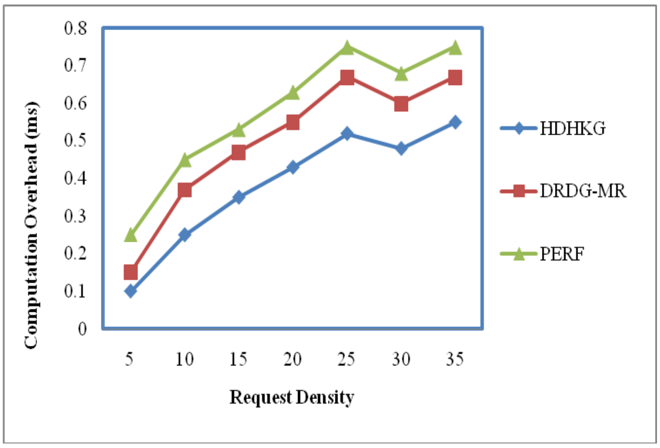

Figure 4: Impact of computation overhead

Figure 4 shows the impact of computation overhead with respect to varied request density made by different users in the grid environment. Figure 4 show that the proposed Heuristic Dynamic Hash Key Generation (HDHKG) framework provides lower computation overhead when compared to DRDG-MR [1] and PERF [2]. This is because of the application of Local Meta-heuristic PSO based Search that eventually identify the resources based on the resource availability and identifying the optimal solution rapidly and reduces the computation overhead by 21 – 50 % when compared to DRDG-MR [1] technique. In addition to that with the use of the Local Meta-heuristic PSO based search that significantly identifies the desired solution according to the user query and therefore matches the resource required by the user to the task appropriate elliptic shape, which helps in improving the computation overhead by 36 – 80 % than the PERF [2] technique.

Measure of data storage

The comparison of data storage is presented in table 2 with respect to the varying number of request density in the range of 5 – 35 at a time interval of 5 seconds using GEOSTAT_Grid_POP_2006_1K European population grid dataset obtained from the year 2006. With increase in the number and size of request density from different users, the data storage in grid environment is also increased.

Table 2: Tabulation for data storage

Request density	Data Storage (MB/s)		
	HDHKG	DRDG-MR	PERF
5	27	25	24
10	45	36	30
15	69	50	44
20	81	72	66
25	78	69	63
30	92	83	76
35	115	106	100

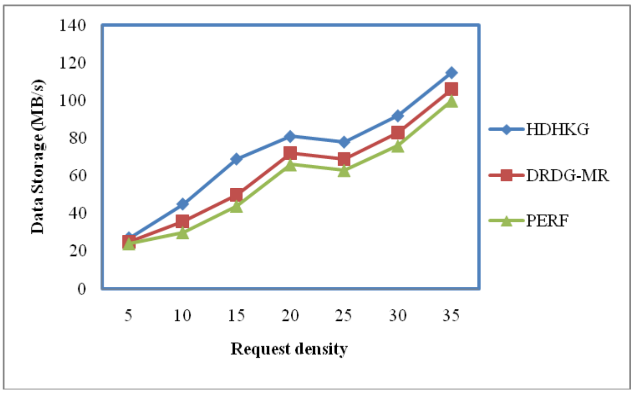

Figure 5: Impact of data storage

To ascertain the performance of the data storage, comparison is made with two other existing techniques, Dynamic Replication in Data Grid using Modified Region (DRDG-MR) [1] and Popular File Replicate First (PERF) [2]. In Figure 5, the request density sizes to obtain the population density in the year 2006 for European grids are varied between 5 and 35. From the figure it is illustrative that the data storage which is measured in terms of MB/s is comparatively higher using the proposed Heuristic Dynamic Hash Key Generation (HDHKG) framework when compared to the two other existing techniques. This is because with the application of the Dynamic Hash Key Storage that uses the multiple data grids in a sequence of dynamic hash keying optimizes the storage during heterogeneous grid data sharing. The efficiency is improved by 7 – 27 % when compared to DRDG-MR [1]. Furthermore, by providing Dynamic Hash Table and in virtual grid points combines the two processes for effective data storage in grid environment by 11 – 33 % than compared to PERF [2].

