DEVELOPMENT OF POWER SYSTEM OPERATIONAL DATA MANAGEMENT SYSTEM USING CLOUD SERVER

Thesis submitted by

Subhra Jyoti Sarkar

Doctor of Philosophy (Engineering)

Department of Electrical Engineering
Faculty Council of Engineering and Technology
Jadavpur University, Kolkata– 700032, India

2019

Dedicated to

My Parents

My Grandmother

My Wife

Index No.: 289/16/E

1. **Title of the Thesis: DEVELOPMENT OF POWER SYSTEM OPERATIONAL DATA MANAGEMENT SYSTEM USING CLOUD SERVER**

2. **Name, Designation and Institution of the Supervisors:**
   (a) Dr. Palash Kumar Kundu
   Associate Professor
   Department of Electrical Engineering
   Jadavpur University, Kolkata– 700032, India
   (b) Dr. Gautam Sarkar
   Associate Professor
   Department of Electrical Engineering
   Jadavpur University, Kolkata– 700032, India

3. **List of Publications:**
   - Subhra J. Sarkar, Palash K. Kundu, Gautam Sarkar, "Development of Embedded Power System Data Management System", **Modelling, Measurement and Control A**, Vol. 90, No. 3, September 2017, pp. 279-295.
   - Subhra J. Sarkar, Palash K. Kundu, Gautam Sarkar, "Development of DBEA Compressed Data Transfer System over Power Line", **Modelling, Measurement and Control A**, Vol. 90, No. 3, September 2017, pp. 262-278.
   - Subhra J. Sarkar, Palash K. Kundu, Gautam Sarkar, "Development of Lossless Compression Algorithms for Power System Operational Data", **IET Generation, Transmission and Distribution**, Vol. 12, Iss. 17, September 2018, pp. 4045-4052.
   - Subhra J. Sarkar, Palash K. Kundu, Gautam Sarkar, "Development of Cloud based Power System Operational Data Management System", **IET Generation, Transmission and Distribution**, Vol. 13, Iss. 5, March 2019, pp. 644-651.

4. **List of Patents: NIL**

5. **List of Presentations in National/ International Conferences**
   - Subhra J. Sarkar, Palash K. Kundu, "A Novel DCSK Communication Scheme for PLCC based DAS", Proc. of International Conference on Electrical and Computer Engineering (ICECE), 2014, 20^{th} – 23^{rd} December 2014, Dhaka, Bangladesh, pp. 50 – 53.

- Subhra J. Sarkar, Palash K. Kundu, “A Proposed Method of Load Scheduling and Generation Control using GSM and PLCC Technology”, Proc. of Michael Faraday IET International Summit (MFIIS 2015), 12th – 13th September, 2015, Kolkata, India, pp. 273- 277.
- Subhra J. Sarkar, Palash K. Kundu, Ipsita Mondal, “A Novel Data Compression Approach to Reduce the String Size for PLCC based DAS”, Proc. of 12th IEEE India International Conference on Electronics, Energy, Environment, Communication, Computer and Control (INDICON-2015), 17th -19th December, 2015, New Delhi, pp. 1-5.
- Subhra J. Sarkar, Palash K. Kundu, Ipsita Mondal, “Modified DCSK Communication Scheme for PLCC based DAS with Improved Performance”, Proc. of International Conference on Control, Instrumentation, Energy and Communication (CIEC), 2016, 28th – 30th January, 2016, Kolkata, India, pp. 456- 460.
- Subhra J. Sarkar, Nabendu Kr. Sarkar, Aritra Banerjee, “A Novel Huffman Coding based Approach to Reduce the Size of Large Data Array”, Proc. of IEEE International Conference on Circuit, Power and Computing Technologies (ICCPCT- 2016), 18th – 19th March 2016, Nagercoil, Tamilnadu, India, pp. 1-5.
- Subhra J. Sarkar, Palash K. Kundu, Gautam Sarkar, “Comparison of the performance of DCSK for PLCC with and without data compression”, Proc. of International Conference on Intelligent Control Power and Instrumentation (ICICPI), 2016, Kolkata, India, 21st – 23rd October, 2016, pp. 246-250.
- Subhra J. Sarkar, Palash K. Kundu, Gautam Sarkar, “DBEA: A Novel Approach of Repetitive Data Array Compression for Power System Applications”, Proc. of 2nd International Conference for Convergence in Technology (I2CT, 2017), 7th – 9th April, Pune, India, pp. 1-4.
- Subhra J. Sarkar, Kalyan Kar, Indranil Das, “Basic Arithmetic Coding based Approach for Compressing Generation Scheduling Data Array”, Proc. of 2017 IEEE Calcutta Conference (CALCON 2017), 2nd - 3rd December, 2017, Kolkata, India, pp. 21-25.
- Subhra J. Sarkar, Palash K. Kundu, Gautam Sarkar, “Performance Analysis of Resumable Load Data Compression Algorithm (RLDA) for Power System Operational Data”, Proceedings of 2017 IEEE Calcutta Conference (CALCON 2017), 2nd - 3rd December, 2017, Kolkata, India, pp. 16-20.
- Subhra J. Sarkar, Palash K. Kundu, Gautam Sarkar, “Development of Compressed Encrypted Data Management System for Power System

Applications", Proc. of 2017 IEEE Calcutta Conference (CALCON 2017), 2nd - 3rd December, 2017, Kolkata, India, pp. 11-15.

- Subhra J. Sarkar, Palash K. Kundu, Gautam Sarkar, "Realization of a Cloud Server based Power System Operational Data Management System" Proc. of International Conference on Electrical and Computer Engineering (ICECE), 2018, 20th – 22nd December 2018, Dhaka, Bangladesh, pp. 105 – 108.
- Subhra J. Sarkar, Palash K. Kundu, Gautam Sarkar, "Development of Combined Differential Binary Encoding Algorithm for Power System Operational Data Compression" Proc. of International Conference on Electrical and Computer Engineering (ICECE), 2018, 20th – 22nd December 2018, Dhaka, Bangladesh, pp. 109 – 112.

CERTIFICATE

This is to certify that the thesis entitled "Development of Power System Operational Data Management System using Cloud Server" submitted by Sri. Subhra Jyoti Sarkar, who got his name registered on 29th June, 2016, for the award of Ph.D. (Engg.) degree of Jadavpur University is absolutely based upon his own work under the supervision of Dr. Palash Kumar Kundu and Dr. Gautam Sarkar and that neither his thesis nor any part of the thesis has been submitted for any degree/diploma or any other academic award anywhere before.

Dr. Palash Kumar Kundu
Associate Professor
Electrical Engineering Department
Jadavpur University
Kolkata– 700032, India

Dr. Gautam Sarkar
Associate Professor
Electrical Engineering Department
Jadavpur University
Kolkata– 700032, India

ACKNOWLEDGEMENT

During the journey in obtaining my PhD, several domains were explored which might not be possible without the support and encouragement of numerous persons. This thesis is the end of my journey in obtaining my PhD which has been kept on track and been seen through to completion with the kind support of my family, friends, teachers, colleagues and well wishers. It would not have been possible to write this doctoral thesis without the help and support of these kind people around me. It is a pleasant task to express my thanks to all those who contributed in many ways to the success of this study and made it a memorable experience for me. At the end of this thesis, I would like to thank all those people who made this thesis possible.

First of all I would like to thank my supervisors Dr. Palash Kumar Kundu and Dr. Gautam Sarkar for their advice, guidance and support. Under their guidance, I successfully overcame many difficulties and learned a lot. I am extremely indebted to them for providing the necessary infrastructure and resources to accomplish my research work. Their advice, reviews and suggestions around my work were extremely valuable during the period of research.

I must also acknowledge the technical and administrative support provided by Jadavpur University during the research. I am obliged to Prof. Subrata Paul, HOD and Professor, Department of Electrical Engineering, Jadavpur University for his valuable guidance during the work. I am also thankful to Prof. Sugata Munshi, and Prof. Amitava Chatterjee for their constant encouragement during the research. I must extend my sincere thanks to Prof. (Dr.) Ankur Ganguly, Principal, Techno India Batanagar to provide the necessary support to pursue my research. I also thank Dr. Ashok Kumar Naskar, HOD, Dr. Debabrata Roy and Mr. Kalyan Kar, Department of Electrical Engineering Department, Techno India Batanagar for their support and assistance during my research work. I must also thank Mr. Indranil Dasgupta, HOD, Computer Science and Engineering Department, Techno India Batanagar for his kind help and suggestions.

I must acknowledge Dibendu Bhattacharya, West Bengal State Load Despatch Centre (WBSLDC) for providing the necessary system practical monitoring data for testing the effectiveness of proposed algorithms. I am also thankful to Mr. Bhaskar Sengupta, Chief Engineer (Retired), West Bengal State Electricity Board (WBSEB) for his help and guidance during the course of work. I must also extend my sincere thanks to Mr. Dipak Kumar Pal, Retired Manager (Construction), Calcutta Electricity Supply Corporation (CESC) for his warm suggestions and constant encouragement during the research work.

Above all, I would like to thank my wife Ipsita for her personal support and great patience at all times. My parents and my brother have given me their unequivocal support throughout, as always, for which my mere expression of thanks likewise does not suffice.

For any errors or inadequacies that may remain in this work, of course, the responsibility is entirely my own.

Subhra Jyoti Sarkar

Kolkata

March, 2019

CONTENTS

ABSTRACT

Present dissertation on the development of power system operational data management system using cloud server has been taken up in the backdrop of major changes occurring in the power systems. Remote Terminal Units (RTUs) and Intelligent Electronic Devices (IEDs) are gradually replaced by Phasor Measurement Units (PMUs) for making the system smarter. With this technological improvement, there is a significant rise in the sampling frequency of data acquisition, which results in the formation of the enormous volume of information. Storage of this bulk volume of information is extremely important for energy accounting and understanding the system behaviour during extreme conditions. These data will be useful for validating mathematical models developed for representing a power system under different conditions. The present trend of storing such data periodically in separate storage devices is a challenging task for any data centre. Data compression can be a good solution to reduce the volume of information and thereby reducing overhead expenses. From communication prospect, it is obvious that for increased volume of information, higher energy is required to transmit the information through any communication channel. This implies that the transfer of compressed data will result in reduced energy requirement. The majority of the available works in this domain are based on entropy based algorithms where probability distribution table is required for decoding the information. This will not only increase computational burden, but also limits the implementation of these algorithms for communication between two systems (PC or microcontrollers).

Power system operational data can be either scheduling related information transferring between Load Despatch Centres (LDCs), Generating Stations (GSs) and Distribution Companies (DCs) and parameter monitoring information transferring between control rooms of LDCs and field devices. In terms of data volume, parameter monitoring information is of more significance than scheduling related information. Analysis of practical data indicates that this information is in the form of large data array. During the initial phase of the work, Basic Arithmetic and Huffman Coding based lossless compression algorithms were developed for compressing generation scheduling information. The work is extended to determine the effectiveness of these entropy based compression algorithms for practical monitoring data. The CR obtained by the algorithms is much lower than that achieved by the works available in some literatures. In order to enable data communication, equal probability distribution is considered at encoding and decoding end. It is obvious that the achievable Compression Ratio (CR) can be increased by using adaptive entropy based compression algorithms. But this in turn requires the exact probability distribution table at decoding end and thereby limiting its application for communication purpose.

Analysis of practical data indicates the slow varying and the repetitive nature of data array. Resumable Load Data compression Algorithm (RLDA) was a lossless compression algorithm available in the literature for compressing load profile data of smart meters and a high CR was achieved. In RLDA, differential coding is followed by variable length coding and Binary Arithmetic Coding for compressing a large data array to a binary string. While analysing the performance of RLDA with the practical power system operational data, it was found that it can give high CR for those data as well. But for large, non-repetitive data array, there is a significant rise in the computation and thereby limiting its implementation for managing power system operational data. As decoding of binary string requires the probability distribution information, RLDA is not suited for communication purpose. Due to these limitations of RLDA, the objective was to develop a low computational lossless compression algorithm which does not require probability distribution information at the decoding end.

Based on the practical data analysis, Differential Binary Encoding Algorithm (DBEA) was developed for compressing a large, slow varying data array to a character string. As DBEA is not based on any entropy based compression algorithm, there is a significant reduction in computation. For highly repetitive data array, the CR obtained by DBEA is extremely high. As the majority of the available data exhibits the high repetition feature, high CR is achieved with them. Under transient condition of the grid, data array might have a sudden change between the consecutive elements where DBEA will fail. To overcome this limitation of DBEA, the work is extended to develop E-DBEA. Unlike DBEA, E-DBEA can handle data arrays over a much wider range and can compress all the available information. But due to the algorithmic feature of E-DBEA, the CR obtained by E-DBEA is approximately half of that achieved by DBEA for majority of available data. As DBEA and E-DBEA can be useful both for data storage and transfer, these algorithms were tested offline and in real time. The real time test bench of DBEA and E-DBEA comprises of two computers connected through cross connected RS232 cable. A sample array is compressed to a character string and is transmitted to the decoding end where the actual data array is decoded from this character string and the result is printed on the screen.

Power line carrier communication (PLCC) is still in operation for transferring data through power lines. With the rapid increase in electronic devices, there is a significant increase in power line noise. Due to the presence of noise, conventional modulation techniques are not suited for the power line environment. Differential Code Shift Keying (DCSK) modulation scheme, patented by Yitran Technologies was a good choice for PLCC and many PLCC modems uses this modulation scheme for transferring data through power lines. Initially this DCSK modulation scheme was developed in MATLAB where a superchirp corresponds to a character string is obtained at the encoding end. The superchirp is then decoded to the respective character string at the decoding end. In the subsequent works, Basic Arithmetic and Huffman Coding were introduced to reduce the length of superchirp. As DBEA can effectively compress a data array to a character string, a DBEA compressed data transfer system for the power line environment was developed. At the encoding end, a data array associated with power system is compressed by DBEA before obtaining the superchirp corresponds to the data array. The compressed character string is obtained at the decoding end from which actual data array is decoded. There is a significant reduction in data volume in this method which will result in reduced energy requirement for transferring the information.

Though the simplicity of DBEA and E-DBEA enables its implementation at low level microcontrollers, data security over the communication channel is really a matter of concern. Rivest-Shamir-Adleman (RSA) algorithm is one of the oldest public key cryptographic algorithms used widely for secured data transmission where an integer is encrypted to another integer according to public (or private key) at encryption end. This encrypted integer is decrypted by using private (or public key) and the actual integer is obtained at the decrypting end. As the output of DBEA (or E-DBEA) is a character string which is nothing but a combination of ASCII values (integers), it is possible to encrypt the ASCII values by using RSA algorithm. Based on this idea, RSA encrypted DBEA algorithm was developed in the preliminary stage where the output character string is an encrypted string from which actual data array can't be extracted without the respective key. But as the range of the data array handled by DBEA is much lower, the algorithm will have its own limitations. Though E-DBEA had much wider data array handling, the reduced CR demands necessary modifications so that both data security and high extent of compression can be ensured.

