ISSN 2321-807X
Volume 12 Number24
Journal of Advances in chemistry

Processing of Spatio-Temporal Hybrid Search Algorithms in Heterogenous Environment Using Stochastic Annealing NN Search

Dr. K. Balasaravanan[1], D. Manivannan[2] , N.K.Manikandan[3]
[1]Associate Professor, Sree Naranaya Guru College of Engineering and Technology, Payyanur, Kerala, India
[2, 3]Assistant Professor, Veltech Rangarajan Dr.sakthunthala R & D Insitute of science and technology, chennai – 600 062 , India
[1]E-mail: researchingscholar@gmail.com

ABSTRACT

Problem statement: In spatio-temporal database the mixed regions are present in a random manner. The existing work produces the result to create new research opportunities in the area of adaptive and hybrid SLS algorithms. This algorithm develops initialization algorithms which are used only for the homogenous environment. Most current approaches assume, as we have done here, only the homogenous mixtures. **Approach:** To overcome the above issue, we are going to implement a new technique termed Stochastic Annealing Nearest Neighbor Search using hybrid search algorithms (SANN- HA) for spatio-temporal heterogeneous environment to retrieve the best solution. It provides enhanced fits for definite run length distributions, and would be useful in other contexts as well. **Results:** Performance of Stochastic Annealing Nearest Neighbor Search using hybrid search algorithms is to discover different sub explanations using different mixture of algorithms in terms of run length distribution and average time for execution based on data objects. **Conclusion:** It considers the problem of retrieving the high quality solution from the heterogeneous environment. An analytical and empirical result shows the better result with the efficient hybrid search algorithms of our proposed SANN scheme.

Keywords: **Hybrid search algorithms, spatio-temporal database, Search retrieval, Stochastic NN search, Heterogeneous environment**

1. INTRODUCTION

Classical data mining techniques often execute inadequately when useful to spatial and spatio -temporal data sets because of the many reasons. A.N.M. Bazlur Rashid., and Md. Anwar Hossain., 2012 embed dataset in continuous space, whereas classical datasets are frequently distinct. Second, patterns are often local where as classical data mining techniques regularly focus on global patterns. Finally, one of the common assumptions in classical statistical analysis is that data samples are separately generated. When it comes to the analysis of spatial and spatio -temporal data, the assumption regarding the independence of samples is usually false because such data tends to be highly self correlated.

For instance, public with comparable individuality, occupation and environment tend to cluster collectively in the similar neighborhoods. In spatial statistics this tendency is called autocorrelation. Ignoring autocor relation when analyzing data with spatial and spatiotemporal characteristics may produce hypotheses or models that are inaccurate or inconsistent with the data set.

The similarity of spatial and temporal phenomena has been predictable for an elongated time in literature. Both the happenings are deal with the 'spaces' or 'dimensions' of some kind and are thus closely related. The consciousness of their deep associations has led to both in spatial and in temporal data modeling. It increases the interest by integrating both directions into a common research branch called spatiotemporal data modeling and in constructing spatio-temporal data bases. Their underlying basic entities are called spatiotemporal objects and are ubiquitous in everyday life.

Temporal changes of spatial data objects encourage adjustment of their mutual topological associations over time. For instance, at one time, two spatio-temporal objects might be disjoint whereas they might intersect. These modifications frequently proceed incessantly over time but can, of course, also occur in discrete steps. At present it's still an open issue, in relating the nature and recognized definition of these spatio-temporal associations by spatio-temporal predicates.

So distant, only the minority data models for spatio-temporal data have been proposed. One approach is to correctly expand a spatial data model by temporal concepts. Spatio-temporal objects are defined as spatio-bi temporal complexes. Their spatial advantages are described by simplified complexes, and their temporal advantages are agreed by bi temporal elements friendly to all mechanism of simplify complexes.

Simulated Annealing is one the most well known local search methods. In practice, it is frequently used to resolve discrete optimization problems; particularly very tough problems. Global optimization is computationally tremendously difficult and for large instances, exact methods attain their limits quickly. Saed Alazamir., et.AL., 2008 often uses the local optimization methods. Simulated annealing provides a powerful tool for avoidance local optima by allowing moves to lower quality solutions with a pre-specified probability. An additional big plus of Simulated Annealing is its simplicity of implementation.

It's important to point out that in this work; we focus on Hybrid Search Algorithms in a spatio-temporal database. It presents a Stochastic Annealing Nearest Neighbor Search using hybrid search algorithms (SANN- HA) for spatio-temporal heterogeneous environment is to retrieve the best solution. The hybrid search algorithms are used to combine the Ordered Local Search, (OLS), Monitor Ordered Local Search, (MOLS), Marginal Object Weight (MOW) and Global Spatial (GS) algorithms to fit the long run distribution in the heterogeneous environment. Stochastic Annealing Nearest Neighbor Search which generates the arbitrary variables with the random iterations. The hybrid search algorithm used to obtain the optimal solution from heterogeneous environment.