Measure of parallel accessing time

The parallel accessing time value for HDHKG framework is elaborated in table 3. We consider the framework with varying number of request density acquired from 35 different users at different time period for access and use of the population data using Gridsim simulator in JAVA.

Table 3: Tabulation for parallel accessing time

Request density	Parallel accessing time (ms)		
	HDHKG	DRDG-MR	PERF
5	37	39	53
10	48	53	68
15	59	64	79
20	71	76	91
25	68	73	88
30	81	86	101
35	98	103	115

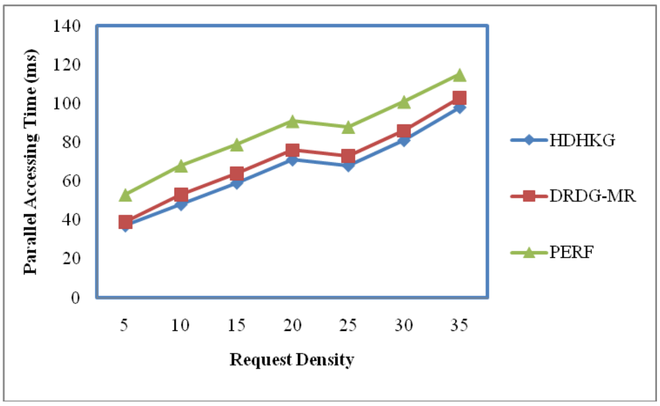

Figure 6: Impact of parallel accessing time

In figure 6, we depict the parallel accessing time using varying request density obtained from 35 different users for experimental purpose and applied in Gridsim simulator. From the figure, the value of parallel accessing time achieved using the proposed HDHKG framework is lower when compared to two other existing techniques namely, Dynamic Replication in Data Grid using Modified Region (DRDG-MR) [1] and Popular File Replicate First (PERF) [2]. Besides we can also observe that by increasing the number of request density made to access and use population data, the parallel accessing time is increased using all the methods. But comparatively, it is lower in HDHKG framework because of the application of Optimized Meta-heuristic Dynamic Hash Key algorithm that efficiently processes the user request in grid environment improving the parallel accessing time by 5 – 10 % compared to DRDG-MR. In addition, by applying OMDHK algorithm, data grid operation using virtual grid points is performed in an efficient manner reducing the parallel accessing time by 17 – 43 % compared to PERF respectively.

Measure of reliability

Table 4 Tabulation for reliability

Methods	Reliability (%)
HDHKG framework	74.35
DRDG-MR technique	62.15
PERF technique	58.25

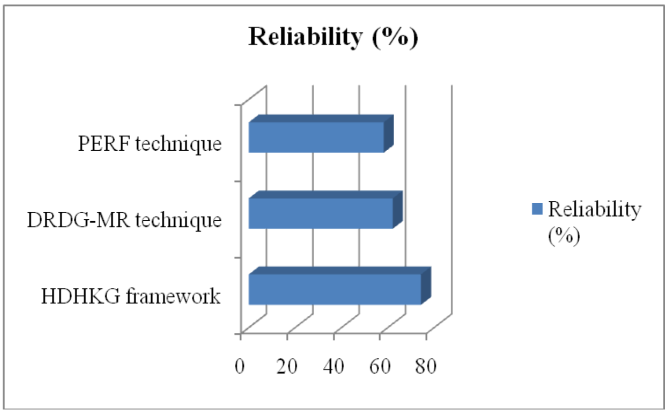

Figure 7: Impact of reliability

Table 4 and Figure 7 illustrate the reliability versus the request domain made by different user density at different time interval to obtain the grid dataset population of Europe in 2006 that is measured in terms of percentage (%) for experimental purpose conducted using Gridsim simulator in JAVA. From the figure, it is illustrative that the reliability using the proposed HDHKG framework is higher than when compared to two other existing techniques. This is because of the integration of two methods PSO based search for efficient and optimized storage using Dynamic Hash Key Storage for different user domain density. This in turn efficiently represent the multiple data grid by ensuring reliable grid data storage and therefore increases the reliability of the framework involved by 16.40 % and 6.27 % improvement when compared to the existing techniques, DRDG-MR [1] and PERF [2] respectively.