Combined DBEA (C-DBEA) is a better solution of compressing power system operational data as it selects DBEA or E-DBEA according to data array. An additional character is included at the beginning of compressed information to convey the information regarding the type of compression algorithm being employed. Inclusion of this additional character will result in reduced CR in comparison to that obtained with DBEA or E-DBEA. But as it selects DBEA or E-DBEA according to the data array, it is capable to extract high CR along with wider data array handling capacity. As serial communication is an obsolete communication technique and internet based communication is getting importance in the modern world, RS232 based test bench must be upgraded. As cloud computing is emerging rapidly in India, real time testing of C-DBEA was performed by using Dropbox based cloud environment. Analysis of practical power system operational data indicates the possibility of multiple versions of data. Management of such multiple versions of the same information is not possible with none of the algorithms discussed so far. Some fields were identified from parameter monitoring and generation scheduling information which must be included with the compressed information for selecting a particular version of data from a large pool of data.

For management of practical power system operational data, Smart DBEA Encryption (S-DBEAE) was developed from C-DBEA. In S-DBEAE, the compressed character string obtained with DBEA or E-DBEA is encrypted with RSA algorithm and the necessary fields were included in four identity characters. After inclusion of these fields it is possible to store multiple data in a single field by appending the character string obtained by S-DBEAE in the existing file. Due to the inclusion of four identity characters, CR being obtained by S-DBEAE is much lower than that obtained with DBEA or E-DBEA. But this feature will enable compressed, encrypted data management facility which is not available in DBEA or E-DBEA. Similar to C-DBEA, cloud based real testing setup was realised for S-DBEAE. Development and testing of algorithms for power system operational data will be incomplete without system realization. A testing setup comprising of 4 PCs were realised in the laboratory for resembling a miniature version of the power system. These PCs are connected through internet, and the file containing compressed, encrypted information will be stored in the cloud. As the work focuses on the management of generation scheduling data, these PCs will resemble 1 LDC, 1 DC and 2 GS. For managing parameter monitoring information, these PCs will resemble 1 data centre and 3 field devices.

According to Indian Electricity Grid Code (IEGC), scheduling of different GS during 96 time blocks is done according to its next day availability and system requirement and thus this information must be provided by GS and DC beforehand. In the developed prototype, the respective GS will compress the next day availability information by S-DBEAE and update the respective file at some time t_1. Simultaneously, load demand information is compressed by S-DBEAE in a program running in DC and the respective file is updated. At time t_2, all this information is extracted by LDC by using S-DBEAE decoding algorithm. The solution of Economic Load Despatch (ELD) problem for this 2 GS system can be done by the Lambda Iteration method, Iterative Weight Particle Swarm Optimization (IW-PSO) or Artificial Neural Network (ANN). This information is then compressed by S-DBEAE and is updated in the respective file. Next day load allocation information of DC for 96 time blocks is also compressed by S-DBEAE and is updated in the respective file. At time t_3, next day generation allocation information is decoded by S-DBEAE and is displayed in GS end. At that very same time, load allocation information is decoded and displayed in DC end. The proposed system has several technological advantages over the existing system. In the existing system, data storage is a matter of concern which was considerably taken care in the proposed system. The proposed system will also ensure data security, which is unavailable in the existing system. Multiple .xls files corresponding to different versions and dates containing information of all available GS and DC is maintained in uncompressed form in the existing system. Due to the inclusion of data management function, extraction of a particular file (in .xls form) from a single text file corresponds to a particular parameter or GS (or DC) is relatively easier in the proposed system.

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

Abbreviations	Description
PMU	Phasor Measuring Unit
TCP	Transmission Control Protocol
IP	Internet Protocol
UDP	User Datagram Protocol
DTP	Data Transfer Protocol
SCADA	Supervisory Control and Data Acquisition
GS	Generating Station
DC	Distribution Company
LDC	Load Despatch Centre
SLDC	State Load Despatch Centre
RLDC	Regional Load Despatch Centre
NLDC	National Load Despatch Centre
PGCIL	Power Grid Corporation of India Limited
POSOCO	Power System Operation Corporation Ltd
WBSEDCL	West Bengal State Electricity Distribution Company Limited
CESC	Calcutta Electricity Distribution Company
DVC	Damoder Valley Corporation

IEGC	Indian Electricity Grid Code
UI	Unscheduled Interchange
NTPC	National Thermal Power Corporation
ABT	Availability Based Tariff
RTU	Remote Terminal Unit
PLC	Programmable Logic Controller
IED	Intelligent Electronic Device
WBSLDC	West Bengal State Load Despatch Centre
ERLDC	Eastern Regional Load Despatch Centre
FSTPP	Farakka Super Thermal Power Plant
KHSTTP	Kahalgaon Super Thermal Power Plant
WBPDCL	West Bengal Power Development Company Limited
BTPS	Bandel Thermal Power Station
STPS	Santaldih Thermal Power Station
KTPS	Kolaghat Thermal Power Station
BkTPP	Bakreswar Thermal Power Plant
SgTPP	Sagardighi Thermal Power Plant
TLDP	Teesta Lower Dam Project
HEL	Haldia Energy Limited
IEC	International Electrotechnical Commission
PLCC	Power Line Carrier Communication
CR	Compression Ratio

RLDA	Resumable Load Data compression Algorithm
PSO	Particle Swarm Optimization
IW-PSO	Inertia weight PSO
ANN	Artificial Neural Network
BPNN	Back Propagation Neural Network
UC	Unit Commitment
ELD	Economic Load Despatch
DP	Dynamic Programming
DBEA	Differential Binary Encoding Algorithm
E-DBEA	Extended DBEA
M-DBEA	Modified DBEA
DCSK	Differential Code Shift Keying
UST	Unit Symbol Time
WAMS	Wide-Area Monitoring Systems
PDC	Phasor Data Concentrator
IoT	Internet of Things
GoI	Government of India
RSA	Rivest-Shamir-Adleman
C-DBEAE	Combined DBEA Encrypted
LAN	Local Area Network
WAN	Wide Area Network
S-DBEAE	Smart DBEA Encrypted
PC	Personal Computer
DAS	Data Acquisition System

Chapter 1

INTRODUCTION

It is expected that there will be increased power demand in the upcoming decades, which not only put an additional burden on existing power infrastructure but can also degrade the power quality. The existing power grid is unreliable due to the lack of efficient monitoring, fault diagnostic, and automation techniques. Integration of distributed energy sources to the existing grid is also extremely challenging. These problems can be tackled effectively by Smart Grids which is nothing but an intelligent power infrastructure capable of making power systems more secure, reliable, efficient, flexible and sustainable [1, 2]. Data management and communication are extremely crucial for operation of smart grid and thus various works are going on all around the globe for managing the information associated with grid [3-7]. The integration of Wireless Sensor Networks (WSNs), actuators, smart meters, and other components of the power grid with together with information and communication technology (ICT), is referred to as the Internet of Energy (IoE). IoE uses the bidirectional flow of energy and information within the smart grid to gain deep insights on power usage and predicts future actions to increase energy efficiency and low overall cost [1]. IoT can deliver smarter grid so as to have more information and connection throughout the infrastructure and to homes [8-10]. In comparison to the conventional power grid, smart grid possesses the some additional features as given below [11].

i. Integration of renewable energy resources (PV, wind turbine, etc.) at distribution network.

ii. Supervisory control and real time status monitoring on power network

iii. Self- monitoring
iv. Adaptive response to any fault
v. Self healing, etc.

As energy demand is dynamic in nature, existing grid has to balance electricity supply and demand between consumers and utility providers. This can be achieved by incorporating Energy Management Systems (EMS) including Building Energy Management Systems (BEMS), Demand Side Management (DSM) and Home Energy Management (HEM). A smart grid allows various renewable energy sources to have an efficient management of supply and demand. The special characteristic of a smart grid is its heterogeneous architecture, which includes Demand Response (DR), distributed generation, resource scheduling, and real-time pricing model [12, 13]. The problem of distributed energy management for both generation and demand sides in a smart grid is investigated in various works [14, 15]. In Smart Grid, raw information on network healthiness or performance is obtained from smart information subsystem like smart meters, sensors and phasor measurement units (PMU). Smart Meters are deployed in many countries from the beginning of the 21st century that have inbuilt features like tamper proof, fault detection, etc. The concept of Automated Meter reading (AMR) is replaced by Advanced Metering Infrastructure (AMI) in the recent times. Two way communication for information, communication, etc. and data system available in AMI had evolved from one way communication to collect meter data in AMR. Smart meter data analytics is one of the key factors behind the success of smart meters, which deals with data acquisition, transmission, processing, and interpretation that bring benefits to all stakeholders [11, 16-18].

PMU, as the name suggests is a device capable of measuring the synchronized voltage and current phasor in a power system. Synchronism among PMUs is achieved by same-time sampling of voltage and current waveforms using a common synchronizing signal from GPS. While approaching towards smart grid, the importance of PMU in power system monitoring and control is inevitable [19]. The resolution of PMU is up to 60 samples per second and the provision of wide- area monitoring is included. The voltage and current output of instrument transformers are fed to the analog input of PMU. An anti-aliasing filter is used to filter out high frequency signals (frequencies above the Nyquist rate of A/D converter) from the input waveform. Sampling of waveforms is done according to GPS pulse and necessary DFT calculation is performed by microprocessor present in it. Finally, time stamped phasor information is uploaded to a collection device (Data concentrator) through Ethernet (TCP/ UDP) [19-22]. Supervisory Control and Data Acquisition (SCADA) is extremely important for the existing power system and numerous Remote Terminal Units (RTUs) are installed in remote field data interface. Similar to PLCs, the RTUs operate in remote locations for equipment monitoring or control. In contrast to PLCs, RTUs includes additional hardware features like power supply, master control element, process and communication elements, peripheral elements diagnostic displays and support for battery backup. These features make them more powerful than PLCs. An RTU comprises of a CPU, volatile and non-volatile memory for processing / storing programs and data. Serial ports or an on-board modem with I/O interfaces is provided for their communication with other devices. This CPU can be any microprocessor or microcontroller to which necessary sensors are connected. These data values acquired by the system are responsible for monitoring the condition system [23-27].

Data mining is the application of specific algorithms for extracting patterns from any available data set. It consists of applying data analysis and discovery algorithms that, under acceptable computational efficiency limitations, produce a particular enumeration of patterns (or models) over the provided data. With increased system monitoring equipments, it becomes necessary to develop the applications which can make appropriate use of this data to perform the task of power system analysis. It is obvious that monitoring data acquired during fault conditions will be different from that during normal (or stable) condition of the grid. Based on the acquired data, it is possible to determine the nature and location of the fault. Data mining can also be employed to identify real data and to generate load profile at distribution end. Electricity load forecasting is also extremely important for maintaining system stability. In practice, experienced engineers predict the forecasted information by observing data patterns. As data pattern can be obtained by applying data mining and knowledge discovery, load forecasting is possible by employing these techniques [28-34].

1.1. Power System Operation

In order to earn some profit on the capital investments, economic operation of power system is extremely important. This can be achieved by producing power at minimum cost which is categorized as economic dispatch. If minimum power loss is achieved during delivery of generating power to load centres, maximum profit can be obtained. The prime objective in economic dispatch is to reduce cost per unit active power generation of different generating station (GS) while satisfying load conditions and transmission losses. A generalized nonlinear programming (NLP) formulation of the economic dispatch problem was introduced in 1962 by Carpentier which includes voltage and other constraints. This problem was termed as Optimal Power Flow (OPF) and plays an important role in power system operation and planning. Variables of OPF problem comprise of a set of dependent variables (Node voltage, phase angle, MVAR output of generators controlling node voltages) and control variables (real and reactive power output of generators, voltage settings of voltage control nodes, tap positions of transformer etc.). There are some constraints which are included in OPF. These constraints may be equality or inequality constraints which are enlisted below [35-38].

i. Power flow equation
ii. Maximum and minimum possible output (active and reactive power) of any generator
iii. Capacity constraints on shunt capacitors and reactors
iv. Upper and lower limit of tap positions of transformers/ phase shifter
v. Branch transfer capacity limits
vi. Node voltage limits

Initially power system operation in India was governed by Power Grid Corporation of India Limited (PGCIL). Later on, a new organization Power System Operation Corporation Ltd (POSOCO), a subsidiary of the PGCIL was formed in March 2010 to handle the power management functions and comprises of a series of load despatch centre (LDC) at national, regional and state level. The national load despatch centre (NLDC), situated in New Delhi governs the operation five regional load despatch centres (RLDC) for north, west, south, east and northeast regions. RLDC is followed by state load despatch centres (SLDC), responsible for maintaining stable operation within the state with the help of one or multiple sub-LDCs operating under it. In the context of West Bengal, there are several sub-LDCs like LDCs of WBSEDCL (primary beneficiary), CESC, DVC, etc. operating in conjunction with WBSLDC to maintain system stability. The hierarchy of Load Despatch Centre is given in Figure 1.1 [39-42].

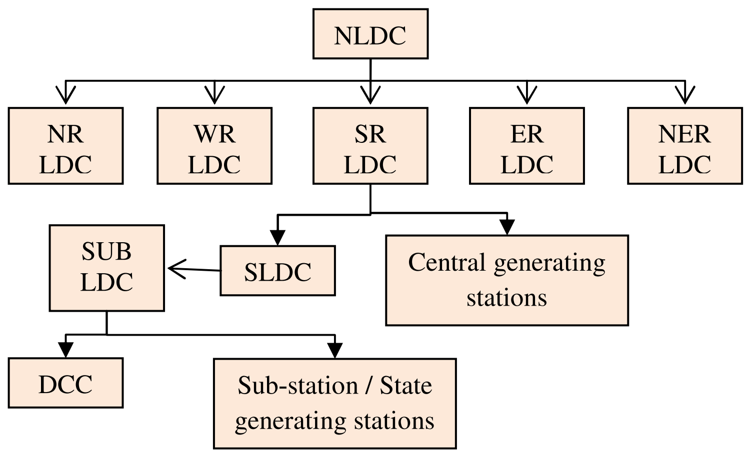


Fig. 1.1. LDC Hierarchy in India for power system operation

Indian Electricity Grid Code (IEGC) brings together a single set of technical and commercial rules, encompassing all the Utilities connected to/or using Inter State transmission system (ISTS) and provides the following [39]:

i. Documentation of the principles and procedures defining the relationship between the various Users of the ISTS, National, Regional and State Load Despatch Centre (NLDC, RLDC and SLDC).
ii. Facilitation of the optimal operation of the grid, facilitation of coordinated and optimal grid maintenance planning and facilitation of development and planning of economic and reliable National/ Regional grid
iii. Facilitation for development of power markets by defining a common basis of operation of the ISTS, applicable to all users of the ISTS.
iv. Facilitation of the development of renewable energy sources by specifying the technical and commercial aspects for integration of these resources into the grid.

LDC is responsible for coordinating the following [39]:

i. Scheduling of a generating station, within its control area

ii. Real-time monitoring of the station's operation and checking for any gaming in its availability declaration
iii. Revision of availability declaration and injection schedule
iv. Switching instructions, metering and energy accounting, issuance of UI accounts within the control area
v. Collections/disbursement of UI payments, outage planning, etc.