We provide here an overview of Stochastic Annealing Nearest Neighbor Search on spatio-temporal heterogeneous environment. The rest of this paper is arranged as follows: Section 2 produces the literature survey of the paper. Section 3 introduces architecture diagram of the proposed scheme. Section 3.1 and 3.2 describes about proposed method; Section 4 shows the development and experimental evaluation; Section 5 evaluated the results and discuss about it. Section 6 describes conclusion and vision.

2. LITERATURE REVIEW

The spatio-temporal database (STDB) has received substantial attention during the past few years. A.N.M. Bazlur Rashid., and Md. Anwar Hossain., 2012 have accessible the challenging issues of spatio-temporal data mining. Saed Alizamir., et.AL., 2008 improve the efficiency of solving combinatorial optimization problems using simulated annealing method by taking the advantages embodied in neighborhood structures. The algorithm improves simulated annealing in two different aspects, First, is by employing multiple neighborhood structures, performing a extra controlling search and subsequent using optimal stopping problem, it finds the best time to alter the temperature which is a critical issue in simulated annealing. Perez, M., et.Al., 2012 assesses the effectiveness of the Hybrid Genetic Algorithm Simulated Annealing (HGASA) algorithm in selecting features for various classification architectures.

Kaiming He., and Jian Sun., 2012 propose Propagation-Assisted KD-Trees to rapidly calculate an estimated solution. We develop a novel propagation search method for kd-trees. In this method the tree nodes checked by each query are propagated from the nearby queries. Luca Di Gaspero., and Andrea Roli., 2008 present an approach based on the combination of local search metaheuristics and a decrease process based on an analysis of the problem structure. Results on a set of Haplotype Inference benchmarks show that this approach achieves a good trade-off between solution quality and execution time.

Maryam Karimzadehgan., et.Al., 2011 propose a simple yet effectual stochastic optimization algorithm to directly reduce any loss function, which can be defined on NDCG or MAP for the learning-to-rank problem. The algorithm employs Simulated Annealing along with Simplex method for its parameter search and finds the global optimal parameters. Experiment results using NDCG-Annealing algorithm, an instance of the proposed algorithm, on LETOR benchmark data sets show that the proposed algorithm is both effectual and steady when compared to the baselines provided in LETOR 3.0.

Mingjie Lin, and John Wawrzynek., 2010 develops a animatedly adaptive stochastic tunneling (DAST) algorithm to keep away from the "freezing" problem commonly found when using simulated annealing for circuit placement on field-programmable gate arrays (FPGAs). The main objective is to decrease the placement runtime and get better the quality of final placement.

The error probability decays exponentially fast, and we offer bounds for the error exponent. We then focus on the case where the tree has convinced symmetry properties. We obtain the appearance of the optimal exponent within a controlled class of effortlessly implement able strategies, as well as most favorable strategies within that class. Wee Peng Tay., et.AL., 2009 provide confirmation that in designing a network it is preferable to keep the branching factor small for nodes other than the neighbors of the leaves.

Venu Satuluri., and Srinivasan Parthasarathy., 2012 present BayesLSH, a principled Bayesian algorithm for the subsequent phase of similarity search - performing candidate pruning and similarity estimation using LSH. A simpler variant BayesLSH-Lite, calculates similarities exactly. Nenad Tomašev., et.Al., 2011 present a new probabilistic approach to kNN classification, naive hubness Bayesian k-nearest neighbor (NHBNN), which employs hubness for computing class likelihood estimates. Experiments show that NHBNN compares positively to diverse variants of the kNN classifier, including probabilistic kNN (PNN) which is habitually used as a fundamental probabilistic framework for NN classification.

Dewan Md. Farid., et.Al., 2011 initiate a new approach to the classification of streaming data based on bootstrap aggregation (bagging). The proposed approach creates an collection model by using ID3 classifier, naïve Bayesian classifier, and k-Nearest-Neighbor classifier for a learning scheme where each classifier gives the weighted prediction. A memetic algorithm called GSMPSO by combining the PSO with a Gaussian mutation operator and a Simulated Annealing (SA)-based local search operator is urbanized to weight the features for K Nearest Neighbors (KNN) regression by JiaCheng Ni., et.AL., 2012.

To evolve a NN object in heterogeneous environment, Stochastic Annealing Nearest Neighbor Search using hybrid search algorithms (SANN- HA) for spatio-temporal model is presented.