CONCLUSION

In this work, we address the problem of storage and computation overheads while performing parallel data accessing during the operation of multiple data grid services in grid environment and propose a Heuristic Dynamic Hash Key Generation (HDHKG) framework. The HDHKG framework is based on the Local Meta-heuristic Search over dynamic hash key storage to minimize computation overhead and achieve efficient multiple grid data storage using virtual grid points. We show how different request density made by different users at varied interval of time with multiple data exchange grid datasets from European population grid dataset can be used in a ubiquitous fashion to drastically reduce the parallel accessing time taken during interchanges on the data grid in grid environment. We study how this reduction in parallel accessing time translates into efficient data storage and obtain optimal solution user requests through PSO based Search. We further show attainable performance gains of the proposed framework in terms of optimized storage that can achieve well above 33 percent compared with the state-of-the-art methods in grid environment. Moreover, with the application of grid data points using Dynamic Hash Key Generation increases the reliability of the grid data storage, that utilize the Optimized Meta-heuristic Dynamic Hash Key algorithm resulting in marginal improvements in increasing the reliability for many user requests. Finally by efficient mapping between user given task and resource assignment, minimal amount of disruption and storage is required and therefore further improvements in efficient mapping and therefore optimizing the storage is attained. The simulation results show that HDHKG framework outperforms the state-of-the-art methods in terms of computation overhead, data storage size, reliability and parallel accessing time when user requests changes with time.

REFERENCES

[1] K. Sashi, Antony Selvadoss Thanamani, “Dynamic Replication In A Data Grid Using A Modified Bhr Region Based Algorithm,” Elsevier Future Generation Computer Systems, Vol 27, Issue 2, Pages 202-210, February 2011.

[2] Ming-Chang Lee, Fang-Yie Leu, Ying-Ping Chena, “Pfrf: An Adaptive Data Replication Algorithm Based On Star-Topology Data Grids,” Elsevier Future Generation Computer Systems., Vol 28, Issues 7, Pages 1045-1057, July 2012.

[3] Manghui Tu, Peng Li, I-Ling Yen, Bhavani Thuraisingham, And Latifur Khan, "Secure Data Objects Replication In Data Grid," Ieee Transactions On Dependable And Secure Computing, Vol. 7, No. 1. Page(S): 50-64, 2010.

[4] Shu Yin, Xiaojun Ruan, Adam Manzanares, Xiao Qin, And Kenli Li, "Mint: A Reliability Modeling Framework For Energy-Efficient Parallel Disk Systems," Ieee Transactions On Dependable And Secure Computing, Vol. 11, Issue. 4, Page(S): 345-360, 2014.

[5] Ni Zhang, Yu Yan, Shengyao Xu, Wencong Su, "A Distributed Data Storage And Processing Framework For Next-Generation Residential Distribution Systems," Elsevier Electric Power Systems Research, Vol. 116, Pages 174-181, 2014.

[6] Jesus Montes, Alberto Sanchez, María S. Perez, "Riding Out The Storm: How To Deal With The Complexity Of Grid And Cloud Management," Journal Of Grid Computing, Springer, Vol. 10, Issue 3, Pages 349-366, 2012.

[7] Gideon Juve, Mats Rynge, Ewa Deelman, Jens-S Vöckler And G. Bruce Berriman, "Comparing Futuregrid, Amazon Ec2, And Open Science Grid For Scientific Workflows," Ieee Computing In Science And Engineering, Vol. 15, Issue 4, Page(S): 20-29, 2013.