According to IEGC [39], scheduling guidelines are as given below:

i. For GS supplying power to interstate transmission networks only, scheduling is coordinated by RLDC
ii. For GS supplying power to state transmission networks, schedule coordination in done by SLDC
iii. For GS supplying power to both to interstate transmission networks and state transmission networks, scheduling is coordinated either by RLDC or SLDC depending on the amount of sharing of generated power.
iv. For same sharing of power, scheduling coordination is done by RLDC

Scheduling and dispatch process will be as given below [39]:

i. Listing of ISGS (Inter State Generating Station), its capacity and allocated/contracted shares or different beneficiaries must be done in RLDC or SLDC website.
ii. State entitlement for thermal GS will be in MW. But for hydro GS, it will be in MWh.
iii. By 9 AM everyday, ISGS will declare its MW or MWh capabilities for the next day (00:00-24:00 hours).
iv. This information is circulated to all the beneficiaries by 10 AM.
v. SLDC will declare their drawal schedule for each ISGS by 3 PM to RLDC after taking care of its foreseen load pattern, self generation capability and bilateral exchanges.
vi. By 6 PM each day, RLDC will convey next day despatch schedule to individual ISGS (in MW/time block)

vii. RLDC must be informed within 10 PM by ISGS/ SLDC in case of any modification/change of drawal schedule/ foreseen capabilities.

viii. Based on the prepared despatch schedule obtained by individual GS/beneficiaries from RLDC/SLDC, generation/ drawal is controlled accordingly so as to avoid any Unscheduled Interchange (UI) charge.

1.2. Tariff in Power System Distribution in India

Initially, when NTPC had started its production, the single part tariff was introduced for the prising of thermal power plant. The tariff was such calculated so that it can cover the fixed charge and variable charge at a particular generation level (say at 65% of the plant load factor). But this tariff scheme suffers from the disadvantage of losing revenue of generating station (GS), even beyond the fixed cost when the production is lower than the threshold value. But when the production exceeds the threshold value, the plant is benefitted with surplus revenue. This scheme of increasing the generation during both peak and off-peak hours is suitable only when there is an extreme power shortage. The variation of annual revenue with the % plant load factor in the single part tariff is illustrated in the Figure 1.2 [43].

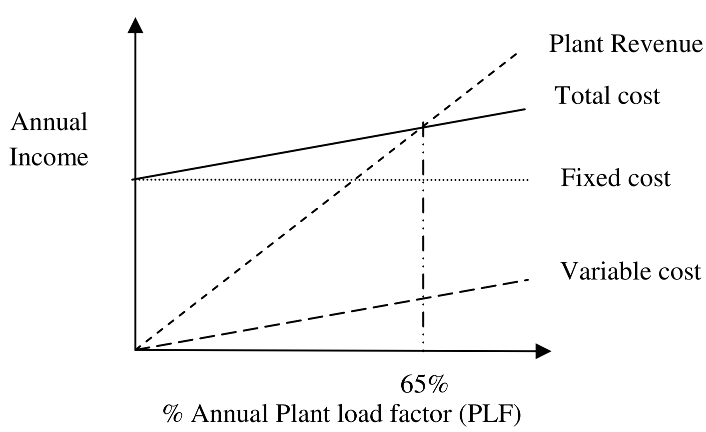


Fig.1.2. Single Part Tariff

As state owned thermal GS might not have higher demand in comparison to central generating stations (NTPC) resulting lower value of PLF. Based on the recommendations of K. P. Rao committee, two-part tariff comprising of fixed and variable charges was adopted by the Government of India in 1992 for NTPC GS. In this scheme the central GS must be paid with the fixed cost even if a distribution company hadn't harness a single unit of electricity. Apart from that, two-part tariff do not consider the frequency variation of the grid with the generation variation. As two-part tariff is not applicable to non-central generating units, over-drawal during peak hours and under-drawal during off-peak hours had become unabated for some state electricity boards. All these limitations results in the formation of a modified tariff scheme Availability Based Tariff (ABT), which was developed by a task force committee formed in the late 90s. This tariff was finally implemented in 2002-2003 and is under operation till now. The main objective of the tariff is to improve the efficiency of generating plants as the fixed cost will be provided only if the plant availability is found. ABT can be considered as a 3-part tariff involving fixed charge, energy charge and unscheduled interchange (UI) charge. The capacity or fixed chart of a GS is computed after considering the following points: [42, 43]

i. Interest on the loan and working capital
ii. Depreciation of the equipments
iii. Return on equity
iv. Operational and maintenance expenses including insurance

Energy charge includes the fuel cost of the energy fed to the buses which is lower than that being generated as the remaining energy is used to run plant auxiliaries. Both generating stations and distribution companies prepare their schedule in advance for 96 time blocks and submit the information to the corresponding LDC. If due to some faulty calculation of load forecasting or tripping of one or multiple alternators, there might have some over-drawal or under-drawal thereby violating the schedule. Unscheduled interchange in a particular time block can be defined as the difference between total actual generation and total scheduled generation for a GS or a seller and the difference between total actual drawal and total scheduled drawal for a beneficiary or a buyer. The penalty or reward for this unscheduled interchange is termed as unscheduled interchange charge and is based on the average frequency of the time block. Penalty or reward on the beneficiaries and generating stations are decided according different conditions as given in Table 1.1 [42, 43].

Table 1.1

Nature of UI charges under different conditions

Conditions	Frequency over rated value	Frequency less than rated value
Over drawal by beneficiaries	Reward for UI	Penalty for UI
Under drawal by beneficiaries	Penalty for UI	Reward for UI
Generation beyond schedule	Penalty for UI	Reward for UI
Generation less than schedule	Reward for UI	Penalty for UI

1.3. Power system operational data

From the discussions of Section 1.2., it is clear that LDCs have to deal with enormous volumes of data, which can be either scheduling related data or system monitoring data. As system demand does not varies abruptly, a day is sub-divided into 96 time blocks of 15 minutes each within which load variation is low and can be handled by GS. The structure of the scheduling charts for GS and distribution company(s) is different. For distribution company(s), the drawal schedule comprises of the amount of power export and import from (or to) other distribution company or power traders, allocated schedule by the LDC and actual drawal by the company(s) for all 96 time blocks. On the other hand, the generation schedule comprises of notional availability (possible generation when favourable condition is met), actual availability (generation possible in the present circumstances) and scheduled generation for all 96 time blocks. It is here to be noted that, fixed cost of any generating station will be based on its notional availability. So, LDCs also has to check for any gaming in notional availability declaration of the plants. The preparation of the generation schedule of any GS for a particular day is done after considering the following constraints [39, 44].

i. Ramp-up or ramp-down rate of the alternator in (MW/hr)
ii. Minimum generation of the alternator for economic operation
iii. Ratio of minimum and maximum generation level

SCADA systems are computer based control systems used for monitoring and control of physical systems. Modern day SCADA systems consist of RTUs, field device monitors, and control process equipment and systems from multiple locations and exchange data from various distributed control systems along the local and wide area networks. The acquired data in electrical systems include critical operational data, performance metering and events and alarms. In SCADA architecture, the role of Remote terminal units (RTUs) are to connect sensors in the system, transmit the acquired data to the supervisory system and receiving instructions from supervisory system. On the other hand, Programmable logic controllers (PLCs) have sensors connected to it, but do not have any inbuilt telemetry hardware. So, PLCs can replace RTUs due to its economy, versatility, flexibility and configurability. The communication system between field devices and control centre might be wired or wireless. The control centre comprises of Human-Machine Interface (HMI), Data Historian, control/data acquisition server, communication router, etc. HMI presents the processed data to a human operator for monitoring and interaction. Data Historian has various data, events and alarms in a database which is HMI accessible. Both HMI and Historian are the clients for data acquisition server, which allows them to access any data from field devices using suitable protocol [45-49]. Existing SCADA in WBSLDC can acquire data typically at every 4-5 seconds and the information are updated in HMI present in the control room. This implies that each system monitoring data array contains > 700 elements per hour and thus constitutes an enormous volume of information. Of the numerous sources of data associated with power system operation, some are enlisted in Table 1.2. From the discussions, it is clear that the existing system has the following limitations:

i. In the available works proposed for compressing power system operational data, considerable computation is involved which limits their applications.
ii. Due to numerous sources of data, management of power system operational data is really a challenging task.
iii. The existing method of generation scheduling data transfer between GS, LDC and DC requires human effort which can be automated by introducing some additional features.

Table 1.2

Sources of data associated with power system operation

Nature of data	No. of elements in array	Regularity of data transfer	Category
Next day availability information	96	Fewer times a day	Data associated with generation scheduling
Generation allocation information			
Load forecasting information			
Bus voltage variation information	>700 per hour	Continuous data transfer	Data associated with parameter monitoring
System frequency variation information			
Line current variation information			
Power flow through line variation information			
Plant generation variation information			
Energy exchange information			

The survey of the unit commitment data of six different GS operating under WBSLDC [50] for 159 days during March 2015 and February 2016 gives some interesting results. Unit commitment data of various GS scheduled by ERLDC [51] also give similar observations. Variation of generation of four GS (2 each of Coal and Hydel based) at different time blocks of 1st July, 2017 is given in Figure 3. Based on the difference between the maximum and minimum generation for all 954 unit commitment data, Table 1.3 can be obtained for which the following statements are valid. From Table 1.3 and Figure 1.3, it is clear that unit commitment data are highly repetitive in nature and there is a small variation between the consecutive elements.

i. For 4.51% of the available unit commitment data, all the 96 elements will be identical. Out of the 43 instances, there are 5 instances where the generation of the unit is zero for the entire day.

ii. For 21.49% of the available data, the difference will vary within 11-30 MW.

iii. For 76.73% available information, the difference is within 100 MW.

iv. It is only 19.39% when the difference varies within 101-200 MW and higher deviations in the elements are possible.

v. For 3.88% cases, the difference exceeds 200 MW but the majority of them is due to sudden generation failure where large repetition may occur.

Table 1.3

Survey of 954 unit commitment data of 6 GS during March 2015 and February 2016

Month	**Difference between maximum and minimum generation**							
	0	**1-10**	**11-30**	**31-50**	**51-70**	**71-100**	**101-200**	**>200**
March 2015	9	6	8	15	8	8	16	2
April 2015	1	5	14	7	13	12	19	7
May 2015	4	8	11	11	7	14	22	1
June 2015	4	9	25	8	5	9	11	7
July 2015	5	2	22	17	8	13	6	5
Aug. 2015	8	2	23	11	10	7	13	4
Sep. 2015	5	2	19	16	9	16	13	4
Oct. 2015	3	2	13	18	14	7	25	2
Nov. 2015	2	3	24	16	11	9	12	1
Dec. 2015	0	7	18	24	16	11	13	1
Jan. 2016	0	3	11	13	13	20	17	1
Feb. 2016	2	3	17	17	8	11	18	2

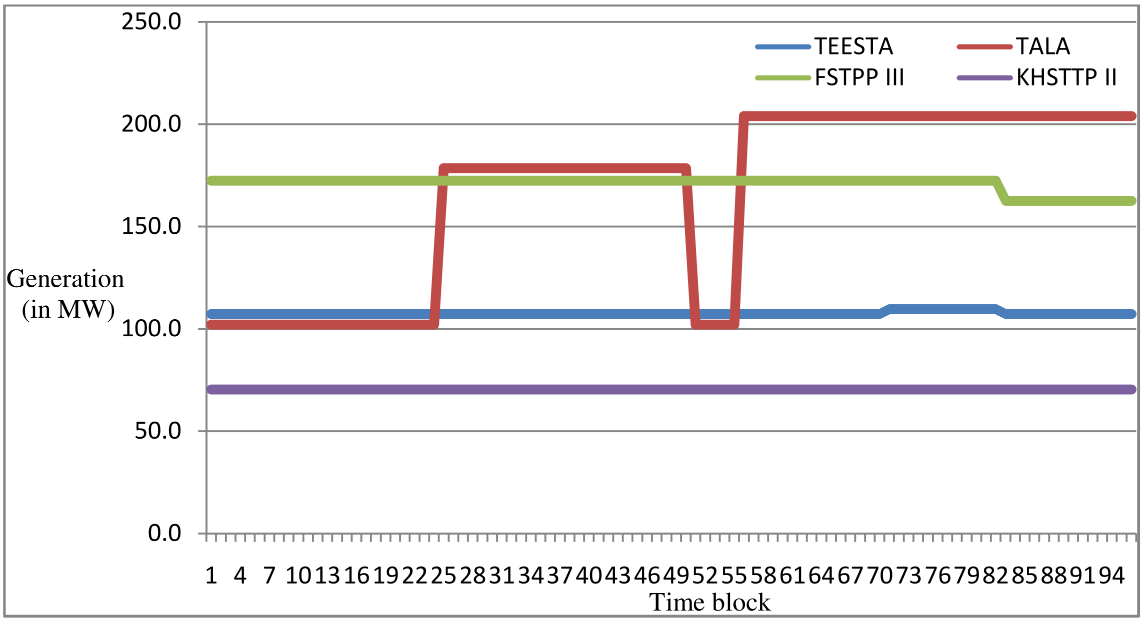


Fig. 1.3. Generation variation of four Generating Stations operating under ERLDC on 01/07/2016

It was mentioned repeatedly that unit commitment data are highly repetitive and thus difference between two consecutive elements will be zero for majority of differential values. The statement is justified by the unit commitment data for six different generating stations operating under WBSLDC during 1st and 7th February, 2016 is considered. From the analysis of 42 unit commitment data, 3990 differential values are obtained. The range of differential values for different GS is given in Table 1.4 which clearly establishes the statement. The direction of data transfer between LDC and Generating Station (s) and/ or Distribution Company (s) involved in scheduling data transfer is given in Figure 1.4.

Table 1.4

Variation of differential values of 42 unit commitment data of coal based GS

Range of differential values (in MW)	**No. of occurrences**						
	BTPS	**STPS**	**KTPS**	**BkTPP (1 to 3)**	**BkTPP (4 and 5)**	**SgTPP**	**Total**
0	599	602	621	603	604	598	3627
1-10	18	9	33	18	12	15	105
11-20	17	9	6	11	12	7	62
21-30	6	29	2	30	28	43	138
31-40	23	5	1	2	1	2	34
41-50	3	1	0	1	2	0	7
51-60	0	10	0	0	5	1	16
>60	0	0	1	0	0	0	1

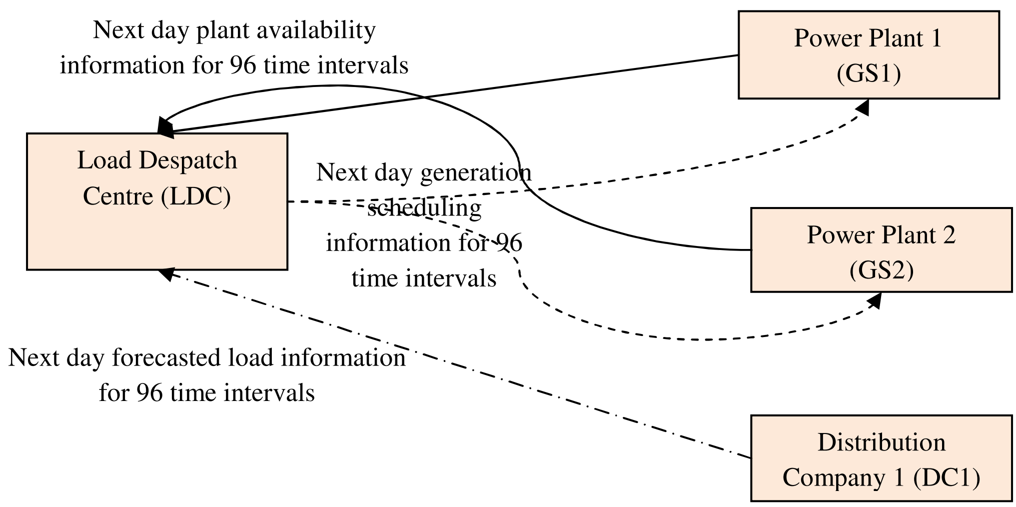


Fig. 1.4. Data transfer involved in generation scheduling

It is a fact that various parameters associated with power system are interrelated. Any imbalance in the active power of the system will result in the variation of system frequency. But variation in system voltage may occur when any imbalance in reactive power occurs. It implies that parameter monitoring is very crucial for power system operation. Accurate load forecasting and generation scheduling can reduce the unbalance to quite some extent. The responsibilities assigned to LDC clearly indicate that they are responsible for maintaining system stability. Conventional SCADA can acquire various parameters regularly at every 5 seconds or so and the data are stored in the data centre. Hourly variation of generation data of a thermal and hydel generating station (GS), power delivered by a 220kV line and system frequency for an entire day is given in Figure 1.5.