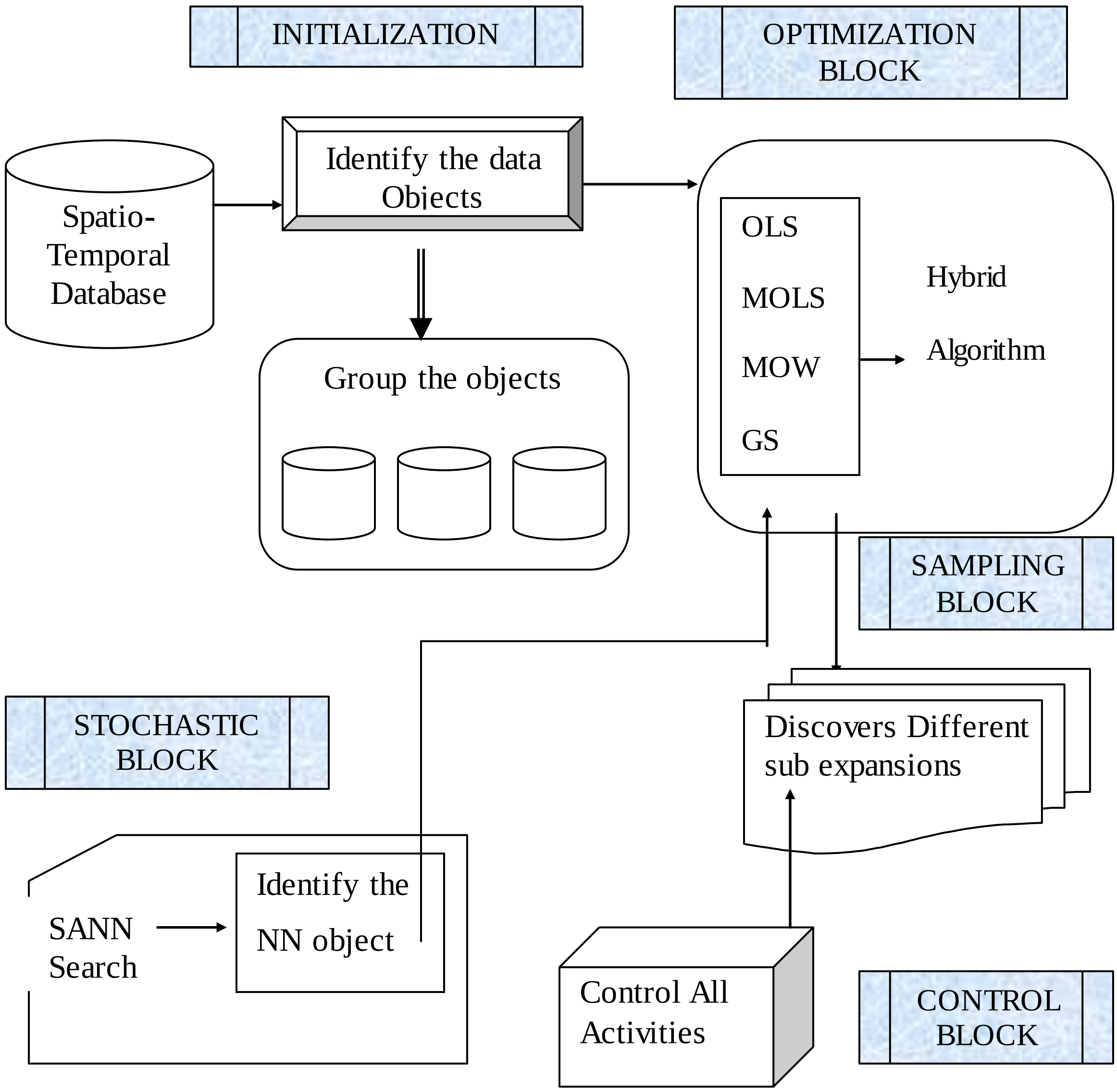
3. PROPOSED STOCHASTIC ANNEALING NEAREST NEIGHBOR SEARCH FOR SPATIO-TEMPORAL HETEROGENOUS ENVIRONMENT

The proposed Stochastic Annealing Nearest Neighbor Search model is process ed under different blocks. Spatio Temporal database is a type of database which contains both the space and time information. The initialization block first initializes the algorithms. The Initialization identifies the data objects and groups the objects a ccording to the similarity measure. The groupings of the similar objects are used to retrieve the information efficiently.

The optimization block efficiently used for combining the algorithm as the hybrid search algorithm. The hybrid search algorithm produces the optimal result by variety of algorithms namely Ordered Local Search, (OLS), Monitor Ordered Local Search, (MOLS), Marginal Object Weight (MOW) and Global Spatial (GS) algorithms to fit the long run distribution in the heterogeneous environment. Ordered Local Search, (OLS) considers the search space as the graph where each solution corresponds to a node having some discrepancy degree.

Monitor Ordered Local Search, (MOLS) combined with the OLS by keeping a memory for all the solution generated. Marginal Object weight (MOW) scheme perform the NN search efficiently for both high and low dimensional data structure by retrieving the information. Global Spatial (GS) algorithms are search methods based on the concepts of usual alteration and the survival of the fittest individuals. The sampling block discovers the different sub expansions from the optimization block hybrid algorithms.

The architecture diagram of the proposed Stochastic Annealing Nearest Neighbor Search using hybrid search algorithms (SANN- HA) for spatio-temporal heterogeneous environment is shown in Fig 3.1.



ig 3.1 Architecture Diagram of Stochastic Annealing Nearest Neighbor Search model

The proposed work is efficiently designed for Stochastic Annealing Nearest Neighbor Search model. The different sub expansions and the varying run length distribution are forbidden by the control block of the system. The Stochastic block contains the Stochastic Annealing Nearest Neighbor search which identifies the nearest object efficiently. SANN consider optimization of a system in which the cost is an arbitrary variable whose allocation depends on the state of the system. Challenging states are prepared according to the mean of the cos t which we calculate approximately by arbitrary sampling. By similarity with simulated annealing, we demonstrate that energetically controlling the ambiguity connected with the cost estimate can improve the degree of optimization obtained, at the same time as providing significant gain in computational efficiency.

