The Review of Contemporary Scientific and Academic Studies
An International Multidisciplinary Online Journal
www.thercsas.com

ISSN: 2583-1380 Vol. 3 | Issue No. 7 | July 2023 Impact Factor: 4.736 (SJIF)

Estimation in Deregulated Environment by using Spark and Big Data Analytics for Real Power Tracing

Venugopal Reddy Modugu (modhuguvenu@gmail.com), Corresponding Author
Senior Software Engineer, Idol Soft Inc, USA
Hemish Veeraboina (hemishv111@gmail.com), San Jose State University, San Jose, CA, USA
Dr. V. Punnaiah (punnaiah@cdfd.org.in), In-charge-Engineering (E), CDFD, India
Dr. Syam Sunder (sslingala@gmail.com), Assistant Professor, Prince Mohammad Bin Fahd University, Saudi Arabia

Abstract: *Tracing real power in a deregulated environment is particularly challenging, because of unbundling restrictions and an expanding number of market participants. This study suggests utilizing Spark analysis and Bigdata to perform actual power tracing. For predicting the future power system for system planning, the load sharing of generators is calculated using the RED method. Spark is a ground-breaking data analytics and processing platform for the big data era. Spark is far more effective than traditional computing at tracing real power*

Keywords: Big Data, Deregulated Environment, Real Power Tracing, RED Method, Spark

Article History: **Received: 6 July 2023;** **Accepted: 20 July 2023;** **Published/Available Online: 30 June 2023;**

Introduction

In a deregulated electricity environment, it is now crucial to analyze the effects of a certain generator on the power system. This problem can be resolved by using power tracing techniques, which can determine exactly how much of a generator's output goes toward certain loads or line losses [1]. In a deregulated market, there are more buyers and sellers [2]. The new model will restructure some areas of planning and operation, but the fundamental concepts will not change. The challenge of tracing real electricity is more difficult in an open market, deregulated environment since there are more buyers and sellers. Real power tracing helps in locating and comprehending the sources and destinations of the power in the network, requiring the distribution of transmission costs [3,4]. It is highly challenging to identify real power because of unbundling regulations in a free-market environment and contracted and uncontracted demands of consumers and DISCOMS [5]. In an open market, it is essential to comprehend the importance of actual power tracing for the optimal generation schedule and future power system forecasting for system planning [6]. The quantity of power flowing across the lines will be calculated by the network's power flow calculations, and a power tracing method will offer a quantitative study of the power involving a network component, whether it be power produced by a generator or received by a load point [7]. The power system operator will benefit directly from these assessments by being able to improve the current tariffs and transmission/distribution services.

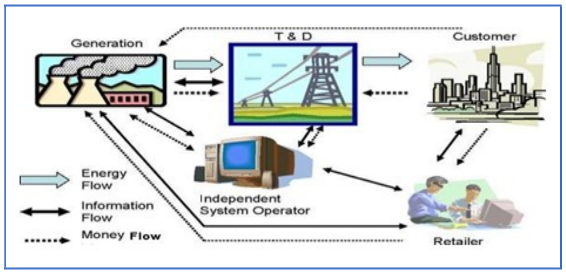

Figure1: Structure of deregulated power system

The computation of the optimal power flow is time and resource intensive. A rapid and effective procedure that is necessary close enough to the ideal solution is required in order for operators to act quickly and effectively in both normal and emergency power system conditions [8]. This problem has been addressed by power system researchers using a variety of power tracing algorithms like Pro rata (PR) procedure Marginal procedures, proportional sharing principle, RED, etc. Traditional or conventional computing is used for the majority of power tracing computation studies. There are not many articles available related to this topic in distributed computation. This paper's goal is to accurately calculate the contributions of generators to

ISSN: 2583-1380 Vol. 3 | Issue No. 7 | July 2023 Impact Factor: 4.736 (SJIF)

transmission lines utilizing big data instead of conventional computing. This paper discusses the RED approach [9], and the big data spark framework [10] handles the calculation. A large amount of generated data makes traditional computations for power tracing quite time-consuming. Analyses are delayed as a result. Here, the Spark, Big data framework is helpful, to process data to calculate outcomes considerably with greater speed. Section II discusses the RED Method, while Section IV discusses big data analysis using the Spark framework.

