International Journal of Research in Advent Technology, Vol.6, No.6, June 2018
E-ISSN: 2321-9637
Available online at www.ijrat.org

Simple Hadoop Map Reduce function on Parallel Implementation using Genetic Algorithms

N. Revathi[1], Dr. P. Sengottuvelan[2]
[1]*Ph. D Research Scholar[FT] Department of Computer Science Periyar University PG Extension Centre, Dharmapuri, India.*
[2]*Associate Professor & Head Department of Computer Science Periyar University PG Extension Centre, Dharmapuri, India.*

Abstract-We show parallel usage of hereditary calculation utilizing Map Reduce programming worldview. Hadoop execution of guide/lessen library is utilized for this reason. We contrast our execution and usage introduced in. The correlation criteria between usages are wellness joining, nature of conclusive arrangement, calculation versatility, and cloud asset use. Our model for parallelization of hereditary calculation indicates preferable exhibitions and wellness joining over model displayed in [1], yet our model has bring down nature of arrangement in view of species issue. Straightforward Genetic Algorithms are utilized to take care of advancement issues. Hereditary Algorithm additionally accompanies a parallel usage as Parallel Genetic Algorithm (PGA). PGA can be utilized to lessen the execution time of SGA and furthermore to take care of bigger size occurrences of issues. In this paper, distinctive executions for PGA have been talked about with their systems. In this execution, all PGA depend on a solitary SGA system. These are executed on a parallel machine and tried on some benchmark issue occurrences of Traveling Salesman issue (TSP) from TSPLIB. TSPLIB is an outstanding library for informational collection of benchmark issue occasions. A fundamental structure has been proposed for executing PGA on the present parallel PCs.

General Terms
Genetic Algorithm, Parallel Genetic Algorithm, Traveling Sales representative Problem
Keywords
Streamlining, Parallel Genetic Algorithm, Straightforward Genetic Algorithm, Traveling Salesman Problem

1. INTRODUCTION

Genetic Algorithm is a populace based meta-heuristic inquiry method comprising of following administrators: Initialization, Selection, Reproduction and Replacement [1] [2] [3] [4]. Right off the bat, the populace is produced arbitrarily in any case. At that point choice administrator finds the fittest individuals. This administrator takes after the "survival of the fittest" rule. These fit individuals are utilized as a part of proliferation to make new offspring's. Hybrid and Mutation are utilized as a part of generation stage. Hybrid makes new youngster from guardians and change is utilized to modify the chromosomes to expel the issues of hereditary float and so on. At last substitution is utilized to deal with the old and new arrangement of Chromosomes to repeat for assist ages. GA regards touch base at bowls of fascination in an expansive arrangement space. What's more, to decrease the huge measure of calculation time PGA can be utilized. Voyaging Salesman Problem is a directing issue with numerous conceivable arrangements. In any case, the issue must follow the optimality of arrangements. This optimality prompts the scourge of dimensionality. It is an outstanding NP-Hard combinatorial advancement issue [5]. It can be portrayed as a gathering of N number of urban areas and one businessperson, which needs to visit each city precisely once and come back to the beginning city from the accumulation of urban areas. The objective is to locate the ideal/least cost way. The quantity of arrangements in a N-city issue will shift from (N-1)! To N!, which turns out to be more regrettable soon with extensive estimations of N. This paper is composed in following segments. Segment 2, insights about Simple Genetic Algorithm is given. In area 3, Parallel Genetic Algorithm has been examined. The execution points of interest of SGA and PGA are given in area 4. At last, segment 5 shows the conclusions and discoveries of the paper.

1.1. Map Reduce Model

Guide/Reduce demonstrates is first time proposed by Google [3] in 2004 and it is motivated by utilitarian dialects like List. Guide/Reduce

demonstrates speak to rearranged method for parallelization for all projects which are composed in Map/Reduce soul. In Map/Reduce programming worldview, the essential unit of data is a (key; esteem) combine where each key and each esteem are parallel strings. The contribution to Map/Reduce calculation is set of (key; esteem) sets. Tasks on an arrangement of sets happen in three phases: the guide organize, the rearrange arrange and the decrease organize as appeared on figure 1.