3.1. Hybridization of Search techniques

The Hybrid search method combines the more familiar search heuristics such as local and global search, with spatio temporal database in order to rapidly position high-quality optimal solutions. In calculation to their usefulness as standalone methods, they can be employed in combining all the methods as a hybrid search to speed up retrieval of the optimal solutions by orders of magnitude.

3.1.1 Ordered Local Search

Ordered local Search contains the 'm' datasets with same cardinality M, the graph has Mm nodes. If one is able to derive from the other by changing the instantiation of a particular variable, then two nodes are associated through an edge. A variable can take M-1 values for each solution having m· (M-1) neighbors, exclusive of present assignment. A node that has lower discrepancy degree than all its neighbors is a local ceiling. Notice that a local ceiling is not necessarily a global ceiling because there present a solution with superior similarity in other regions of the graph.

Local search methods start with an arbitrary solution called kernel, and then tries to reach a local ceiling by visiting neighbors with higher similarity and uphill moves. When they reach a local ceiling they restart the same process from a dissimilar kernel until the time limit is fatigued.

All the way through this process produces best solutions. Algorithms based on this universal concept have been productively employed for a variety of problems.

3.1.2 Monitor Ordered Local Search

Monitor Ordered Local Search (MOLS) combines the above ideas by maintaining a memory for all the solutions of the variable assignments at local ceiling. When a local ceiling ($l_{w,a},..,l_{x,b}, .., l_{y,c},., l_{z,d}$) is found, some of the assignments $u_w \leftarrow l_{w,a},..$, $u_x \leftarrow l_{x,b},..$, $u_y \leftarrow l_{y,c},.$, $u_z \leftarrow l_{z,d}$, get a consequence. In particular, MOLS discipline the assignments with the smallest amount of consequence so far; e.g., if $u_w \leftarrow l_{w,a},..$, already has a reprimand from a preceding utmost assignments and to avoid the over punishments.

The code for MOLS is similar to OLS since both algorithms re-instantiate the worst variable for improving the current solution. Their difference is that MOLS only generates one arbitrary kernel during its execution and has some extra code for consequence assignment. The consequence is used to increase the irregularity degree of the present local ceiling, and a smaller amount of solutions that include a subset of the assignments. In particular, for its correspondence computations MOLS applies the effective changeability degree which is computed by adding the consequence to the actual inconsistency degree of a solution.

$$\lambda \sum_{i=1}^{n} consequence(u_i \leftarrow l_{i,x})$$

Where,

λ –constant that tunes the relative importance of consequence

It controls the effect of memory in search. A large value of λ will reprimand significantly local ceiling and their neighbors causing the algorithm to quickly visit other areas of the graph. A small value will achieve better investigation of the neighborhoods around maxima at the expenditure of global graph exploration.

3.1.3 Marginal object Weight Scheme

In the spatio temporal databases of objects, the nearest object to the querying point desires to be identified at its optimality to present a discrete and continuous probability of objects. The nearest neighbor object search creates few uncertainty on recognition of instance points at which definite spatial object is closest than its other entire neighbor to the referring point. In order to provide a clear differentiation to the object at an instance, marginal weight is introduced to all the nearest neighbor objects of the reference points. The marginal weight assigned to each object is based on its nearness of reference points at the spatial events.

The marginal weight of object keeps on changing with the numerous dimensional of the spatial events. The marginal weight is calculated based on the instance event generated with objects adjacent to the querying point in terms of frequency of object being at the closest position and the objects distance to the querying point on the spatial database. All the closest objects are assigned with the designed marginal weight and ranking is made based on the chronological order. With the rank assigned for each objects for nearest neighbor search, the object is identified for any specific spatial instances to its optimality without any ambiguity.

3.1.4 Global Spatial Search

Global Spatial (GS) are search methods based on the concepts of standard mutation and the endurance of the fittest individuals. Ahead of the search process starts, a position of C solutions is initialized to outline the first generation. Then, three genetic operations selection, crossover and mutation, are repeatedly applied in order to obtain a population with better characteristics. This set will encompass the next generation, at which the algorithm will execute the identical actions and so on, until a stopping criterion is met. GS takes advantage of spatial indexes and the problem formation to improve solutions.

Selection mechanism: This operation consists of two parts: assessment and issue allocation. Assessment is performed by measuring the similarity of every solution. Issue creation then allocates to each solution, a number of issue proportional to its similarity. Techniques for issue allocation include ranking, proportional selection, stochastic remainder.

Crossover mechanism: It *is* the driving force of exploration in GS algorithms. For each pair a crossover point is distinct arbitrarily, and the solutions beyond it are mutually exchanged with probability by producing two new solutions.