[8] G. Aruna Kranthi And D. Shashi Rekha, "Protected Data Objects Replication In Data Grid," International Journal Of Network Security & Its Applications (Ijnsa), Vol.4, No.1, Pp: 29-41, 2012.

[9] Xiaojun Ruan, Qing Yang, Mohammed I. Alghamdi, Shu Yin, And Xiao Qin, "Es-Mpich2: A Message Passing Interface With Enhanced Security," Ieee Transactions On Dependable And Secure Computing, Vol. 9, Issue No. 3, Pp: 361-374, 2012.

[10] K.S. Reddy, Madhusudankumar, T.K.Mallick, H.Sharon, S.Lokeswaran, "A Review Of Integration, Control, Communication And Metering (Iccm) Of Renewable Energy Based Smart Grid," Elsevier Renewable And Sustainable Energy Reviews, Vol. 38, Pages 180 – 192, 2014.

[11] Raghavi Chouhan, Abhishek Chauhan, "An Ameliorated Partitioning Clustering Algorithm For Large Data Sets", International Journal Of Advanced Research In Computer And Communication Engineering Vol. 3, Issue 7, Pp: 7617-7621, 2014.

[12] Rajiv .K. Bhatia, Varsha Bodade, "Smart Grid Security And Privacy: Challenges, Literature Survey And Issues," International Journal Of Advanced Research In Computer Science And Software Engineering, Volume 4, Issue 1, Pp. 702-706, 2014.

[13] Sheida Dayyani, Mohammad Reza Khayyambashi, "A Novel Replication Strategy In Data Grid Environment With A Dynamic Threshold, International Journal Of Computer Science Engineering, Vol. 3 No.05, Pp: 244-252, 2014.

[14] Priyanka Vashisht, Rajesh Kumar, And Anju Sharma, "Efficient Dynamic Replication Algorithm Using Agent For Data Grid", Hindawi Publishing Corporation, The Scientific World Journal Volume 2014, Article Id 767016, 10 Pages, 2014.

[15] Mais Nijim, Xiao Qin, Meikang Qiu, Kenli Li, "An Adaptive Energy-Conserving Strategy For Parallel Disk Systems", Elsevier Future Generation Computer Systems, Vol. 29, Issue 1, Pages 196-207, 2012.

[16] Su-Wan Park, Jaedeok Lim, And Jeong Nyeo Kim, "A Secure Storage System For Sensitive Data Protection Based On Mobile Virtualization," Hindawi Publishing Corporation International Journal Of Distributed Sensor Networks Volume 2015, Article Id 929380, 8 Pages, 2015.

[17] Kawanabe Masazumi, Yoshimura Shigeru, Utaka Junya Yoshioka Hiroshi, Mizumachi Hiroaki, Kato Mitsugu, "Large-Capacity, High-Reliability Grid Storage: Istorage Hs Series", Nec Technical Journal Vol.7 No.2, Pp: 47-51, 2012.

[18] Alberto Sanchez, Jesus Montes, Marıa S. Perez, Toni Cortes, "An Autonomic Framework For Enhancing The Quality Of Data Grid Services," Elsevier Future Generation Computer Systems, Vol. 28, Issue 7, Pages 1005-1016, 2011.

[19] Galip Aydin, Ibrahim Riza Hallac, And Betul Karakus., "Architecture And Implementation Of A Scalable Sensor Data Storage And Analysis System Using Cloud Computing And Big Data Technologies," Hindawi Publishing Corporation Journal Of Sensors Volume 2015, Article Id 834217, 11 Pages, 2015.

[20] Xiong Fu, Yeliang Cang, Xinxin Zhu, And Song Deng, "Scheduling Method Of Data-Intensive Applications In Cloud Computing Environments," Hindawi Publishing Corporation Mathematical Problems in Engineering Volume 2015, Article ID 605439, 2015.