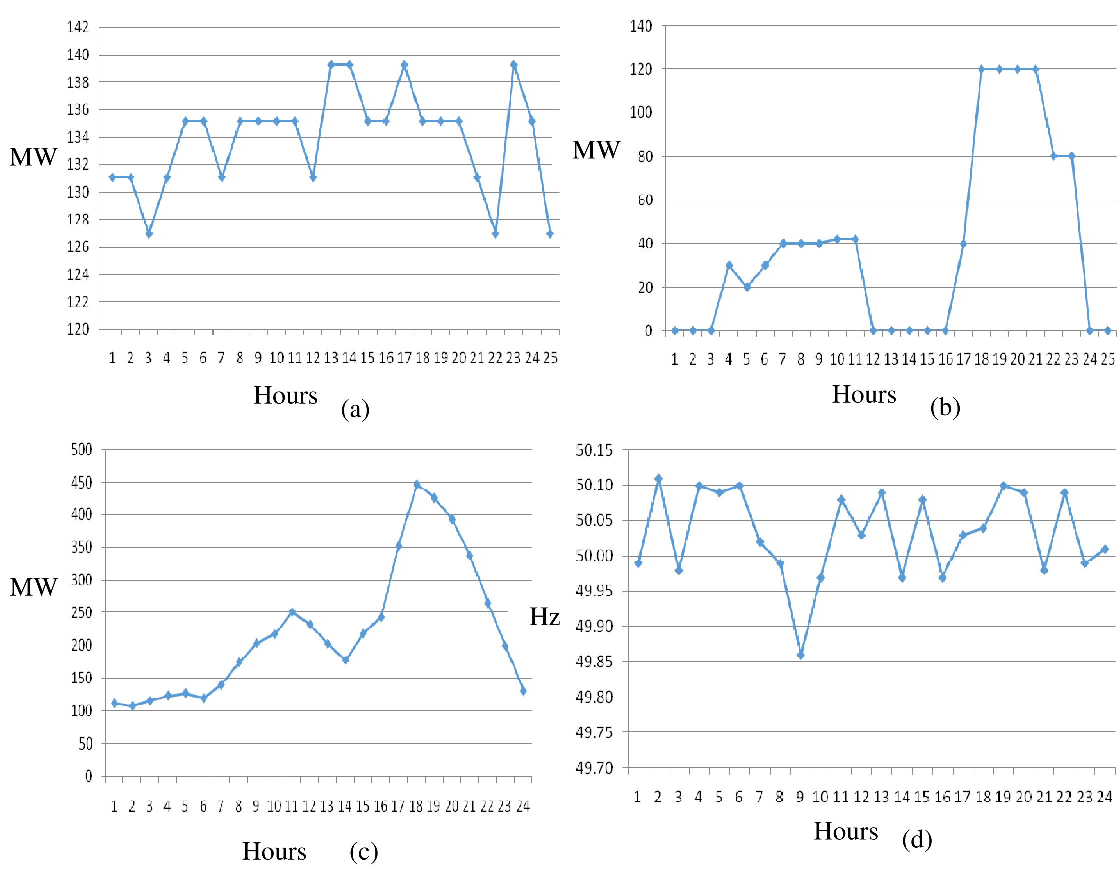


Fig. 1.5. Hourly variation of system parameters over a day: (a) MW generation of a thermal GS; (b) MW generation of a hydel GS; (c) MW transferred by a 220 kV line and (d) System frequency

1.4. Communication Techniques in Modern Power System

Any distribution automation system (DAS) involves the integrated technologies of computer, communication, and electric devices. In Korea Electric Power Corporation (KEPCO) several communication techniques including radio frequency, telephone line, cellular phone, power line communication and fibre-optic cable were considered for the DAS applications. In 2002, a prototype CDMA communication system was installed for DAS application to evaluate cost effectiveness and performance of CDMA mobile network in the real field. The CDMA channels are the least costly of all solutions when a user wants a small amount of data sent infrequently. It is a great fit for reporting alarms and simple monitoring and control. CDMA applications are best suited for capacitor banks, reclosers, line switches, fault locators, load management, outage detection and voltage regulators [52]. Various wired and wireless communication techniques like ZigBee, Wireless Mesh, GSM and Cellular network communication are important for smart grid application [53]. Conventional Power Line Carrier Communication (PLCC) is one of the most popular employed communication techniques in power system. But it does not meet the customers' demands due to its rugged communication environment in the condition of plenty of Intelligent Electronic Devices (IED) in modern distributed power system need to exchange large amount of information for system monitoring, protection and control. So, suitable alternative communication techniques are required in modern power systems. Ethernet is the modern trend now-a-days and internet based data transfer systems like IEC 60870 are used for data transfer [54, 55]. Developments in various communication systems have facilitated real time monitoring of conditions and performance of electric power system elements from generation side to end user levels. Four types of communication media, Power Line Carrier Communications (PLCCs), Optical Fibre, Satellite Communications and Wireless Communications were identified as possible candidates for automation [56].

In context of the Indian power system, Power Line Communication (PLCs), Microwave and Optical Fibre are the three modes of data transfer between LDCs present at various levels. In Power Line Carrier Communication (PLCC), power line acts as a channel where power frequency waves (conventional 50Hz supply) and high frequency communication waves exist simultaneously to transfer power as well as data from one place to another. It is already under operation of voice and data transfer through high voltage power line between substations (or LDCs) and was extremely popular till the last decade. But due to the limitations of PLCC, other communication techniques are replacing it and exist only in some circuits only. Some of its limitations are given below [57, 58].

i. Competing standards
ii. Booster is required for long distance data transfer
iii. Data security is not strong
iv. Possibility of interference with short wave radio signals
v. Limited rate of data transfer

Microwave communication is the transfer of information by employing EM waves with wavelengths varying from 30cm to 0.1cm and frequencies varying from 1GHz to 300GHz. It is widely used for point-to point communication as due to their smaller wavelengths, convenient type antennas direct them in narrow beams in the direction of receiving antenna. This feature enables nearby microwave equipment to use the same frequency without any interference. Due to high frequency of microwaves, the bandwidth is very high i.e. large information can be conveyed through it at any time instant. But due to the line of sight propagation, it can’t pass through hills or mountains, thereby limiting the use of this communication technology [59]. This results in the application of optical fibres for power system data communication.

The optical fibre communication is flexible and a number of fibres can be bundled as cables. In each fibre there might have multiple independent channels, each using different wavelengths of light by using wavelength division multiplexing (WDM). For long distance data transmission, optical fibre is a good choice as light attenuates a little while passing through the fibre and only a few repeaters are required to cover a large distance. The rate of data transmission in optical fibre is in the range of tens to hundreds GHz/s. As the optical fibre can be bent in any direction and have high speed data transmission, it overcomes the problems involved with microwave communication significantly [60]. So, the modern trend is to connect all the substations, LDCs and generating station with optical fibre network.

1.5. Literature Survey

With the advent of technology, newer technologies are emerging and the volume of data generated is increasing significantly in all domains. Obviously, handling of this enormous volume of data is extremely challenging. So, the concept of data compression is getting extensively popular for reducing this bulk volume of data and is employed in various applications in electrical power system. It is also a matter of fact that while approaching towards the formation of smart grid, data management and communication plays an important role. A decade back, data compression was a very hot topic of research in data communication, multimedia file conversion domain, etc. But with the increased need of data management in different applications including signal processing, data acquisition, power system etc. data compression had found its importance in all these domains. At the recent time, topics like application of data mining in power system, data acquisition followed by compression of ECG signals, PMU data compression, compression of load profile data etc. are not at all unusual. Now-a-days there is a search of newer low computational compression techniques, so that it can be implemented at the low level micro-controllers successfully. There are several works of applying some hybrid compression algorithm at embedded level. Some of the available works on data compression for different applications is discussed in the subsequent paragraphs.

Two data sets of fault data for high voltage (HV) lines and half-hourly electrical load readings for customers were investigated in [61] using data mining techniques. Fault database comprises of several thousand abnormality reports on HV power lines which includes data and time of interruption, method(s) of fault clearing, etc. On the other hand, half-hourly load databases comprise of meter readings every half hour over some years for various customers, tariff information of different customers, hourly weather conditions (temperature, wind speed, humidity, cloud cover and rainfall) including daily sunrise and sunset times, etc. during this period. Data mining was used to analyse daily load or time-of-day profiles according to weather, day type, month, and customer information. This process termed as load profiling and is employed primarily for load forecasting [61]. Data Mining will extract implicit useful knowledge from huge database and can produce better implementable systems for predicting a real value as well as to classify existing datasets. Wind turbine installation is constrained by multiple problems and locations with higher wind speed and frequent wind blow may not always be the right choice of installation. Economical, ecological and planning factors can reduce the importance of any site for wind turbine installations [62]. In [62] two data mining techniques are used to reduce dimensionality of datasets for deeper data insight and to predict the suitability of new wind turbine installation on the basis of historical facts. In [63], real system data (voltage, frequency, MW and VAR generation, system demand) are analysed to find interrelation between several system parameters. The recorded data was collected from Southern Regional Load Despatch Centre (SRLDC).

Several digital signal processing techniques are used to extract relevant features, characteristics and information from measured quantities and their waveforms for high-level monitoring, diagnosis, metering and modelling, predicting solutions, etc. The present trend of making an electrical grid more flexible, reliable, reconfigurable, efficient, secure, green, sustainable, intelligent, adaptable, and observable at all voltage levels and thus approaching towards smart (er) grids. The systematic combination of information and communication technologies and sensing, measurement, monitoring and control technologies to form the complex system (smart grid) that constitutes the electric power systems [64]. Embedded Zerotree Wavelet Transform (EZWT) -based technique is employed for the data denoising and compression of the 1-D signal of smart grid where selected, significant wavelet coefficients (WCs) are refined progressively and encoded by following the spanning tree structure through several dominant and subordinate passes [65]. A wavelet packet decomposition (WPD) is proposed in [66] for the analysis, denoising, and compression of power system data in the smart grid (SG) communication. It is a general framework that can compress up to 2% of the original data size and can remove noise with an improvement of up to 30dB in the signal to noise ratio (SNR). The algorithm can be implemented at FPGA level and the achievable compression ratio is within 12.5 and 14.5. A frequency compensated difference encoding is introduced in [67] that can reduce angle data entropy significantly. In this algorithm, preprocessing is followed by Golomb-Rice coding, which can routinely halve bandwidth and storage requirements for phasor angles without any loss of information [67]. In [68], high data compression ratio is obtained by processing the stationary and non-stationary components of a power waveform separately. Stationary components are identified by Lobe Slope detection and the stationary components are reconstructed. Integer Lifting Wavelet Transform (ILWT) is used for compressing the non-stationary component. Huffman Coding based compression compresses the Wavelet Coefficients further.

A system is developed in [69] where original power information for the whole distribution system can be reconstructed from the three modes of compressive sensing, where the power measurements are heavily compressed. Two novel methods based on Newton-Raphson algorithm, indirect and direct method of state estimation are developed from compressed power measurements. For smart metering application, a linear time complexity lossless algorithm, Resumable Load data compression algorithm (RLDA) is developed in [70] and gives satisfactory performance with majority of the data array. In this algorithm, differential coding is followed by zeroth order signed exponential-Golomb coding and Basic Arithmetic Coding based compression to achieve the compression ratio of about 40. A highly efficient lossy data compression is designed in [71] to store key information about load features and is termed as feature-based load data compression (FLDC). This will reduce the great burden on data transmittance, storage, processing, application, etc and increases the compression ratio to 55. Four schemes based on four variants of the arithmetic coding are proposed in [72] and is based on the fact that steady state power system data exhibits a high degree of correlation between consecutive measurement values. The hourly data is compressed by the four algorithms to achieve the results and it was indicated that the compression ratio varies over a wide range.

Apart from the algorithm discussed in [66], there are some other available works on the development of suitable data compression algorithms at embedded level. A DSP based processor is used to develop a prototype smart meter in [68] which can communicate through ZigBee, which is a low-power, low-bit-rate sensor network protocol. The range of compression ratio varies according to SNR and the nature is different for different type of supply voltage waveforms. In [73], small separated dictionary and variable mask numbers were used with the BitMask algorithm to reduce the codeword length of high frequency instructions and a novel dictionary selection algorithm is proposed to increase the instruction match rates. Compression ratio varies within the range of 56% - 76% for benchmarks on the ARM Cortex-A9 processor. But for C62×× and C64×× processor, this variation is 61%- 77% and 65% to 83% respectively. Design of a low power 3-lead ECG on-chip with integrated real-time QRS detection and lossless data compression for wearable wireless ECG sensors is discussed in [74] which can achieve an average compression ratio of 2.15 times with standard test data. An ECG signal acquisition system followed by a DSP unit containing the ECG compression algorithm is developed in [75], where the compression ratio of about 6.86 is achieved. In [76], a highly integrated VLSI implementation of a mixed bio-signal lossless data compressor is developed which is capable of handling multichannel ECG, EEG and DOT. The average compression ratio of about 2 can be achieved by the data compressor present in this portable, wireless brain-heart monitoring system.

There are few lossy and lossless compression algorithms in ECG data compression, which can compress an ECG data array to a character string before transferring the information through a suitable communication channel. In [77], a software based ECG data compression algorithm (LLEDCCE) is proposed where the ECG data array is compressed to character string. The compression ratio achieved by the algorithm is 7.18. Another software based ECG data compression algorithm, EDCCE is proposed in [78] which can give a compression ratio of 15.72. The combination of lossy and lossless compression algorithm (L2CRQ and LCNRQ respectively) will compress an ECG data array in [79] to form a character string and send the information through GSM communication. At the receiving end, the character string is decoded to obtain the actual ECG data array. The compression ratio achieved by this algorithm is 22.47. A lossless method based on modified American standard code for information Interchange (ASCII) character encoding for ECG data compression is proposed in [80]. The compression algorithm comprises of sign count, generation of array representing ECG sample's signs (+ve and -ve alternatively), adaptive amplification factor; and grouping method of ECG samples. For reconstruction of the ECG data reverse process is followed. The compression algorithm is tested with MIT-BIH Arrhythmia Database to obtain the results and the average compression ratio (CR) of about 8.38.