Mutation mechanism: Even though it is not the major search operation and sometimes is omitted, mutation is very important for GS and the only operation that uses the order. At each generation, mutation is applied to every solution in the population with probability μm, called the mutation rate. The process is similar to OLS; the worst variable is chosen and it gets a new value using discover finest value. Thus, in our case mutation can only have positive results.

3.2 Hybrid Search Algorithm in Heterogeneous Environment

```
//Ordered Local Search
Begin
        A: = Arbitrary Kernel
   WHILE NOT (Local Ceiling)
        Determine Variable u i
        Value: = Find finest value (Root l i, u i )
          If (better value)
                Then A = A^ {u i ← value}
          If (S → best solution)
                Then best solution =A
        END WHILE
End
//Monitor Ordered Local Search
Begin
        A: = Arbitrary Kernel
    WHILE NOT(Local_ Ceiling)
        Determine Variable u i
        Value: = Find finest value (Root l i, u i )
          If (better value)
                Then A = A^ {u i ← value}
          If (S → best solution)
                Then best solution =A
        END WHILE
          C = From the current local ceiling, select the ones with the minimum consequence
        FOR EACH assignment u i ← l i, x in C
                Consequence (u i ← l i, x) = consequence (u i ← l i, x) + 1
```

```
    END FOR
End
//Marginal Object Weight
Begin
    Assume NN Objects

            FOR EACH
              Object O do,
             Compute the weight of object w
              Rank each object R, based on Frequency f, Distance d
            END FOR
End
//Global Spatial Search
Begin
    C := generate initial set of solutions {S1,..,Sp}

    WHILE NOT (Instance boundary)
        Compute crossover point c
    FOR EACH
        Si in C /* Assess allocation */
        Assess Si
    IF Si is the best solution found so far
    THEN keep Si
    FOR EACH
        Si in C /* Issue allocation */
        Compare Si with T other random solutions in C
        Replace Si with the best among the T+1 solution
    FOR EACH
        Si in C /*Crossover*/
        Probability µc change Si as follows
        Determine set of c variables to keep their current values
        Re-instantiate the outstanding variables using their values in another solution Sj (Sj ∈ C)
    FOR EACH
        Si in C /* Mutation */
        with probability µm change Si as follows
        Determine worst variable vk
        uk ← Find finest value (Root of tree Lk, uk)

    END WHILE
End
```

The above Hybrid Search Algorithm combines the different types of search and produces the optimization solution for different run length distribution. The Stochastic Annealing Nearest Neighbor Search (SANN) can improve the degree of optimization obtained and at the same time provided a significant gain in computational efficiency.

4. EXPERIMENTAL EVALUATION

The experiments of the proposed Stochastic Annealing Nearest Neighbor Search using hybrid search algorithms (SANN- HA) for spatio-temporal heterogeneous environment is evaluated with the massive spatial- temporal data set, annealing dataset obtained from UCI repository and Global dataset. The experiment is implemented in Java 1.6 SDK and core java concept with over 1200 instances of spatial dataset. We ran our experiments with various data sets obtained from UCI repository.

Spatial information for large spatial and spatio-temporal datasets is demanding. The size of the dataset 'n' causes troubles in computing optimal spatial predictors, since its computational complexity is on the order of the cube of 'n'. In addition, a large dataset is often distinct on a large spatial or spatio-temporal domain, so that the spatial process of attention classically exhibits no stationary performance over that domain. In this section, we develop a sequence of experiments measured to approximate the correctness of the proposed algorithm in terms of

i) Similarity Measure,

ii) Average Time for execution,

iii) Run length distribution.

5. RESULTS AND DISCUSSION

In this work we have seen process of spatio-temporal hybrid search algorithms in heterogeneous environment. The below table and graph describes the performance of the proposed Stochastic Annealing Nearest Neighbor Search using hybrid search algorithms (SANN- HA) model. In the consequence, we compared Stochastic Local Search by Viterbi algorithm (SLS-VA), Normalizing Regions of Data in Spatial Data grid (NRDSD) and Bayesian Nearest Neighbor Search using Marginal Object Weight (BNN-MOW).

Query Size	Similarity measure			
	Proposed SANN-HA	SLS-VA Model	NRDSD Scheme	BNN-MOW
250	99	85	80	75
500	98	85	79	75
750	98	87	81	77
1000	99	89	81	78
1250	98	90	82	78
1500	98	89	82	79
1750	97	87	83	79
2000	98	85	83	78
2250	99	84	81	75

Table 5.1 Query Size vs. Similarity Measure

The above table (table 5.1) describes the similarity measure based on the query size formed with respect to the massive spatio-temporal dataset. The dimensionality of the heterogeneous environment of the proposed Stochastic Annealing Nearest Neighbor Search using hybrid search algorithms (SANN- HA) model, in terms of similarity measure is compared with an existing Stochastic Local Search by Viterbi algorithm (SLS-VA), Normalizing Regions of Data in Spatial Data grid (NRDSD) and Bayesian Nearest Neighbor Search using Marginal Object Weight (BNN-MOW).