In the guide arrange, the mapper takes as information a solitary (key; esteem) match and creates as yield any number of new (key; esteem) sets. It is significant that the guide activity is stateless - that is, it works on one sets at any given moment. This considers simple parallelization as various contributions for the guide can be prepared by various machines.[9]

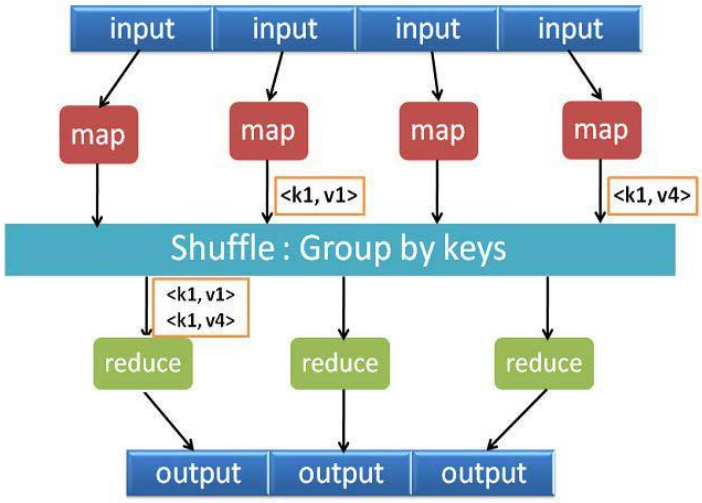

Figure 1: Operation stages in Map/Reduce programming model [8]

Amid the rearrange stage the fundamental framework that actualizes Map/Reduce sends the greater part of the qualities that are related with an individual key to a similar machine. This happens consequently, and is consistent to the software engineer. In the diminish arrange, the reducer takes the majority of the qualities related with a solitary key k, and yields a multi set of (key; esteem) sets with a similar key, k. This features one of the successive parts of Map/Reduce calculation:

2. BASIC GENETIC ALGORITHM

GA is a versatile and heuristic based hunt system which can be utilized to look from the inquiry space. GA is proposed by John Holland [1] at University of Michigan in 1975. These are additionally depicted as versatile heuristic hunt calculations [2] in view of the transformative thoughts of normal choice and regular hereditary qualities by David Goldberg. GA gains ground toward the ideal arrangement by consolidating better and better arrangements in every age to make all the more better arrangements [5] [7]. Each single arrangement is spoken to by chromosomes, which will be assessed for the wellness of that arrangement. This wellness is utilized as a guide about the arrangement quality. This procedure keeps on accomplishing the ideal arrangement. General structure of genetic algorithm is:

Procedure SGA (fitness, pop_size, pc, pm, no_of_generations)

// *fitness* is the fitness function used to evaluate chromosomes in population

// pop_size is the population size in each generation (say 1000)

// pc is the probability of crossover (say 0.90)

// pm is the mutation rate (say 0.001)

// no_of_generations is total number of generations

pop = generate pop_size individuals randomly to

start with

gen_number =1 //denotes current generation

while gen_number <= no_of_generations do
{

L = Select(pop, pop_size, nogen)

S = Crossover(L, pop_size, pc)

M = Mutation(S, pop_size, pm)

pop = M

gen_number = gen_number + 1;

}
end proc

2.1.*Scaling Measurements On Hadoop Cluster*

In this segment we are trying exhibitions of two unique executions of hereditary calculations. For those reasons we have been utilizing 10 hubs bunch (i7 - 4 centers 2.6 GHz, 4GB
DDR3 RAM, 300GB HDD) with CentOS 5.6 working framework and Hadoop CDH3u0 dispersion. We have performed two trial of the two models are thought about outcomes

- Convergence of hereditary calculation with steady number of guide decrease assignments,

- Scalability of hereditary calculation with steady load per hub in clust

Convergence of genetic algorithm with constant number of map reduce tasks

With this test we have track best wellness in populace of One Max issue over cycles of hereditary calculation and looked at comes about because of the two models. Aftereffects of examination are introduced on figure 4.