Computation of RED Method

This method [11] was created using the admittance matrix of the transmission network. Consider the total number of buses in the system is n with 1, 2, …, g, g number of generator buses, and $g+1, \ldots, n$, remaining $(n-g)$ buses. The Y-bus matrix is represented by

$$\begin{bmatrix} IG \\ IL \end{bmatrix} = \begin{bmatrix} YGG & YGL \\ YLG & YLL \end{bmatrix} \begin{bmatrix} VG \\ VL \end{bmatrix} \qquad 1$$

where $\boldsymbol{I}_G$, $\boldsymbol{I}_L$, and $\boldsymbol{V}_G$, $\boldsymbol{V}_L$ Y_{GG} represent complex current and voltage vectors at the generator nodes and load nodes. Y_{GG}

Where

$[I_G] = [I_1 .....I_G]^t$, represent the complex current of the generator buses

$[I_L] = [I_{g+1} ....I_N]^t$, represent the complex current of the load buses

$[V_G] = [V_1 .... V_g]^t$, represent the complex voltages of the generator buses

$[V_G] = [V_{1+g} .... V_n]^t$, *represent the* complex voltages of the load buses

$[Y_{GG}]$, $[Y_{LG}]$, $[Y_{GL}]$, and $[Y_{LL}]$ represent the corresponding portions of the network Y-Bus matrix.

Above eq 1 can be written as

$$[I_G] = [Y_{GG}]*[V_G]+[Y_{GL}]*[V_L] \qquad 2$$

$$[I_L] = [Y_{LG}]*[V_G]+[Y_{LL}]*[V_L] \qquad 3$$

Eq 3 can be modified as

$$[Y_{LL}]^{-1}[I_L] = [Y_{LL}]^{-1}*[V_G]*[Y_{GL}]+[V_L] \qquad 4$$

$$[V_L] = [Y_{LL}]^{-1}[I_L]-[Y_{LL}]^{-1}*[V_G]*[Y_{GL}] \qquad 5$$

Substituting $[V_L]$ of eq 5 in eq 2

$$[I_G] = [Y_{GG}]*[V_G]+[Y_{GL}] *\{[Y_{LL}]^{-1}[I_L]-[Y_{LL}]^{-1}*[V_G]*[Y_{GL}]\} \qquad 6$$

Eq 5 &2 can be written as

$$\begin{bmatrix} VL \\ IG \end{bmatrix} = \begin{bmatrix} ZLL & FLG \\ KGL & Y'GG \end{bmatrix} \begin{bmatrix} IL \\ VG \end{bmatrix} \qquad 7$$

Where, $[F_{LG}] = -[Y_{LL}]^{-1} *[Y_{GL}]$; $[K_{LG}] = [Y_{LL}]^{-1} *[Y_{GL}]$;

and $[Y'_{GG}] = \{[Y'_{GG}]-[Y_{LG}]*[Y_{LL}]^{-1} *[Y_{GL}]\}$

The $[F_{LG}]$ matrix elements are complex. Row vectors are used to represent load buses, while column vectors are used to represent generator buses. The imaginary part of the $[F_{LG}]$ matrix is used to determine the appropriate generation schedule $[D_{LG}]$, which is almost negligible.

$$[D_{LG}]= \text{abs}\ \{[F_{LG}]\}$$

The location is calculated as ow

$$[RED] = [M]-[D_{LG}]$$

Where M is the unit matrix of size L*G Desired Generation Schedule

(DGS) is calculated as

No Of busses

[DGS] G th =Σ ([D jg]*[P j])

J=g+1

Where, [P $_j$] is the load at the jth load bus and

[D $_{jg}$] values are taken from the [D $_{LG}$] matrix of a given network.

Distribution Computing Using Big Data with Spark

Big Data is unstructured, structured, or a combination of the three. It is also huge in volume and is gathered quickly [12]. Traditional approaches make it tough to collect, mine, and handle data, while big data makes it easier. Big data is defined as high volume, velocity and variety of information requiring innovative and efficient information processing techniques for better decision-making [13].

V of BIG Data

Using the 4 V model (Volume, Velocity, Variety, and volume), big data can be defined [14].

Volume: It refers to the quantity of data that businesses or individuals produce. It so indicates the maximum storage space.

Velocity: This term refers to the rate at which data is created, modified, processed, and disseminated. As a result, it designates any access to. Data as of a certain moment

Variety: Data could be structured, unstructured, or semi-structured, depending on the situation.

Value: The output of big data analysis is what will leverage the company.

Processing Calculation Using Spark

Spark is an open-source framework for parallel data processing on clustered computers that includes a computing engine and a collection of libraries [15]. It is a powerful big data tool for analyzing various big data problems. It manages workloads for real-time and batches for analytics data processing [16]. Large volumes of data can be processed up to 100 times faster by Spark than by MapReduce [17]. To achieve faster performance, it partitions the data into chunks on the cluster and stores the data in memory. Spark application can be deployed on a perm or cloud environment [18][19]. For this research, we have used the AMAZON EMR cloud environment to run the workload.

Resilient Distributed Datasets (RDD

RDD is the fundamental data type and a key concept in spark for data computing. The input data is stored in RD [20,21]. Analysis of data is done on RDD by applying transformation and actions serve as an interface for immutable data. It also allows you to recompute our analytics in event failure. It is a data structure that aids in recalculating data in the event of failures.