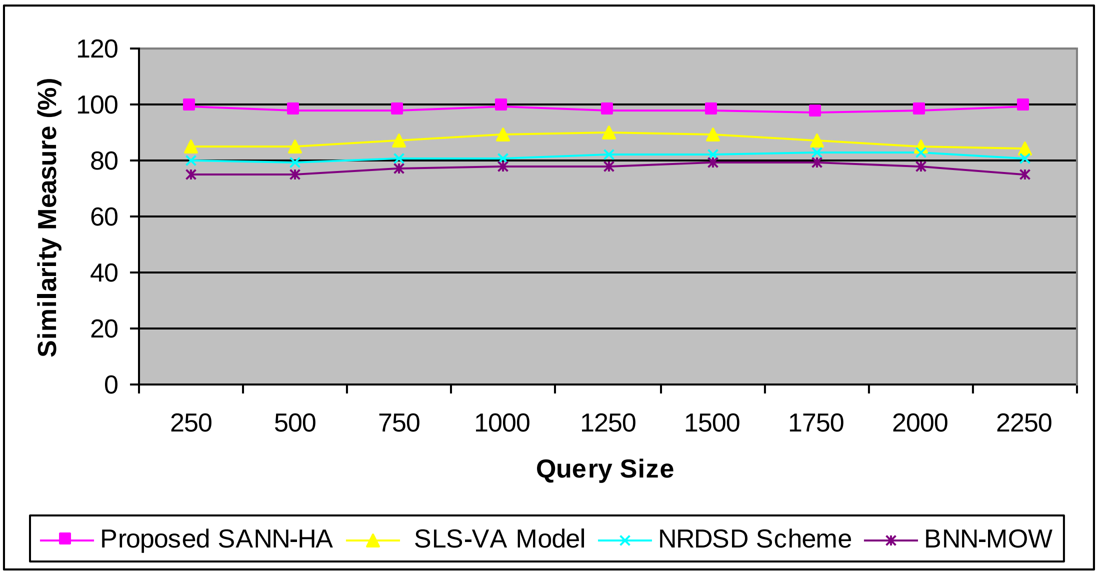


Fig 5.1 Query Size vs. Similarity Measure

Fig 5.1 describes the average similarity measure based on the query size in the massive spatio-temporal dataset. The set of experiments was used here to examine the impact of similarity in fetching the result in the Stochastic Annealing Nearest Neighbor Search using hybrid search algorithms (SANN- HA) model. SANN-HA is capable to complete vastly precise results and its performance is normally reliable with accurate result extraction. As we can see from Fig. 5.1, SANN-HA is more scalable and accuracy in getting the accurate query result and obtains the optimal solution than the existing SLS-VA model, NRDSD Scheme and BNN-MOW.

If the average query size is very low or higher, it provides a efficient search result in SANN-HA. Experiments showed that the proposed SANN- Hybrid Search Algorithm efficiently identifies the result using the difference conditions and its dimensions precisely in a variety of situations. Compared to an existing scheme, the proposed SANN-HA achieved accurate search result and the variance is approximately 10-15% high.

No. of data objects	Average time for execution (sec)			
	Proposed SANN-HA	SLS-VA Model	NRDSD Scheme	BNN-MOW
100	5	35	62	72
200	12	46	95	112
300	18	55	122	150
400	20	62	156	167
500	30	76	182	200
600	32	88	210	245
700	34	92	225	282
800	35	110	267	320
900	36	125	311	345

Table 5.2 No. of data objects vs. Average time for execution

The above table (table 5.2) describes the presence of time taken to execute based on the number of data objects with respect to the Global dataset and annealing dataset. The execution time of the objects of the proposed Stochastic Annealing Nearest Neighbor Search using hybrid search algorithms (SANN- HA) model, in terms of calculating the average time for execution is compared with an existing Stochastic Local Search by Viterbi algorithm (SLS-VA), Normalizing Regions of Data in Spatial Data grid (NRDSD) and Bayesian Nearest Neighbor Search using Marginal Object Weight (BNN-MOW).