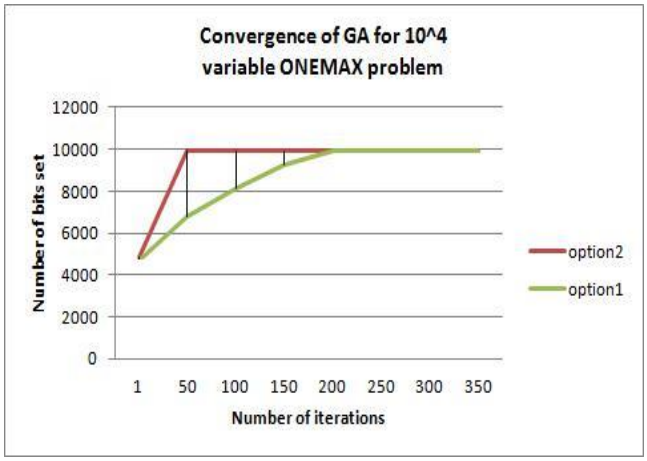

Figure 1: Comparison of two models of parallel GA from arrangement union angle; choice 1 – All hubs utilizes same populace; choice 2 – every hub has its own populace

In this test parameters of hereditary calculation are:

- crossover likelihood = 0.7
- mutation likelihood = 0.01
- population measure = 10000
- number of mappers/reducers = 30

As results demonstrate choice 2 has better merging of wellness since it has numerous mappers, which are chipping away at various populaces, which causes that arrangement is discovered considerably before.

Scalability of genetic algorithm calculation with consistent load per hub in group

In second test we looked at versatility of two models portrayed in this paper. Consequences of examination are introduced on figure 5.

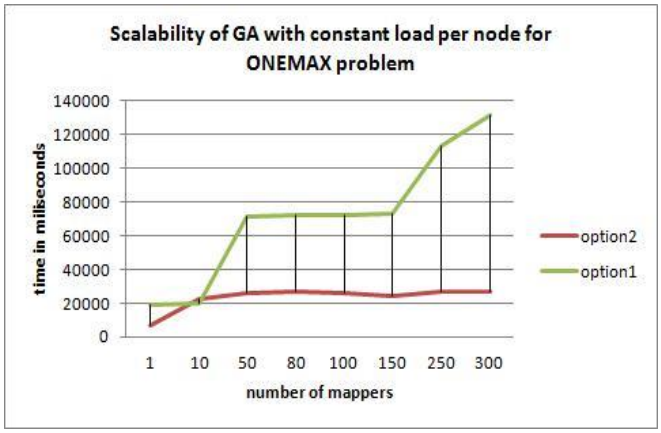

Figure 2: Comparison of two models of parallel GA from adaptability viewpoint; alternative 1 – All hubs utilizes same populace; choice 2 – every hub has its own particular populace

In second test parameters of hereditary calculation are:

- crossover likelihood = 0.7
- mutation likelihood = 0.01
- population measure = 10000
- number of factors for One Max issue = 10000
- number of cycles = 500

As exhibited on figure 5 choice 2 is considerably speedier than choice 1 since IO impression is decreased on the grounds that information isn't composed to HDFS. Choice 1 and 2 can scale to interminability by including more equipment assets into cloud.

Affirmations of individuals, gifts, reserves, and so on ought to be put in a different area before the reference list. The names of subsidizing associations ought to be composed in full. Try not to put affirmation on the main page of your paper or as a reference.

3. PARALLEL GENETIC ALGORITHM

It is hard when all is said in done to list the parallel PC engineering, as it has an expansive history. Beginning from Pipelining to Vector designs, from Dual center machines to current i3, i5, i7 multi-center models and from single CPU to circulated frameworks and cloud PCs. There are two sorts of parallel GAs that can misuse these sorts of parallel designs viably: various populace GAs (likewise called coarse grained or island demonstrate GAs) and ace slave (or worldwide) parallel GAs.

GA is extremely perplexing calculation that is controlled by numerous parameters and its prosperity depends to a great extent on setting these parameters sufficiently. The issue is that no single arrangement of parameter esteems will bring about a powerful and proficient calculation in all arrangements [3] [6]. Consequently, the calibrating of a GA to a specific application is as yet a craftsmanship and science. Ace slave GA is a straightforward GA that disperses the assessment of the populace among a few processors. Though island-show GA every processor begins with an autonomous populace.