Generation scheduling of GS operating below any LDC is extremely important for the economic operation of power system. Based on the next day availability information, generation limit and the cost coefficients of various GS, experienced engineers of LDC prepare scheduling chart for individual units by considering next day load forecasting information and transmission loss. There are many literatures available on scheduling of generating station (GS) by different optimization techniques. Due to the simplicity of Particle Swarm Optimization (PSO), different methods were discussed in different literatures. A Fuzzy adaptive PSO (FAPSO) is developed in [81] for solving unit commitment (UC) problem where a Fuzzy based system is used for tuning the inertial weight in Iterative weight PSO (IW-PSO) for solving UC problem. A novel, short term scheduling optimization of cascaded hydroelectric stations based on Hybrid-PSO (HPSO) and chance constrained programming is proposed in [82]. Its objective is to determine a feasible and realistic operation of a set of coupled hydro plants at a specified risk level for the next 24 hours periods. A quantum inspired Binary-PSO (BPSO) is proposed [83], which is based on the concept and principles of quantum computing, such as a quantum bit and superposition of states to enhance the performance of the conventional BPSO. A novel PSO and Simulated Annealing (SA) based method [84] is used to find out the minimum cost for 3 GS system at different power demand. In [85], three improved PSO algorithms (IPSO-A, IPSO-B and IPSO-C) are developed and are implemented to solve economic load despatch problem for IEEE 5, 14 and 30-bus system. In [86], an ANN is trained with the data obtained with the Lambda Iteration method and the performance was analysed.

1.6. Objective of the present research

With the advent of technologies, there is a significant rise in the volume of information associated with the power system. Storage and transfer of such a bulky volume of information are often extremely challenging due to memory and bandwidth constraints. Data compression is a useful tool to overcome the problem and several lossy and lossless algorithms are proposed in various literatures. Practical data sets will be in the form of the data array and there are only a few works which can compress any data array. The available works for compressing power system operational data are based on entropy based compression techniques and thereby require probability distribution for decoding the information. The thesis is aiming for the following:

- Development of low computational, lossless compression algorithms for compressing large data array associated with the power system.
- Development of secured algorithms to ensure data security.
- Identification of different fields required for managing the information associated with power system operation.
- Realisation of a test bench to solve ELD problem of IEEE 5-bus system by employing a cloud based power system operational data management system.

1.7. Scope of the present work

- Analysis of practical power system operational data to develop low computational, lossless compression algorithms capable of obtaining a higher compression ratio (CR) with practical data sets.
- Management of large volumes of data is a challenge for the existing power system. Furthermore, a secured data communication is desirable in the recent times which are not often available in the existing system. Thus a compressed, encrypted data management system is developed in the laboratory for the management of generation scheduling and parameter monitoring data.
- For extracting the advantages of cloud computing, a cloud server based test bench was developed for power system operational data management.

1.8. Thesis Outline

Chapter 1 presents a brief introduction of the importance of power system operational data management system. The roles and responsibilities of LDCs and analysis of practical generation scheduling and parameter monitoring data are also included in this chapter. Different communication techniques used in power system are discussed in this chapter along with the objective and scope of the present dissertation. During the initial phase of work, works were based on entropy based lossless compression algorithms and testing was done by practical generation scheduling data. **Chapter 2** focuses on the analysis of generation scheduling data and description of Huffman Coding and Basic Arithmetic Coding based algorithms developed for compressing large data array. As the CR obtained with the entropy based algorithms were very low, it becomes necessary to search for another alternative algorithm. RLDA proposed for compressing load profile data from smart meters gives high CR with those data. While investigating the performance of RLDA with practical data arrays, a high CR is obtained as well. Based on practical data analysis, DBEA was developed for compressing large, slow varying data array. It is a novel, low computational approach which can compress the data array to a character string of much reduced size. DBEA fails with many practical data sets and require suitable modification. E-DBEA overcomes the limitations of DBEA at the expense of CR. **Chapter 3** describes these differential coding based lossless compression algorithms. Different possible communication techniques for transferring power system operational data are discussed in **Chapter 4**. This chapter also highlights the importance of cloud environment for data transfer and data management. Solution of ELD problem is extremely important for economic operation of the power system. Solution of ELD problem by using λ-iteration, IW-PSO and ANN is discussed in **Chapter 5**. For managing power system operational data, some parameters must be considered. The significance of these parameters is also described in the chapter. The results obtained by the works discussed in the previous chapters are included in **Chapter 6**. Development of DBEA compressed DCSK communication scheme for transferring power system operational data over power line is discussed in this chapter along with the obtained results. The development of RSA encrypted DBEA and realisation of Combined DBEA Encryption (C-DBEAE) algorithm is also included in this chapter. This chapter also includes the description of real time test bench for managing power system operational data by using Smart DBEA Encryption (S-DBEAE) algorithm and the necessary results. The thesis concludes in **Chapter 7** with future directives.

Chapter 2

POWER SYSTEM OPERATIONAL DATA MANAGEMENT BY ENTROPY BASED COMPRESSION ALGORITHMS

2.1. Introduction

With the advent of sensing systems, there is a significant rise in the volume of the measured quantity. Be it system automation or condition monitoring, sensing is a must. Modern day sensing can acquire data in a fraction of second and also have much higher accuracy. The concept of smart grid itself indicates the fact where data management and communication are extremely crucial in power system. There is a drastic change in power system operation as Remote Terminal Units (RTUs) are replaced by Phasor Measuring Units (PMUs) and Intelligent Electronic Devices (IDEs). Power line Communication (PLCC) was replaced previously by microwave communication which is now subsequently replaced by internet based data transfer like IEC 60870. Data compression was a hot topic in image processing, communication, etc. as it eliminates memory constraints. Presently ECG data compression, electric signal compression, smart meter data compression, etc. are the newest application of data compression. As smart grid deals with enormous volumes of data, a lot of work is going across the globe to manage this bulk volume of data and several lossy and lossless compression schemes are proposed to compress the information.

Data compression can be defined as the process of recombination of bits (or bytes) for making small compact data form. It can be either lossy or lossless compression schemes. In the lossy compression scheme, there is a significant reduction in data volume as non-significant information is identified and is eliminated. But for lossless data compression, statistical redundancy is identified and is eliminated to reduce the volume of data. Lossy compression is extremely popular for compression of multimedia files (i.e. image, audio and video files) and several algorithms are available. But for biological image compression, loss of information can be harmful and thus lossless compression is more suitable. Generally lossless compression is popular for compressing text files as no information loss is acceptable [44, 47, 87]. Power system operational data preservation is extremely important to test the mathematical models developed to describe different power system phenomenon [72]. This implies that no loss of information is desired and thus lossless compression algorithms must be employed for compressing those data. Among Basic Arithmetic Coding and Huffman Coding, the former is more effective in terms of achievable CR.

An algorithm based on Huffman Coding is developed in MATLAB for compressing a data array to an encrypted character string and maximum compression ratio of 1.87 is achieved [44]. Another algorithm based on Basic Arithmetic Coding is then developed [88] which will obtain a higher compression ratio (>2). These two lossless entropy based compression approaches are tested with practical data associated with the power system. Initial testing of these algorithms was performed in MATLAB environment to determine CR of these algorithms. Block diagram of the proposed system is given in Figure 2.1.

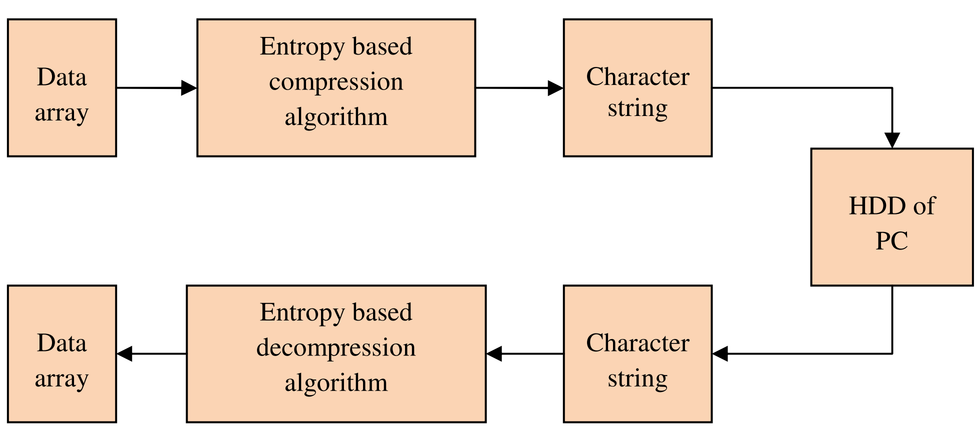


Fig.2.1.Block diagram of proposed entropy based compression algorithm

2.2. Generation scheduling data pattern

Generation scheduling information comprises of next day availability and unit commitment information of the GS. In India power system, a day is sub-divided into 96 time blocks of 15 minute duration. Both next day availability and unit commitment information are provided in the Website of any RLDA or SLDC. In Figure 1.3, variation of MW generation four central GS operating under ERLDC [51] is given. From the figure, it is clear that such data array comprises of 2 or 3 digit elements. Analysis of next day and unit commitment information of the GS operating under WBPDCL [50] also establishes the fact. Variation of next day availability and unit commitment information of BTPS, STPS and KTPS of WBPDCL operating below WBSLDC during 01/02/2018 and 15/02/2018 is given in Table 2.1. Figure 2.2 gives the variation of maximum generation of BkTPP and during 01/02/2018 and 15/02/2018.

Table 2.1

Variation of next day availability and unit commitment information of BTPS, STPS and KTPS

Date of February, 2018	BTPS		STPS		KTPS	
	Declared availability	Unit Commitment	Declared availability	Unit Commitment	Declared availability	Unit Commitment
1	260-280	195-280	462-462	443-462	580-750	532-750
2	50-280	50-280	462-462	319-462	600-760	532-744
3	250-280	195-280	462-462	462-462	600-600	532-600
4	290-310	195-310	462-466	462-466	580-620	532-620
5	300-310	195-310	462-462	319-462	570-615	532-615
6	290-310	195-310	462-462	409-462	600-620	532-620
7	230-310	195-310	462-462	462-462	590-630	532-630
8	275-290	195-260	462-462	462-462	570-600	532-600
9	240-300	195-300	462-462	462-462	518-600	518-600
10	220-300	195-300	462-462	462-462	530-590	530-590
11	280-310	195-310	462-462	462-462	565-600	532-600
12	260-300	195-300	462-462	460-462	600-600	532-620
13	185-280	185-280	462-462	325-462	600-630	532-630
14	250-300	195-300	462-462	462-462	630-630	532-630
15	245-310	195-310	462-462	462-462	630-630	532-630

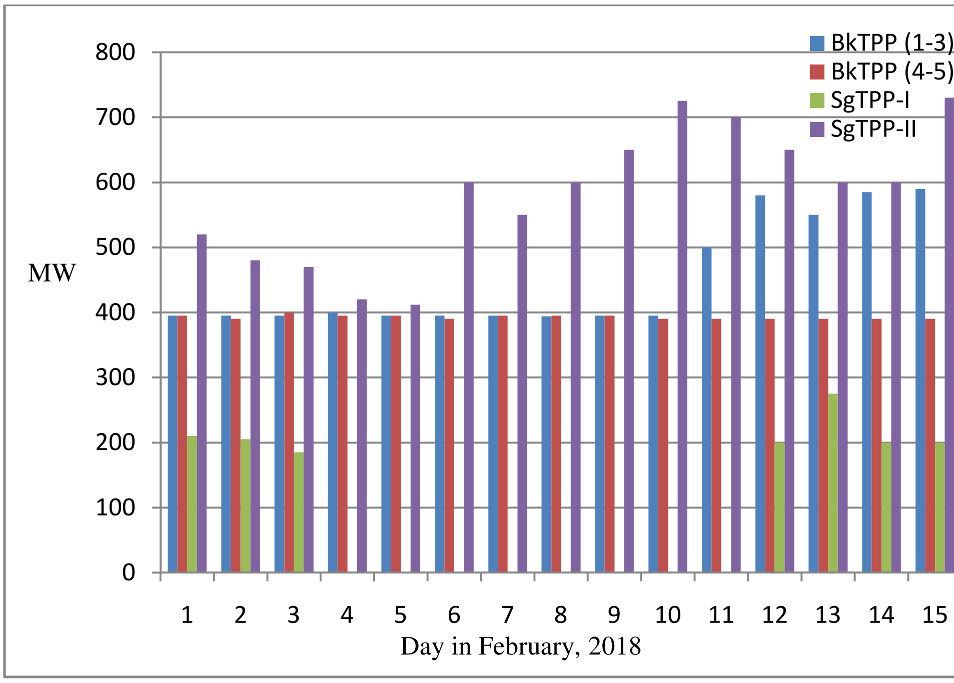


Fig. 2.2. Variation of maximum generation of BkTPP and SgTPP during 1st and 15th February, 2018

Analysis of the practical information considered for the preparation of Table 2.1 and Figure 2.2 indicates the following trends of scheduling data set:

i. The data sets comprise of 96 elements (i.e., multiple of 3 and 4).

ii. The elements contain 1, 2 or 3-digits only.

iii. The occurrence of any digit in the array is very random and any there is no regularity of occurrence.

These observations were the basis of the development of compression algorithms for compressing large data array to a character string. Initially, Huffman Coding was used to compress the information and a lower CR was achieved for the majority of the available data. The algorithm was modified in the subsequent version, where Basic Arithmetic Coding was used to compress the array. While testing the algorithm with practical unit commitment data, it was observed that the CR obtained with Basic Arithmetic Coding based algorithm was higher than that obtained with Huffman Coding based algorithm.

2.3. Entropy based Compression: Huffman and Basic Arithmetic Coding

Huffman Coding is a lossless data compression technique proposed by David A. Huffman in 1952. The steps followed during the formation of Huffman tree is bottom up instead of top down, which was the limitation of Shannon- Fano Coding. Assigning the code word in Huffman Coding is based on the following observations [87, 89]:

i. Symbols occurring frequently will have shorter code words
ii. Least frequently used symbols will have same code length.

For a symbol set s = {a, b, c, d, e} with probability set of the symbols in set s is p = {0.3, 0.25, 0.2, 0.15, 0.1}, the Huffman tree is formed according to the probability distribution according to which code word for each symbol is determined. Huffman tree corresponds to the probability distribution can be formed by following the steps illustrated in Figure 2.3 [87, 89, 90]. Based on this tree, code words for different is obtained which will be as given in Table 2.2.