Spark Architecture

Spark application consists of three main components:

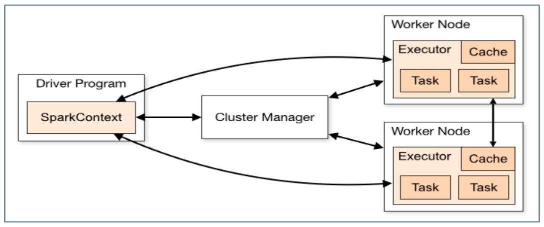

Figure 2: Spark application and its main components

- **The Spark Driver:** The Spark driver is the process that acts like a driver's seat for an application. It manages the tasks of the executors and their state in the Spark cluster as well as how a Spark application is executed. It communicates with the cluster management in order to acquire physical resources and launch executors [22].

- **The Spark Executors:** The Spark driver assigns tasks to executors for processing the data. When a Spark application is submitted for execution, these executors are started and run for the duration of the application. The executor's main functionality is to run the tasks assigned by the driver program and return the success or failure results back to the driver.
- **The Cluster Manager:** The Spark Driver and its Executors do not exist in a vacuum, the cluster manager is required for its orchestration. The cluster manager is in charge of maintaining a collection of computers that will execute spark application(s).

Flow of Program

- The driver application, which runs on the Spark cluster's master node, coordinates with the cluster management to schedule the job execution.
- The driver keeps track of all the resilient distributed databases' partitions' metadata.
- User's program is divided into smaller execution units called tasks by the driver program. Then, the worker processes that conduct particular tasks, or the executors, carry out the tasks.
- Executor performs all the data processing by applying transformation, and actions and returns the results to the Driver.
- The computing output is stored by the executor in memory, in the cache, or on a hard-disk drive.
- The Driver receives results from each executor after the task has been performed.
- In order to increase the system performance, the number of worker nodes must be increased so that the jobs can be divided further into more number of logical portions.

Execution Process

Spark program is written in java and packaged as jar. Input data files are stored in the amazon s3 bucket. Running the spark application is pretty simple in a cloud environment. We moved the packaged jar to the home/Hadoop folder and logged into the master node of the cluster and executed the spark-submit command.

Spark command to run the program: spark-submit power-tracing-analysis.jar

Once the spark program gets started, it will read the data files from S3 location and creates RDD for calculations. The program will apply the necessary transformation and actions to obtain the desired output. These calculations are performed on executor nodes. We can write the output data onto different files based on load or based on the generator. For simplicity, we have collected everything into one file based on load.

Results and Discussions

Big Data processing framework Spark is used in this proposed work to demonstrate the RED approach for tracing real power. We have used the Amazon EMR cluster to run the spark application on 5 node cluster. The cluster is of type m5.2xlarge which has 8 CPU cores and 32gn g ram. The number of worker nodes must be increased in order to increase the number of partitions for data processing, which will boost system efficiency and allows for the analysis of larger datasets. The proposed method is demonstrated on 39 bus systems containing of 29 load buses and 10 generator buses. Spark processing is much faster and can handle large data sets. Optimization mechanism is one of the main reasons for Spark's astronomical performance and effectiveness.

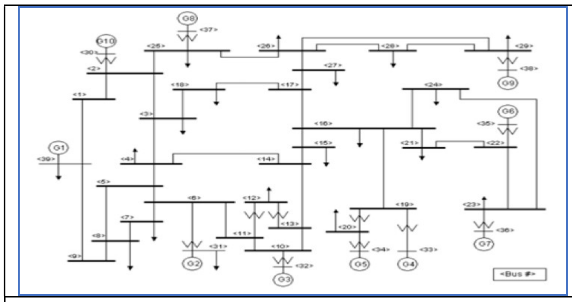

Figure 3: 39 bus system with 10 generations and 29 load

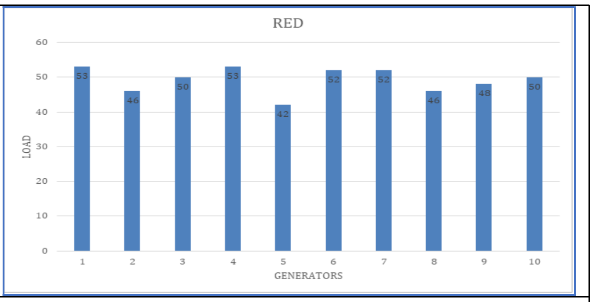

Figure 4: Relative Electrical Distance (RED)

Conclusion

This paper solves the power tracing in. a distribution environment with a RED method using spark. Doing this analysis in a distributed environment helps us to get the results much faster than traditional computation. Spark looks very promising to perform these analyses on even Petabytes of data. We traced the real power from each generator to each load. The results are in line with what is desired, which will increase the power system's effectiveness. This technique will allow load centers and generating centers to communicate effectively and improves transmission services, and congestion management. This will help decrease load failure and shedding of the power system which will also improve the tariff system. This paper discusses batch processing using Apache spark for computation and further research can be extended by collecting some data points regarding the degradation and applying some advanced predictive analysis techniques using spark streaming to predict or minimize the power failure of the network by collecting real-time metrics.