3.1. A Parallel Genetic Algorithm Based On

Hadoop Map Reduce In the accompanying subsections we first give a few foundation, giving an account of the systems proposed in the writing to parallelize Genetic Algorthms and reviewing the main parts of MapReduce and Hadoop MapReduce. Then,we display the outline of the proposed parallel Genetic Algorithm utilizing Hadoop MapReduce.

A. Parallelization Strategies

A few GA parallelization systems exist contingent upon the grain of parallelization to accomplish. Fundamentally, three levels of parallelization can be abused:

_ wellness assessment level (i.e., worldwide parallelization display);

_ populace level (i.e., coarse-grained parallelization or then again island display);

4. EXECUTION AND RESULTS

The greater parts of the calculations are actualized on an IBM Xeon Dual Core Server with 60GB RAM and 1TB Hard Disk. SGA has been actualized as the calculation expressed in area 2. Different elements of GA are utilized as underneath for actualizing it to improve TSP.

- Population Size: 1000
- Initialization: Random
- Selection: Roulette-Wheel Selection
- Crossover: Partial Matched Crossover with 0.9 probabilities.
- Mutation: Swapping with 0.01 probabilities.
- Replacement: Simple substitution (λ, μ).
- Termination: When the best arrangement not enhanced for 100 ages.

Parallel Genetic Algorithms are actualized utilizing the parallel processing tool stash of MATLAB. Parallel Registering Toolbox™ permits the sharing of work inside MATLAB customers. Various customers can be made with the assistance of mat lab pool charge. This pool can misuse the quantity of centers in a processor and hyper threading moreover. It will make the same number of customers of MATLAB execution motor as there are virtual processors. These numerous laborers can be utilized to do various undertakings in the meantime. A neighborhood specialist can be utilized to keep MATLAB customer session free for intelligent work. Additionally the MATLAB Distributed Computing Server permits. Laborers on a remote group of PCs as the permitting of MATLAB permits. In this usage, the primary for circle of ages is supplanted with the par for circle gave by the MATLAB tool compartment.

To intelligently run code that contains a parallel circle, right off the bat MATLAB pool must be opened.It holds an accumulation of laborers to run the circle in parallel. The MATLAB pool can comprise of MATLAB sessions running on nearby machine or on a remote group: It will open a pool of three laborers in the machine. When you are done with your code, close the MATLAB pool and discharge the laborers

Besides PGA is executed as Map Reduce system. Map Reduce is a programming method for breaking down informational collections that don't fit in memory. Map Reduce is a programming model to process huge datasets and make utilization of figuring assets of every server's CPU. It includes two stages: Guide stage and Reduce stage.

In Map stage Mapper must have the capacity to ingest the info and process that information record and afterward that prepared record is sent to reduce stage, where errand will be decreased. The Mapper takes in a key/esteem match and creates middle of the road key/esteem sets [11]. The reducer blends every one of the sets related with a similar moderate key and creates the last yield that is rundown of

key/values. Each activity must contain one guide work took after by discretionary diminish work, these means need to take after this specific request.

Vast occasions of TSP produce loads of information in the middle of the arrangement. MATLAB® gives an execution of the Map Reduce method with the map reduce work. This usage is however somewhat unique in relation to Hadoop Map Reduce. Map reduce utilizes an extraordinary conceptual information compose called data store. It is utilized to process information in little lumps which fits in PC RAM. Every one of these lumps experiences the guide and decrease capacities. These capacities are accessible as guide and lessen in MATLAB.

To take care of any issue with map reduce, one needs to compose these two capacities for the information and after that these are passed as contributions to the principle map reduce work. There are unlimited mixes of guide and decrease capacities to process information, so this system is both adaptable and to a great degree intense for handling expansive information preparing assignments. To actualize Map Reduce usefulness in MATLAB for this examination, following advances are taken after:

Data Preparation: Firstly the populace is assessed for wellness.

Map and Reduce Functions: Selection administrator finds the best people for propagation. The contributions to the guide work are information and interm KV Store. Where information is the after effect of a call to the read work on the info data store. Interm KV Store is the name of the middle of the road Key-Value-Store question which the guide work needs to include key-esteem sets.