Table 2.2

Symbol Code

Sl. No.	Symbol	Binary Code
1	a	11
2	b	10
3	c	00
4	d	011
5	e	010

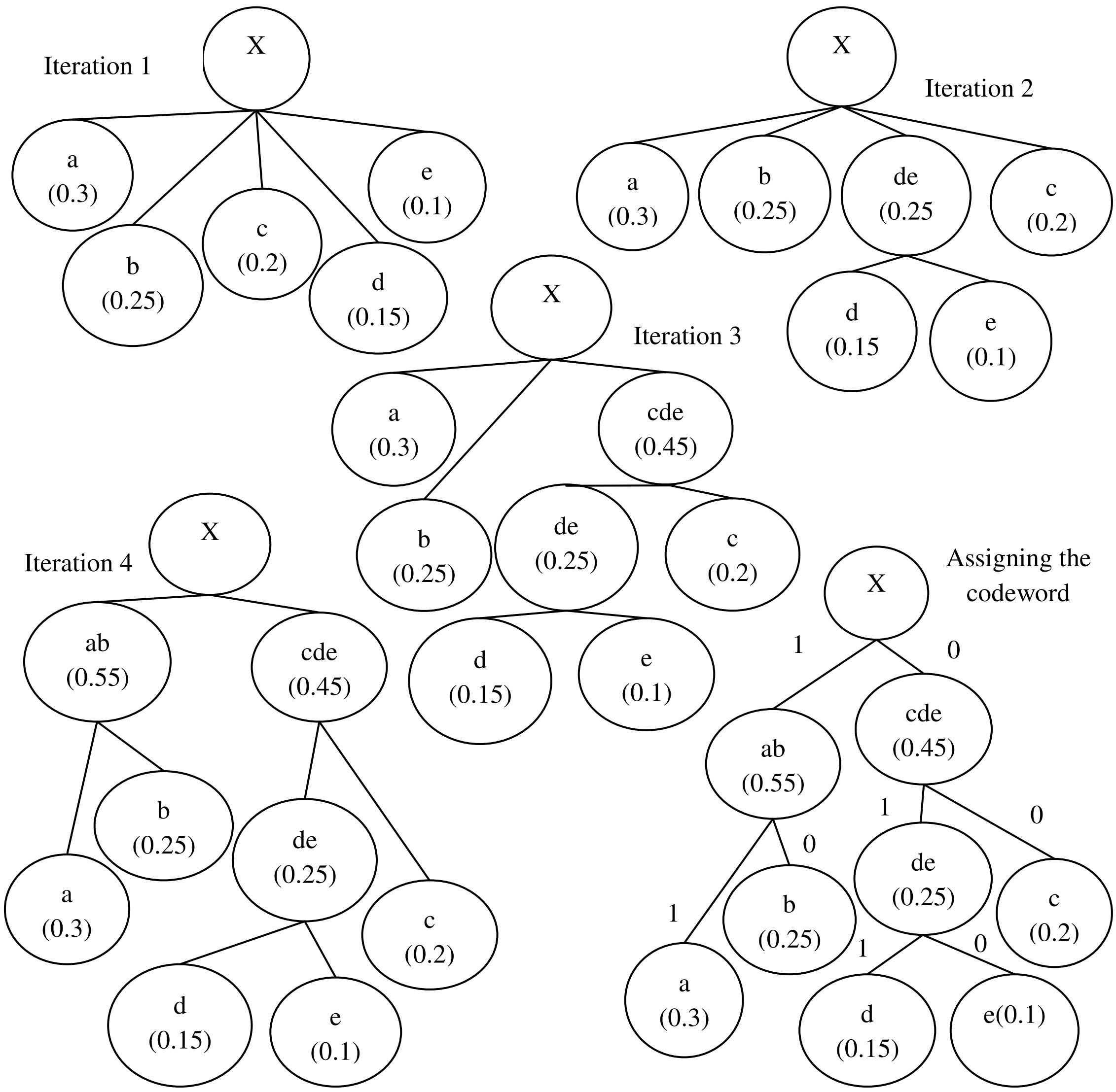


Fig. 2.3. Steps of formation of Huffman tree of symbol set s

So, for a symbol stream 'bad', the corresponding binary stream will be the combination of binary code corresponds to 'b', 'a' and 'd' respectively i.e. '1011011'. At the decoding end, the pseudo code to be followed is as given below [87, 89, 90].

Input: bin[n]

Output: str- Symbol stream

Begin

    Initialize i=1 and str= NULL

    Repeat until i< n+1

        Based on the twig weight i.e. bin[i], parent node is reached

        Repeat while no child nodes available

```
                If no child nodes found
                        str= Concatenate str and symbol corresponding to node
                        reached
                Otherwise
                        i= i+1
                end
        end
    end
End
```

For any data array, possible symbols being encountered will lie between 0-9. If the probability of occurrence of all these symbols is assumed to be equal, i.e. 0.1 for each symbol, the corresponding symbol code table will be as given in Table 2.3.

Table 2.3

Symbol Code for decimal values

Symbol	0	1	2	3	4	5	6	7	8	9
Binary code	011	010	0001	0000	0011	0010	101	100	111	110

Basic arithmetic coding is a lossless data compression technique where a character string is converted in the form of a fractional value lying in between 0 and 1. For a source message comprising of symbols of symbol set S {a, b, c, d, e, h} with probability of occurrence be as shown in Table 2.4, the steps of compression and decompression are discussed in the sub-sequent sections [87, 91].

Table 2.4

Probability distribution table for symbol set, S

Sl. No.	**Character**	**Probability**	**Cumulative probability**	**Range [r_low, r_hi)**
1	a	0.4	0.4	[0, 0.4)
2	b	0.2	0.6	[0.4, 0.6)
3	c	0.1	0.7	[0.6, 0.7)
4	d	0.1	0.8	[0.7, 0.8)
5	e	0.1	0.9	[0.8, 0.9)
6	h	0.1	1.0	[0.9, 1.0)

The pseudo code for compressing a string with characters as in Table 2.4 by using Basic Arithmetic Coding is given below [91].

```
Input: str
Output: val- Float number

Begin
        l= Length (str)
        Initialize variables min= 0, max= 1 and r= 1
        Initialize i=1
        Repeat until i<l+1
                x= str(i)
                Obtain r_low and r_hi corresponds to x
                min= min + r * r_low
                max= min + r * r_hi
                r= max – min
                i= i+1
        end
        Compute num such that min < num < max having minimum binary string length
End
```

From the float value, the character string is decoded by following the given pseudo code [87].

```
Input: val; l
Output: str

Begin
        Initialize i=1 and str= NULL
        Repeat until i<l+1
                x= Character corresponds to val obtained from Table 2.4
                str= Concatenate str and x
                l= r_low (x)
                h= r_hi (x)
```

```
            r= h –l
            val= (val – l) / r
            i= i+1
    end
End
```

2.4. Methodology

The previous section illustrates the algorithms for compressing a character string to binary bit string. As the purpose is to compress a character string to a character string, suitable modification must be needed to serve the purpose. Both approaches are similar, except compression algorithm and based on a fundamental observation that a majority of generation scheduling data encountered are 3-digit elements. As the size of the data array is 96, it is possible to divide it to 24 character strings each containing 12 characters. These character strings are then compressed by Basic Arithmetic or Huffman Coding based compression algorithm. These binary strings are added with few identity bits and then converted to a character string comprising of characters with ASCII value varying between 0-127. These equal character strings are cascaded to form the actual encrypted character string. At the decoding end, encrypted character string is subdivided into equi-length strings which are decoded individually to obtain four consecutive array elements. This decoding requires identity bits included before binary bit strings to decode the information. These elements are then combined to obtain an entire scheduling array being encoded. It is to be noted that both encoding and decoding requires probability distribution of the characters being encountered. For the sake of simplicity, the probability of encountering all the possible characters 0-9 is considered to be equal i.e. 0.1.

The algorithm for compressing a large data array using Huffman Coding based algorithm is given below. The algorithm is based on the observation that the binary string length corresponds to practical 12 bit number will not exceed 50.

Input: arr- Data array containing 3*l elements (l being any integer)
Output: char- Character string containing 8* l

```
Begin
        Add zero (s) before each element of arr so that each element will have 4-digits
        Define str= NULL and char= NULL
        Initialize i= 1
        Repeat until i< l+1
                str= Concatenate arr [3*i], arr [3*i+1] and arr [3*i +2]
                numstr= 12-digit decimal number corresponds to str
                bin= Binary equivalent of numstr obtained by Huffman Coding
                lb= Bit length of bin
                p= 6-bit binary equivalent of lb
                modbin= Binary array comprising of (50-lb) number of zeros and bin
                binstr= 56-bit binary array formed by appending modbin with p
                Initialize j= 1
                Repeat until j<57
                        svnbin= Binary array containing jth to (j+6)th element of binstr
                        deci= Decimal equivalent of svnbin
                        ch= Character equivalent of deci
                        char:= Concatenate char and ch
                        j= j+7
                end
                str= NULL
                i= i+1
        end
End
```

The pseudo code for extracting the actual data array from the character string by using Huffman Coding is given below.

Input: char

Output: arr

```
Begin
        strlen= Length of char
        l= strlen/8
```

```
    Define a zero arrayarr(3*l)
    Initialize i= 1
    Repeat until i< l+1
        tembin= NULL
        Initialize j= 1
        Repeat until j<9
            deci= Decimal equivalent of char(8*i+j)
            bin= 7-bit binary equivalent of deci
            tembin:= Concatenate tembin and bin
            j= j+1
        end
        sxb= Binary string containing first six bits of tembin
        dsx= Decimal value corresponds to sxb
        rslt= Binary string containing last dsx bits of tembin
        decstr= Decompress rslt by using Huffman decoding
        Initialize k= 1
        Repeat until k> 3
            dec= Decimal equivalent of k^th 4 character combination of decstr
            arr(3*i+k)= dec
            k= k+1
        end
        i= i+1
    end
End
```

The pseudo code for compressing a large data array using Basic Arithmetic Coding based algorithm is given below. The algorithm is based on the observation that the binary string length corresponds to practical 12 bit number will not exceed 43.

Input: arr- Data array containing 4*l elements (l being any integer)

Output: char- Character string containing 7* l

```
Begin
    Add zero (s) before each element of arr so that each element will have 3-digits
    Define str= NULL and char= NULL
    Initialize i= 1
    Repeat until i< l+1
        str= Concatenate arr [4*i], arr [4*i+1], arr [4*i +2] and arr [4*i +3]
        numstr= 12-digit decimal number corresponds to str
        bin= Binary equivalent of numstr obtained by Binary Arithmetic Coding
        lb= Bit length of bin
        p= 6-bit binary equivalent of lb
        modbin= Binary array comprising of (43-lb) number of zeros and bin
        binstr= 49-bit binary array formed by appending modbin with p
        Initialize j= 1
        Repeat until j< 50
            svnbin= Binary array containing j^th to (j+6)^th element of binstr
            deci= Decimal equivalent of svnbin
            ch= Character equivalent of deci
            char:= Concatenate char and ch
            j= j+7
        end
        i=i+1
    end
End
```

The pseudo code for extracting the actual data array from the character string by using Basic Arithmetic Coding based algorithm is given below.

Input: char

Output: arr

```
Begin
    strlen= Length of char
    l= strlen/8
    Define a zero array arr(4*l)
```

```
    Initialize i= 1
    Repeat until i< l+1
        tembin= NULL
        Initialize j= 1
        Repeat until j<8
            deci= Decimal equivalent of char(7*i+j)
            bin= 7-bit binary equivalent of deci
            tembin:= Concatenate tembin and bin
            j= j+1
        end
        sxb= Binary string containing first six bits of tembin
        dsx= Decimal value corresponds to sxb
        rslt= Binary string containing last dsx bits of tembin
        decstr= Decompress rslt by using Huffman decoding
        Initialize k= 1
        Repeat until k>4
            dec= Decimal equivalent of k^th 3 character combination of decstr
            arr(4*i+k)= dec
            k= k+1
        end
        i= i+1
    end
End
```

Chapter 3

DIFFERENTIAL CODING BASED ALGORITHMS FOR LARGE DATA ARRAY COMPRESSION

3.1. Introduction

In the previous chapter, attempts were made to compress a large data array to a character string. The low CR was the major drawback of the entropy based compression algorithms as there are few lossless algorithms are available in the literatures where moderate to high CR is achieved while compressing a data array. In some works of compressing ECG data array to a character string [77-79], CR above 7 is achieved. In a couple of works, a data array associated with power system monitoring [63] and smart metering [70] are compressed by Arithmetic Coding based algorithms and a high CR is obtained. Some similarities were observed in both the works. Similar to Arithmetic Coding based compression algorithms [63], normalization of the float data array is performed in RLDA [70]. RLDA comprises of differential coding, which is similar to linear predictors used in [63] for predicting the next value of the sequence. In RLDA, differential coding is followed by variable length coding and adaptive binary arithmetic coding. In [92], the performance of RLDA was analysed for compressing practical power system operational data and a considerable CR was obtained with the majority of available data. Though a high CR was obtained with RLDA for majority of available data, it is not a good choice for compressing power system operational data. Due to the following drawbacks, RLDA is not suited for compressing power system operational data.

i. High computation burden, particularly with large data array.
ii. Use of Adaptive Binary Arithmetic Coding for compression requires probability distribution table for decoding the information.
iii. For a zero array, the output RLDA is 1 from which actual data array can't be decoded.

Differential Coding plays a key role in reducing the volume of information in RLDA. Based on the features of various practical data, a Differential Coding based low computational lossless compression algorithm, DBEA is tailored for compressing a large data array to a character string. Lower computational burden and high CR for highly repetitive slow varying data sets makes it a good choice for compressing power system operational data. When the difference between consecutive elements exceeds a threshold, DBEA fails. This limits the application of DBEA for compressing information under transient conditions. In the extended version of DBEA (E-DBEA), this range is extended considerably at the expense of CR. None of the proposed algorithms require probability information for decoding the information. Thus, they can be a good choice for both data storage and data transfer.

Load Despatch Centres (LDCs) at national, regional and state level are assigned to separate responsibilities required for the operation, planning, monitoring and control of the power system. Data analysis and data transfer are extremely crucial for system monitoring and storage of such enormous volume of data is extremely challenging. According to Indian Electricity Grid Code (IEGC), some operations to be performed by State Load Despatch Centre (SLDC) which includes daily scheduling and operational planning, optimum scheduling and dispatch of electricity within a state, Power system parameter monitoring for grid operations, accounting of electricity quantity transmitted through state grid and real time operation for grid control and dispatch of electricity within the state [39]. Most of the mathematical models used in Electric power system for analysing/ predicting system behaviour are based on several assumptions. Validation of such models is possible only when a detailed measurement of power system operational data is available. Storage and retrieval of data associated with power system is a challenging task [72]. Data compression is extremely popular for the applications dealing with large volume of data as it reduces data volume and enables better utilization of bandwidth. Reduced volume of data, lower transfer time and data security is important for data transfer [70]. If monitoring data can be compressed at field level with suitable compression techniques, memory requirement reduces significantly and compressed data transfer between field devices and data centre can be realised.