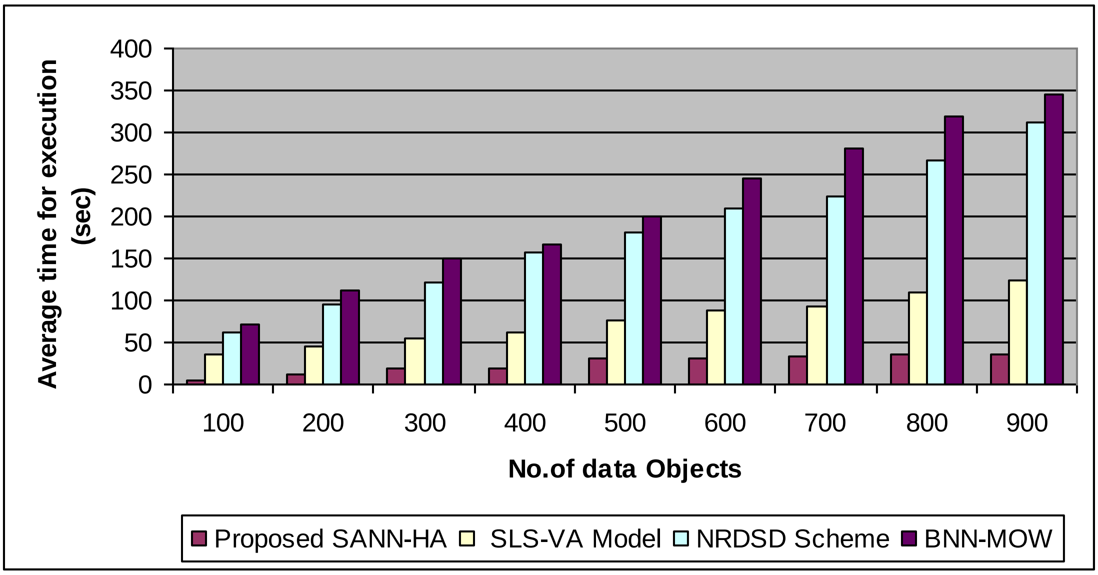


Fig 5.2 No. of data objects vs. Average time for execution

Fig 5.2 describes the presence of time to execute based on the number data objects with respect to the Global dataset and annealing dataset. As observed from the figure 5.2, SANN-HA exhibit reliable performance with lesser execution time from the first set of experiments on data sets taken. In tricky cases, SANN-HA presents much improved results than the existing SLA-VA model, NRDSD Scheme and BNN-MOW. The results stated in Fig. 5.2 recommend that the proposed SANN-HA is more interested to the proportion of data sets in execution time parameter.

Fig 5.3 describes the consumption of time to perform the efficient search on the nearest neighbor nodes. The dataset used for the experiments linearly balances the attributes and produce the result efficiently. The execution time is averagely less in the SANN-HA when compared with the other schemes of Global dataset and annealing dataset. The time consumption is measured in terms of seconds. Compared to the existing SLS-VA model, NRDSD Scheme, and BNN-MOW model, the proposed SANN-HA consumes less time since it gives better similarity result and the variance in time consumption is approximately 30-35% low in the proposed SANN-HA.

No. of operations	Run Length distribution			
	Proposed SANN-HA	SLS-VA Model	NRDSD Scheme	BNN-MOW
10	0.010	0.05	0.05	0.01
100	0.15	0.10	0.05	0.02
1000	0.35	0.20	0.05	0.02
10000	0.40	0.25	0.10	0.05
100000	0.55	0.40	0.15	0.10

1000000	0.60	0.50	0.20	0.15
10000000	0.75	0.55	0.20	0.15
100000000	0.95	0.70	0.35	0.15

Table 5.3 No. of operations vs. Run Length distribution

The above table (table 5.3) describes the run length distribution with respect to the massive spatio-temporal dataset dimensionality. The run length distribution with respect to the operations performed in the proposed Stochastic Annealing Nearest Neighbor Search using hybrid search algorithms (SANN- HA) model, in terms of calculating the average time for execution is compared with an existing Stochastic Local Search by Viterbi algorithm (SLS-VA), Normalizing Regions of Data in Spatial Data grid (NRDSD) and Bayesian Nearest Neighbor Search using Marginal Object Weight (BNN-MOW).