Run Map Reduce: After having a data store, a guide work, and a lessen work, map reduce work is called to play out the estimation. i.e.

outpop = mapreduce(ds, @TSPMapFun, @TSPReduceFun);

****

*MAP - REDUCE PROGRESS *

****

Guide 0% Reduce 0% Map 14% Reduce 0% Map 34% Reduce 0% Map 58% Reduce 0%

Guide 78% Reduce 0%

Guide 96% Reduce 0%

Guide 100% Reduce 100%

View Results: Read all capacity can be utilized to peruse the key-esteem sets from the yield data store. i.e. read all (out pop). In the wake of playing out these means the resultant chromosomes from out pop are utilized for assist computations. These three methodologies (SGA, PGA with par for circle and PGA with map reduce) are executed for 4 TSP cases from TSPLIB i.e. Eil51, Eil101, A280, and Oliva30. The outcomes for every one of the three are introduced in beneath tables and figures.

Table 1: Optimum Tour Cost per generation for all implementation

TSP Instance	Eil51	Eil101	A280	Oliva30
Known Optimum	424	768	658	587
SGA	629	1045	919	768
PGA with par for	2579	4320	3423	3145
PGA with Map Reduce	423	657	611	512

Table 2: CPU Time (in msec) taken by all implementations

TSP Instance	SGA	PGA with par for	PGA with MapReduce
Eil51	5184	4563	3060
Eil101	7466	6352	5340
A280	17468	11754	8623
Oliva30	4378	4161	3545

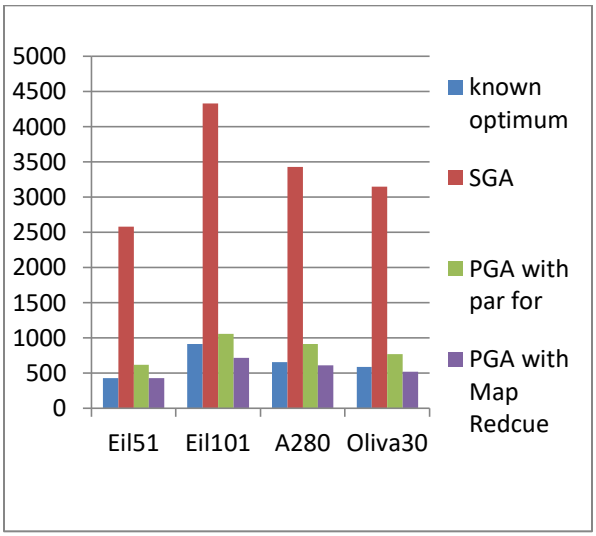

Figure 1: Optimum Tour cost found by usage

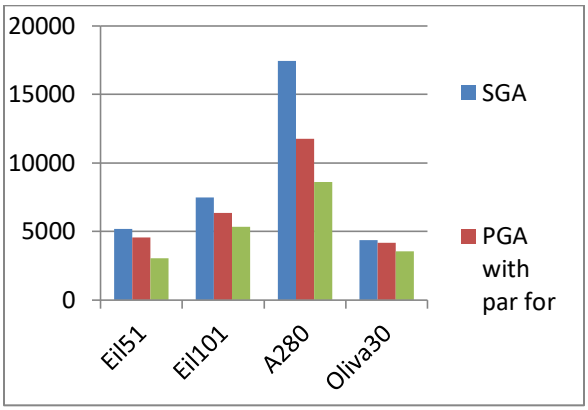

Figure 2: CPU time (in msec) for all executions

The above outcomes demonstrate that the execution of GA in all angles can achieve near the known ideal answers for all examples. Further, PGA as Map Reduce execution is greatly improved in both quality and proficiency terms in contrast with other two usage.

5. CONCLUSIONS

GA is promising calculation regarding meta-heuristics and worldwide streamlining calculations. They are utilized to tackle numerous intricate issues, particularly from the NP-Complete classification. Along these lines, it is dependably an issue to configuration better and better GA to take care of a specific class of issues. In the event that GA can be actualized in parallel then the equipment design can be misused productively. As the intrinsic idea of GA is parallel, so it is additionally persuading that if SGA works better in contrast with other advancement calculations, at that point PGA will doubtlessly do. In this paper, PGA is executed utilizing two methodologies with MATLAB on Intel Xeon Quad Core CPU and contrasted with SGA with unravel TSP. It has been watched that the execution time decreases nimbly with the presentation of parallelization in GA. The ideal visit cost found by PGA is substantially less is correlation with SGA. Additionally SGA sets aside greater opportunity to take care of an indistinguishable size issue in contrast with PGA from appeared. PGA actualized in this work depends on outline design, which can be additionally misused to work with more disseminated and cloud based structures. Likewise different methodologies can be joined with PGA in its execution like Local hunt or other worldwide inquiry.