The available works [70-72, 91, 92] are based on offline compression of practical data for eliminating data storage constraints. Real time testing of DBEA and E-DBEA between two computers is performed successfully. Increased computational burden and use of entropy based compression algorithms restricts their application for data management at low level field device end. In modern day SCADA, microcontroller based data acquisition is extremely common, and this data is transferred to data centre through the suitable communication channel. If field devices can compress steady state data collected over a finite time and transfer the compressed information instead, there will be reduced energy and bandwidth requirement. As DBEA and E-DBEA can be implemented in embedded microcontrollers, it is possible to develop a DBEA or E-DBEA based data acquisition system (DAS) where data acquired during a time block will be compressed and transmitted to receiving end. Block diagram of the proposed system is given in Figure 3.1.

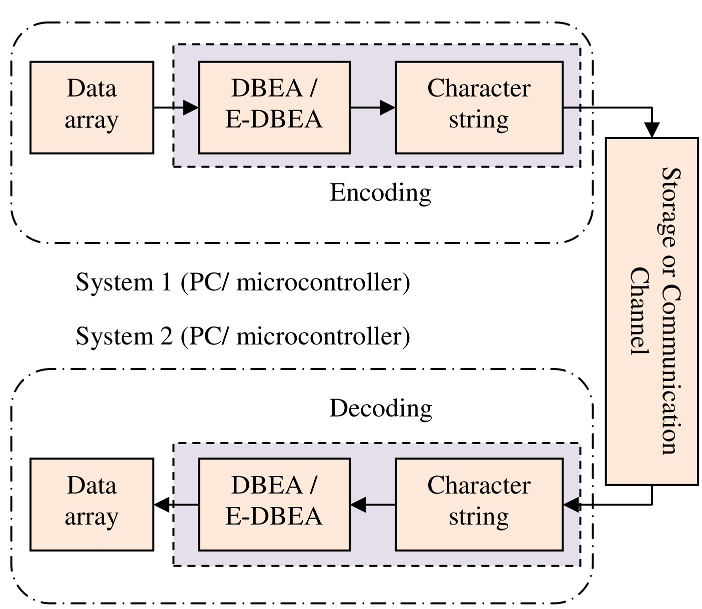
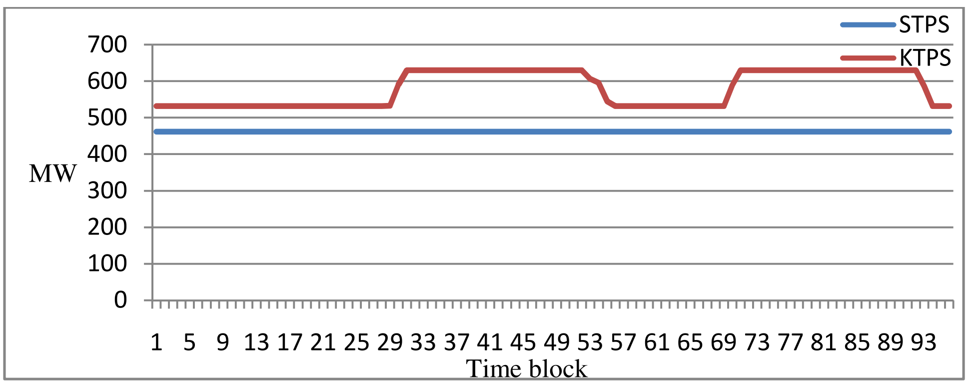
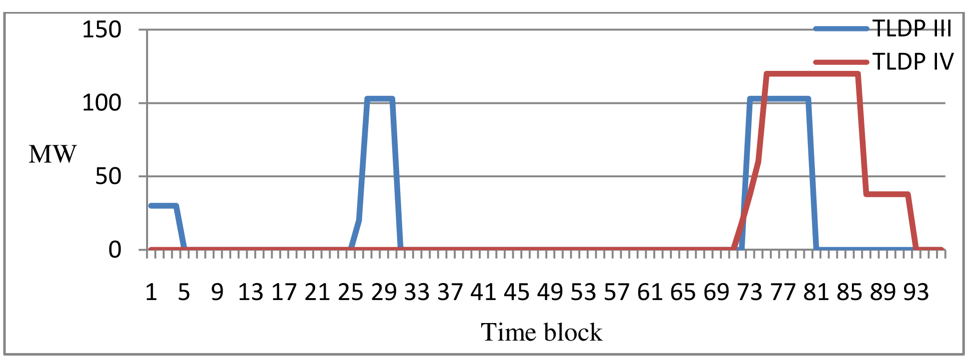


Fig. 3.1. Block diagram of Differential Coding based compression algorithm

In Figure 3.2 (a), the unit commitment information of two GS over an entire day is given where range of variation is very low. Such slow varying information can be compressed successfully by DBEA compression scheme. There are some instances where there is a sudden variation of generation. Variations of scheduled information about a couple of the unit commitment information are given in Figure 3.2 (b). It is obvious that E-DBEA will be the only choice for compressing such information.



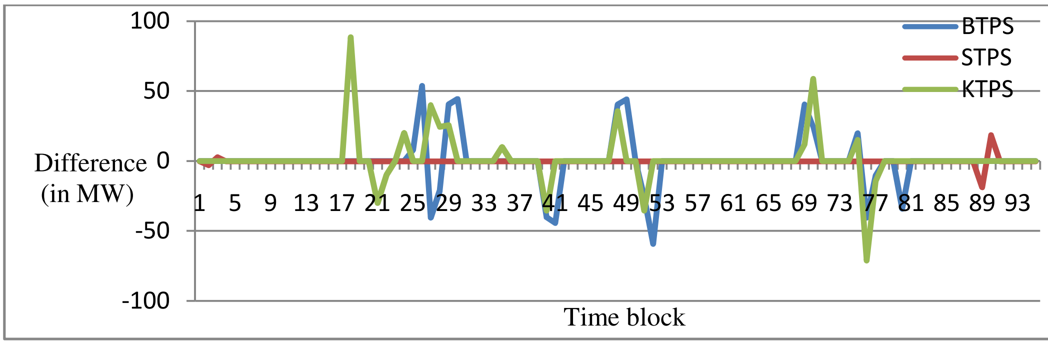
(a)



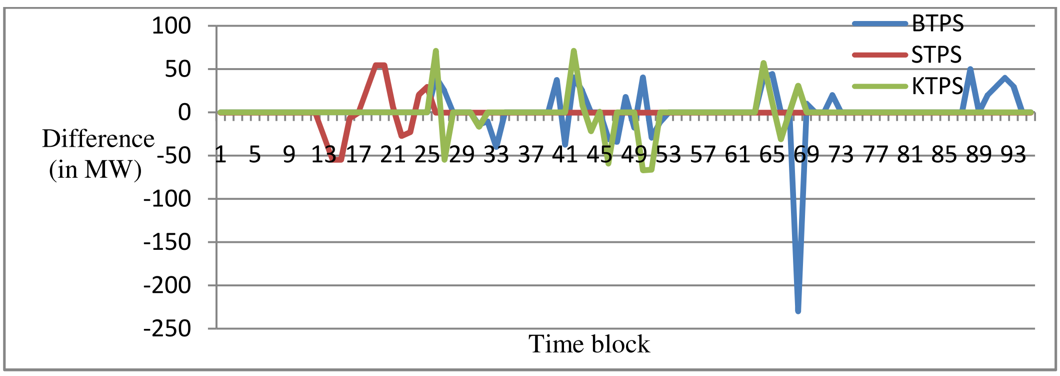
(b)

Fig. 3.2. Variation of MW generation on 14/03/2018 (a) Slow varying generation and (b) Sudden varying generation

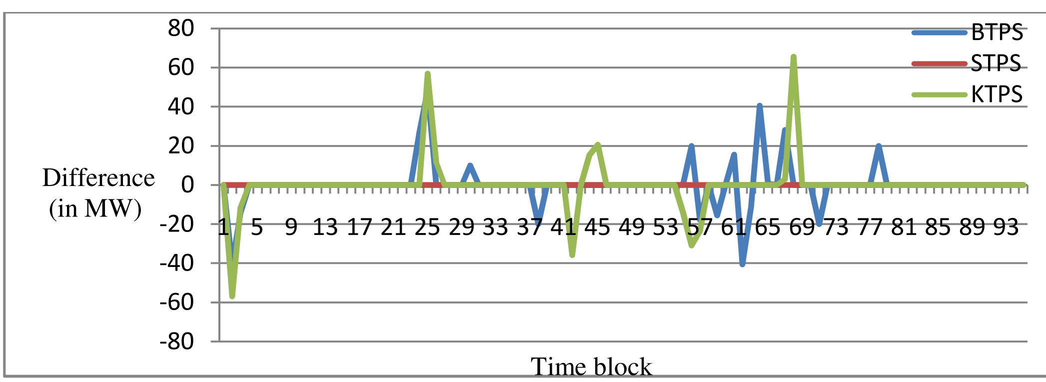
DBEA is based on the difference value between two consecutive elements of the data array. In the chapter, analysis of such differential values is done from where it is clear that it generally varies over a small range. Variation of difference value for unit commitment information of three GS, BTPS, KTPS, and STPS during 1st February, 2018 and 4th February, 2018 is given in Figure 3.3. It is clear from the figure that there is a very high repetition in the data array and this feature is exploited in DBEA and E-DBEA to compress such data array.



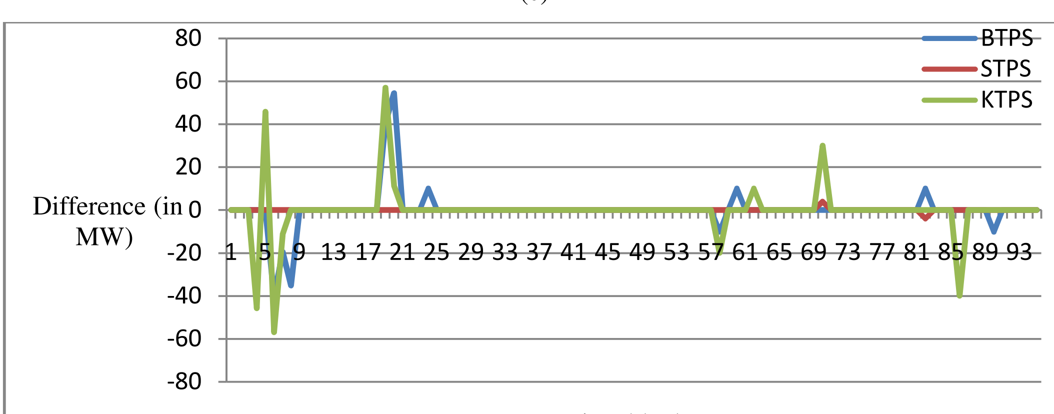
(a)



(b)



(c)



(d)

Fig. 3.3. Differential values of unit commitment data array of BTPS, STPS and KTPS on: (a) 1st February, 2018; (b) 2nd February, 2018; (c) 3rd February, 2018; and (d) 4th February, 2018

3.2. Resumable Load Data Compression Algorithm (RLDA)

During analysis of load profiles associated with consumer households, it was observed that the difference of most of the consecutive elements is often very small. If the sampling frequency would be lower, this difference will increase [70]. As the power system operational data also exhibit similar feature, it can also be compressed by RLDA successfully. The steps employed in RLDA for compressing data array are given in Figure 3.4 (a) and the pseudo code is given in subsequent steps.

Input: ar[l]- Data array

Output: bin- Binary equivalent array; Probability distribution table

```
Begin
        Initialize i=1
        Repeat until i<(l+1)
                a[i]= Multiply ar[i] by powers of 10 to obtain integer value
                i=i+1
        end
        dif[1]=a[1]
        Initialize j=2
        Repeat until j<(l+1)
                dif[j]= a[j]-a[j-1]
                j=j+1
        end
        com= NULL
        Initialize k=1
        Repeat until k<(l+1)
                code= Codeword corresponds to dif[k] obtained by Zeroth order signed
                exponential Golomb code
                com= Append com with code
                k=k+1
        end
        Determine probability distribution table according to com
        bin= Binary array obtained by applying Binary Arithmetic Coding on com
End
```

In Zeroth order signed exponential Golomb code, any positive or negative number can be converted to binary codeword by following the pseudo code given below. The binary code words of some decimal values are given in Table 3.1.

Input: num

Output: code

```
Begin
        If num>0
                bin= Binary equivalent of 2*num
        Otherwise
                bin= Binary equivalent of (2*num+1)
        end
        l= Length(bin)
        code= Append (l-1) zeros before bin
End
```

Table 3.1

Binary code words for some decimal numbers

Decimal value	Binary Codeword
-4	0001001
-3	00111
-2	00101
-1	011
0	1
1	010
2	00100
3	00110
4	0001000

In Binary Arithmetic Coding, a binary string is encoded in the form of a fractional value between 0 and 1. For a binary symbol set {0, 1} with probability distributions of {0.6, 0.4}, for a symbol string of {0110}, fractional value will be in between 0.504 and 0.5616. Different iterations involved in Binary Arithmetic Coding for computing the fractional value for the symbol string {0110} are given in Figure 3.4. The final result will be $(10001)_2$ i.e. the binary equivalent of 0.53125 (between 0.504 and 0.5616) having a minimum string length [87]. In adaptive entropy based compression algorithms, probability distribution of individual strings is computed separately according to which encoding of that string is performed.

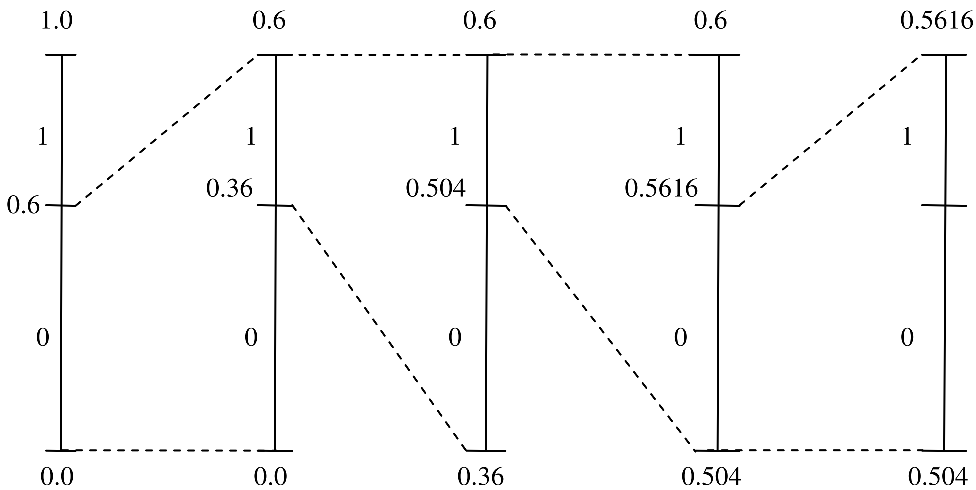


Fig.3.4. Steps of iteration in computing Basic Arithmetic Coding for binary symbol string

For decoding the actual data from the binary string, the following steps as given in Figure 3.5 (b) are followed. The necessary pseudo code is given below.