References

[1] Power Tracing in Distribution Network in Deregulated Power Environment (2018 International Conference on the computation of Power, Energy, Information and Communication.

[2] Alturki Y A., and Lo K.L., Real and reactive power loss allocation in Pool-based electricity markets, Electrical Power and Energy systems, 2010, Vol. 32, pp 262-270.

[3] Gorenstin B.G., Campondonico N.M., Costa J. P., and Pereira M. V. F., Power system expansion planning under uncertainty, IEEE Transactions on Power Systems, 1993, Vol. 8, Issue 1, pp.29-36.

[4] Abyankar A. R., Soman S. A., and Khaparde S.A., Optimization Approach to Real Power Tracing: An Application to Transmission Fixed Cost Allocation, IEEE Transactions on Power Systems, Aug. 2006, Vol. 21, Issue 3, pp. 1350-1361.

[5] Sebastian p. Rosado, Khatib Abdel-Rahman, and Nouredine Hadjsaid. Tracing the path of electric power flow, a study for deregulated power systems, IEEE Conference, 2001, pp. 1479-1484.

[6] Gorenstin B.G., Campondonico N.M., Costa J. P., and Pereira M. V. F., Power system expansion planning under uncertainty, IEEE Transactions on Power Systems, 1993, Vol. 8, Issue 1, pp.29-36.

[7] Kirschen D., and Strbac G., Tracing active and reactive power between generators and loads using real and imaginary currents, IEEE Transactions on Power Systems, Nov. 1999, Vol. 14, pp. 1312-1319.

[8] Carmen L.T. Borges, and Djalma M. Falcao, Optimal distributed generation allocation for reliability, losses and voltage improvement, Electrical power Research, 2006, Vol. 28, pp.413-420.

[9] Reactive Electrical Distance Concept for Evaluation of Network Reactive Power and Loss Contributions in a Deregulated System, The Institution of Engineering and Technology (IET), UK, Gener., Trans., and Distb., November 2009, Vol.3, Issue 11, pp.1000-1019.

[10] Apache Spark: A Big Data Processing Engine 2019 2nd IEEE Middle East and North Africa Communications Conference (MENACOMM)

[11] Big Data: A Review, Seref SAGIROGLU and Duygu, SINANC, IEEE,2013, pp 42-47.

[12] Data Management in the Cloud: Limitations and Opportunities Daniel J. Abadi, IEEE Computer Society Technical Committee on Data Engineering, 2009.pp 1-10.

[13] Parallel Data Processing with MapReduce: A Survey Kyong Ha Lee, Yoon-Joon Lee, Hyunsik Choi, Yon Dohn Chung and Bongki Moon, SIGMOD Record, December 2011, Vol. 40, No. 4, pp 11-20.

[14] Data Mining with Big Data, Xindong Wu, Xingquan Zhu, Gong-Qing Wu and Wei Ding.

[15] Apache Spark: a unified engine for big data processing. Communications of the ACMVolume 59Issue 11November 2016 pp 56–65.

[16] Daniel J.S., Salgado R.S., and lrving M.R., Transmission loss allocation through a modified Ybus, Proc. IEEE. Gen. Transm Distib., Mar.2005, Vol. 152, No.2, pp.208-214.

[17] A Review: Mapreduce and Spark for big data analytics by Vaishali Chauhan international Journal.

ISSN: 2583-1380 Vol. 3 | Issue No. 7 | July 2023 Impact Factor: 4.736 (SJIF)

[18] Parallel Data Processing with MapReduce: A Survey Kyong Ha Lee, Yoon-Joon Lee, Hyunsik Choi, Yon Dohn Chung and Bongki Moon, SIGMOD Record, December 2011, Vol. 40, No. 4, pp 11-20.

[19] Big Data Processing with Hadoop-MapReduce in Cloud Systems, Rabi Prasad Padhy, International Journal of Cloud Computing and Services Science (IJ-CLOSER) Vol.2, No.1, February 2013, pp. 16 27.

[20] Daniel J.S., Salgado R.S., and lrving M.R., Transmission loss allocation through a modified Ybus, Proc. IEEE. Gen. Transm Distib., Mar.2005, Vol. 152, No.2, pp.208-214.

[21] Parallel Data Processing with MapReduce: A Survey Kyong Ha Lee, Yoon-Joon Lee, Hyunsik Choi, Yon Dohn Chung and Bongki Moon, SIGMOD Record, December 2011, Vol. 40, No. 4, pp 11-20.