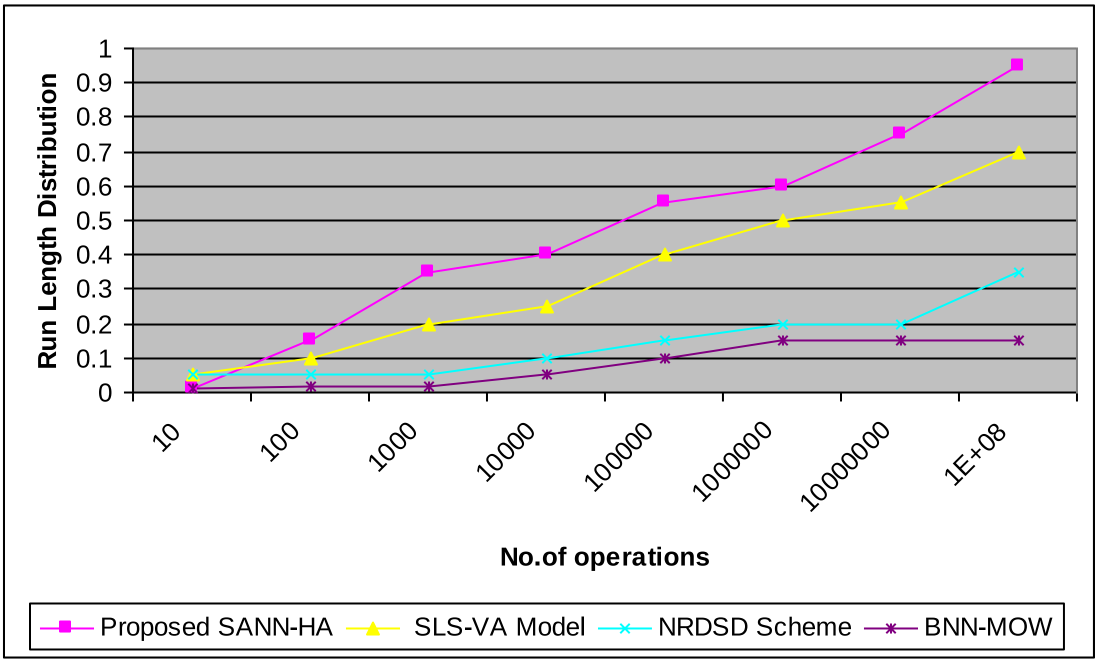


Fig 5.3 No. of operations vs. Run Length distribution

Fig 5.3 describes the run length distribution with the help of the operations performance. In the SANN-HA method, the massive spatio temporal dataset used to find the run length distribution. As the hybrid algorithms is used in our proposed scheme to get an optimal nearest neighbor objects. Compared to an existing SLS-VA, NRDSD scheme and BNN-MOW, the proposed SANN-HA provides the efficient run length in terms of speed formation and the variance in is approximately 5-12% higher in the proposed SANN-HA method.

6. CONCLUSION

In this work, we competently attain the heterogeneous mixture by processing the spatio-temporal database in Massive spatio-temporal Data Set, Annealing Dataset and global data set by professionally introducing the Stochastic Annealing Nearest Neighbor Search using hybrid search algorithms (SANN- HA) model. The proposed scheme describes the combining of different search algorithms to obtain the optimal solution in heterogeneous environment. We compare SANN-HA model with the MPE-BN, normalizing regions of data in spatial data grid (NRDSD) and BNN-MOW schemes in terms of similarity search, average time for execution and Run length distribution. The experimental results showed that the proposed SANN-HA scheme for the heterogeneous environment worked efficiently by improving 20 – 25 % accuracy and less execution time. It discovers different sub explanations using diverse mixture of algorithms in terms of run length distribution and average time for execution based on data objects. The proposed method provides a high quality optimal nearest neighbor solution.

REFERENCES

1. A.N.M. Bazlur Rashid., Md. Anwar Hossain., “Challenging Issues of Spatio-Temporal Data Mining,” Computer Engineering and Intelligent Systems ISSN 2222-1719, ISSN 2222-2863, Vol 3, No.4, 2012
2. Dewan Md. Farid., Mohammad Zahidur Rahman., Chowdhury Mofizur Rahman., “An Ensemble Approach to Classifier Construction based on Bootstrap Aggregation,” International Journal of Computer Applications (0975 – 8887) Volume 25– No.5, July 2011.
3. JiaCheng Ni., Fei Qiao., Li Li., Qi Di Wu., “A memetic PSO based KNN regression method for cycle time prediction in a wafer fab,” Intelligent Control and Automation (WCICA), 2012
4. Kaiming He., Jian Sun., “Computing Nearest-Neighbor Fields via Propagation-Assisted KD-Trees,” Microsoft Research Asia, 978-1-4673-1228-8/12/$31.00 IEEE 2012
5. Luca Di Gaspero., and Andrea Roli., “Stochastic local search for large-scale instances of the Haplotype Inference Problem by Parsimony,” Science direct, 2008
6. Maryam Karimzadehgan., Wei Li., Ruofei Zhang., Jianchang Mao., “A Stochastic Learning-To-Rank Algorithm and its Application to Contextual Advertising,” ACM 978-1-4503-0632-4/11/03, 2011.
7. Mingjie Lin, and John Wawrzynek., “Improving FPGA Placement with Dynamically Adaptive Stochastic Tunneling,” IEEE TRANSACTIONS ON COMPUTER-AIDED DESIGN OF INTEGRATED CIRCUITS AND SYSTEMS, VOL. 29, NO. 12, DECEMBER 2010
8. Nenad Tomašev., Miloš Radovanovi´c., Dunja Mladeni´c., “A Probabilistic Approach to Nearest-Neighbor Classification: Naive Hubness Bayesian kNN,” ACM Transaction, 2011.
9. Perez, M., Marwala, T., “Microarray data feature selection using hybrid genetic algorithm simulated annealing,” IEEE Convention of Electrical & Electronics Engineers in Israel (IEEEI), 2012
10. Saed Alazamir., Steffen Rebennack., and Panos M. Pardalos., “Improving the Neighborhood Selection Strategy in Simulated Annealing using Optimal Stopping Problem in Global Optimization: Focus on Simulated Annealing,” Cher Ming Tan (Ed.), I-Tech Education and Publication, pp. 363–382, 2008
11. Saed Alizamir., Steffen Rebennack., Panos M. Pardalos., “Improving the Neighborhood Selection Strategy in Simulated Annealing using the Optimal Stopping Problem,” Education and Publication, pp. 363–382, 2008
12. Venu Satuluri., and Srinivasan Parthasarathy., “Bayesian Locality Sensitive Hashing for Fast Similarity Search,”Proceedings of the VLDB Endowment, Vol. 5, No. 5, 2012
13. Wee Peng Tay., John N. Tsitsiklis., and Moe Z. Win., “Bayesian Detection in Bounded Height Tree Networks,” IEEE TRANSACTIONS ON SIGNAL PROCESSING, VOL. 57, NO. 10, OCTOBER 2009