REFERENCES

[1] Holland J., (1975), Adaptation in natural and artificial systems, University of Michigan Press, Ann Arbor.

[2] Rakesh Kumar, Girdhar Gopal, Rajesh Kumar, (2013), "Hybridization in Genetic Algorithms", International Journal of Advanced Research in Computer Science and Software Engineering (IJARCSSE), Vol-3, Issue-4, pp 403-409.

[3] Goldberg D. E., (1989), Genetic algorithms in search, optimization, and machine learning, Addison Wesley
Longman, Inc., ISBN 0-201- 15767-5.

[4] Rakesh Kumar, Girdhar Gopal, Rajesh Kumar, (2013), "Novel Crossover Operator for Genetic Algorithm for Permutation Problems", International Journal of Soft Computing and Engineering (IJSCE), Vol. 3, Issue-2, pp 252-258.

[5] Moscato P., Cotta C., (2003), "A gentle introduction to memetic algorithms", Handbook of Metaheuristics, pp 105-144.

[6] P. Jog, J. Y. Suh, and D-, Van Gucht, "The effects of population size, heuristic crossover and local improvement on a genetic algorithm for the traveling salesman problem," 3rd Int'l Conference on Genetic Algorithms, july 1989, pp. 110-115

[7] Bosworth Jack, Foo Norman, and Zeigler Bernard P. (1972), "Comparison of Genetic Algorithms with Conjugate Gradient Methods". Technical Report 00312-1-T, University of Michigan: Ann Arbor, MI, USA.

[8] Bethke Albert Donally. (1980), "Genetic

Algorithms as Function Optimizers". PhD thesis, University of Michigan: Ann Arbor, MI, USA. International Journal of Computer Applications (0975 – 8887) Recent Innovations in Computer Science and Information Technology

[9] Brady R. M. (1985), "Optimization Strategies Gleaned from Biological Evolution." Nature, 317(6040): 804– 806, doi: 10.1038/317804a0.

[10] Sinha Abhishek and Goldberg D.E. (2003), "A Survey of Hybrid Genetic and Evolutionary Algorithms". IlliGAL Report 2003-2004, Illinois Genetic Algorithms Laboratory (IlliGAL), Department of Computer Science, Department of General Engineering, University of Illinois at Urbana-Champaign: Urbana-Champaign, IL, USA.

[11] Dean J., Ghemawat S., "MapReduce: simplified data processing on large clusters", Communications of ACM51 (2008) 107–113.

[12] Scaling Genetic Algorithms using MapReduce - Abhishek Verma, XavierLlor'a, David E. Goldberg, Roy H. Campbell

[13] Adapting scientific computing problems to clouds using MapReduce - Satish Narayana Srirama, Pelle Jakovits, Eero Vainikko

[14] MapReduce: Simplified Data Processing on Large Clusters Jeffrey Dean and Sanjay Ghemawat

[15] Scaling Populations of a Genetic Algorithm for Job Shop Scheduling Problems using MapReduce - Di-Wei Huang, Jimmy Lin

[16] Scheduling divisible MapReduce computations - J. Berliñska, M. Drozdowski

[17] MRPGA: An Extension of MapReduce for Parallelizing Genetic Algorithms - Chao Jin, Christian Vecchiola and Rajkumar Buyya

[18] Parallel Genetic Algorithms R.Shonkwiler Map Reduce web page http://hadoop.apache.org/mapreduce/

[19] A Model of Computation for MapReduce - Howard Karlo, Siddharth Suri, Sergei Vassilvitskii

[20] Raghuraman, R., Penmetsa, A., Bradski, G., and Kozyrakis, C. Evaluating mapreduce for multi-core and multiprocessor systems. Proceedings of the 2007 IEEE 13th International Symposium on High Performance Computer Architecture.