Input: bin; Probability distribution table

Output: arr[l]

```
Begin
        com[p]= Decompress bin by using Binary Arithmetic Coding
        Initialize i=1, cn=1
        Repeat until i<p+1
                Initialize cnt=0
                Repeat until com[p]≠ 0
                        cnt=cnt+1
                end
                rd[cnt+1]= Binary array containing (i+cnt)th and (i+2*cnt+1)th element of
                com
                dif[cn]= Integer corresponds to rd obtained by Golomb code
                cn=cn+1
                cnt=cnt+2*cnt+1
```

```
        end
        ar[1]=dif[1]
        Initialize j=2
        Repeat until j<cn+1
                ar[j]=ar[j-1]+dif[j]
                j=j+1
        end
        Initialize k=1
        Repeat until k<cn+1
                arr[k]=ar[k]/powers of 10
                k=k+1
        end
End
```

Decoding of an integer from a binary codeword by employing Zeroth order signed exponential Golomb code is possible by following the pseudo code given below.

```
Input: bin
Output: num

Begin
        int= Decimal equivalent of bin
        If int is even
                num= int/2
        Otherwise
                num=(1-num)/2
        end
End
```

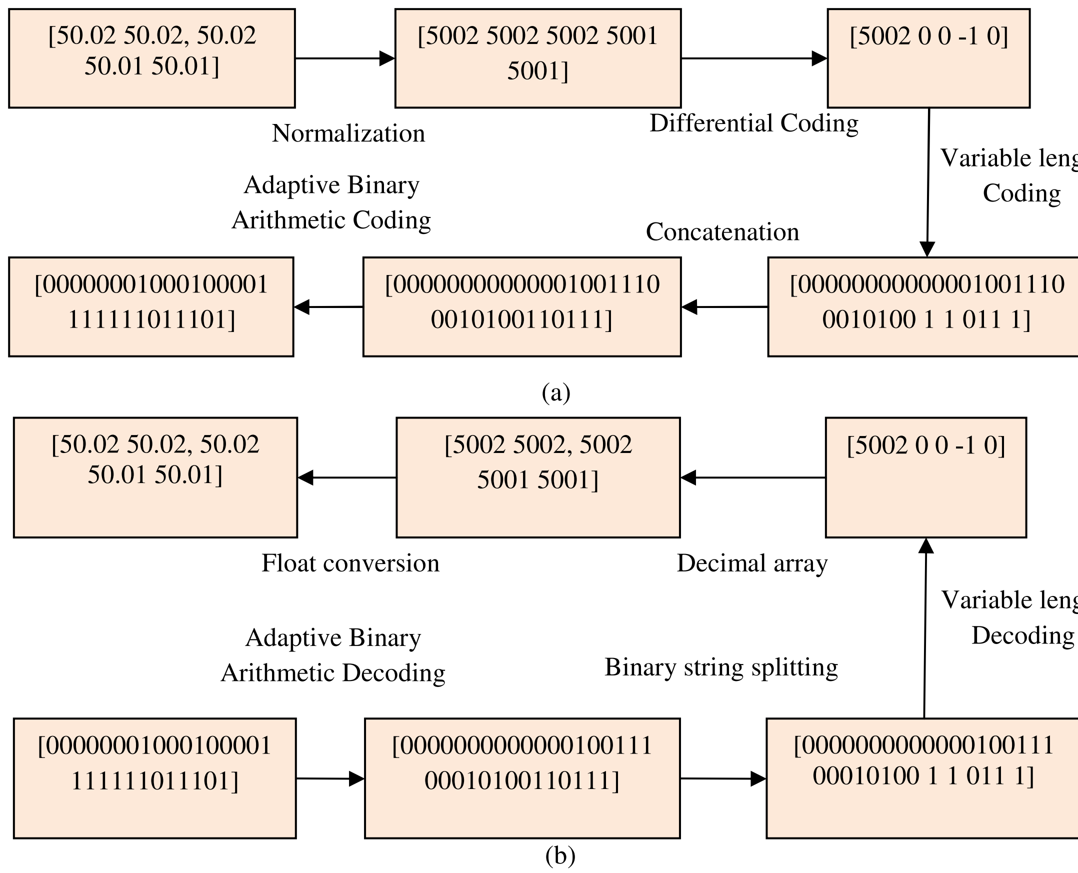


Fig. 3.5. RLDA compression scheme: (a) Encoding and (b) Decoding

3.3. Differential Binary Encoding Algorithm (DBEA)

For any repetitive data array, the array obtained by performing differential coding will have a large number of consecutive zeros in it. If zero elements in differential array are replaced by their count, it is possible to reduce the output size without introducing any computational complexity. This is the fundamental idea behind the development of DBEA for compressing large repetitive data array. From Figure 3.3, it is clear that the unit commitment information is repetitive in nature and thus higher extent of compression can be achieved by DBEA for majority of the data array. The statement is valid for power system operational data as well. Block diagram of this scheme is given in Figure 3.6 (a) and the necessary pseudo code is given below [93, 94].

Input: arr[l]

Output: str

```
Begin
        dif[1]=arr[1]
        Initialize i=2
        Repeat until i<(l+1)
                dif[i]= arr[i]-arr[i-1]
                i=i+1
        end
        str= NULL
        bin[16]= Binary equivalent of dif[1]
        asci[1]= Decimal equivalent of first 8 elements of bin
        asci[2]= Decimal equivalent of last 8 elements of bin
        Initialize j=2 and ascn=3
        Repeat until j<(l+1)
                Initialize cnt=0
                If dif[j]= 0
                        cnt=cnt+1 until dif[j+1]≠0
                        j=j+cnt
                        asci[ascn]=cnt
                        ascn=ascn+1
                If dif[j]< 0
                        po[6]= Binary equivalent of dif[j]
                        bidc[8]= Include identity elements ‘1 0’ with po
                        asci[ascn]= Decimal equivalent of bidc
                        ascn=ascn+1
                        j=j+1
                Otherwise
                        ne[6]= Binary equivalent of -1*dif[j]
                        bidc[8]= Include identity elements ‘1 1’ with ne
                        asci[ascn]= Decimal equivalent of bidc
                        ascn=ascn+1
                        j=j+1
                end
```

```
        Initialize k=1
        Repeat until k<ascn+1
            ch= Character equivalent of asci[k]
            str= Concatenate str and ch
        end
    end
End
```

Table 3.2

Binary Encoding

Sl. No.	Type of element	Identity bit	No. of binary bits	Range of values
1	First element	NA	16	0-65535
2	Zero count	0	7	0-127
3	Positive difference	10	6	0-63
4	Negative difference	11	6	0-63

Unlike RLDA, DBEA do not require probability distribution table for decoding the actual data. The original information is extracted from the compressed character string by following the steps given in Figure 3.6 (b) and the necessary pseudo code is given below [93, 94].

```
Input: str
Output: arr[l]

Begin
    ls= Length(str)
    Initialize i=1
    Repeat until i<ls+1
        asci[i]= ASCII value of i^th character of str
    end
    bfe[16]= Combined binary equivalent of asci[1] and asci[2]
    dif[1]= Decimal equivalent of bfe
    Initialize j=3 and dfcn=2
```

```
        Repeat until j< ls+1
                bidc[8]= Binary equivalent of asci[j]
                If first element of bidc is 0
                        zc= Decimal equivalent of next 7 elements
                        Initialize z=1
                        Repeat until z<zc+1
                                dif[dfcn]=0
                                dfcn=dfcn+1
                                z=z+1
                        end
                Otherwise
                        If second element is 0
                                pn= Decimal equivalent of next 6 elements
                                dif[dfcn]=pn
                                dfcn=dfcn+1
                        Otherwise
                                nn= Decimal equivalent of next 6 elements
                                dif[dfcn]=-1*nn
                                dfcn=dfcn+1
                        end
                end
                j=j+1
        end
        arr[1]=dif[1]
        Initialize k=2
        Repeat until k<cn+1
                arr[k]=arr[k-1]+dif[k]
                k=k+1
        end
End
```

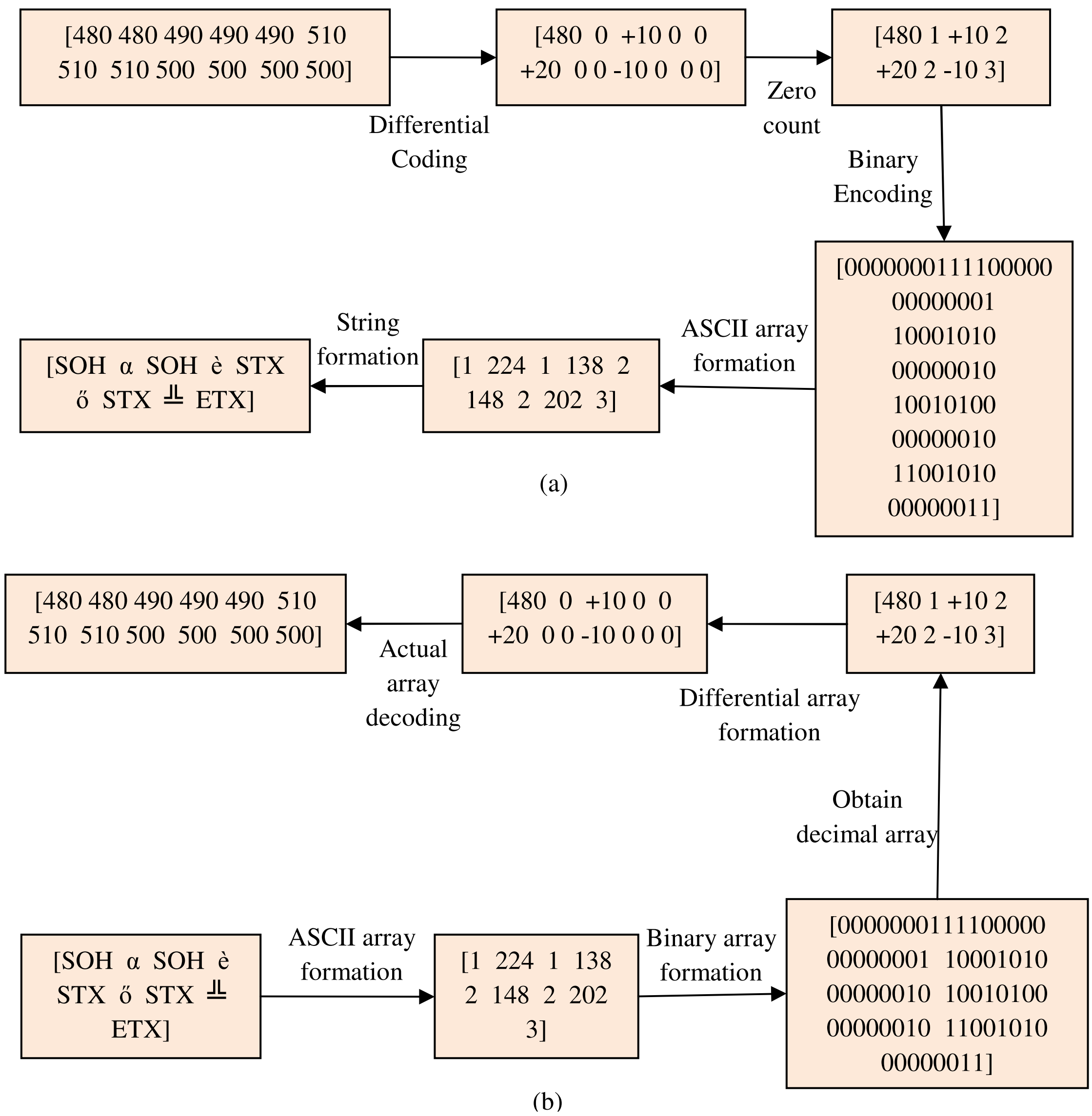


Fig. 3.6. DBEA compression scheme: (a) Encoding and (b) Decoding

3.4. Extended DBEA Compression Algorithm(E-DBEA)

Under fault conditions, there might have significant variation of system parameters. If tripping of a transmission line carrying hundreds of MW power occurs, power transfer will fall to zero in almost no time. The primary constraint of DBEA is the lower range of positive and negative difference, i.e. DBEA will fail if the difference between consecutive elements exceeds 63. Analysis of maximum deviation of differential values associated with the unit commitment information of different GS during 1st to 7th

February, 2018 is given in Table 3.3. From the Table it is clear that DBEA will fail with 9 (out of 49) such data array where the difference between the consecutive elements is more than 63. So, it becomes necessary to extend the range of DBEA so that the algorithm can compress any practical data array without any failure [94].

Table 3.3

Maximum deviation of differential values of practical unit commitment information

	BTPS	**STPS**	**KTPS**	**BkTPP I-III**	**BkTPP IV-V**	**SgTPP I**	**SgTPP II**
01/02/2018	59	19	88	60	60	17	20
02/02/2018	230	55	71	60	60	14	30
03/02/2018	48	0	66	60	60	185	30
04/02/2018	54	4	57	60	47	0	120
05/02/2018	64	55	68	60	60	0	87
06/02/2018	46	27	66	60	60	0	86
07/02/2018	74	0	57	60	46	0	0

This results in the development of E-DBEA where all elements of modified array are replaced by 14-bit binary value which increases the range of zero count and difference value. But the number of characters in the encrypted string becomes nearly twice and thereby reducing CR. The steps followed at encoding end are given in Figure 3.7 (a). Character string obtained with E-DBEA is comprised of ASCII characters only, so that the difficulty of transferring extended ASCII characters for some communication schemes can be omitted. In E-DBEA, the following features are included in the binary encoding section with all the other steps remaining unchanged [94].

i. Zero count is replaced by two characters corresponding to a 14-bit binary such that identity bit ‘0’ followed by 13-bit binary of zero count. This implies that the range will be 0-8191.
ii. Negative difference is replaced by two characters corresponding to a 14-bit binary containing identity bits ‘11’ and 12-bit binary of the corresponding element. So, the range increases to 0-4095.
iii. Similarly, positive difference is replaced by two characters which contains identity bits ‘10’ and 12-bit binary of the element. The range will be 0-4095.

The necessary pseudo code for compressing a data array to a character string is given below [93, 94].

```
Input: arr[l]
Output: str

Begin
    dif[1]=arr[1]
    Initialize i=2
    Repeat until i<(l+1)
        dif[i]= arr[i]-arr[i-1]
        i=i+1
    end
    str= NULL
    bin[16]= Binary equivalent of dif[1]
    asci[1]= Decimal equivalent of first 8 elements of bin
    asci[2]= Decimal equivalent of last 8 elements of bin
    Initialize j=2 and ascn=3
    Repeat until j<(l+1)
        Initialize cnt=0
        If dif[j]= 0
            cnt=cnt+1 until dif[j+1]≠0
            j=j+cnt
            bin[14]= Binary equivalent of cnt
            asci[ascn]= Decimal equivalent of first seven elements of bin
            ascn=ascn+1
            asci[ascn]= Decimal equivalent of last seven elements of bin
            ascn=ascn+1
        If dif[j]< 0
            po[12]= Binary equivalent of dif[j]
            bidc[14]= Include identity elements ‘1 0’ with po
            asci[ascn]= Decimal equivalent of first seven elements of bidc
            ascn=ascn+1
```

```
                    asci[ascn]= Decimal equivalent of last seven elements of bidc
                    ascn=ascn+1
                    j=j+1
            Otherwise
                    ne[12]= Binary equivalent of -1*dif[j]
                    bidc[14]= Include identity elements '1 1' with ne
                    asci[ascn]= Decimal equivalent of first seven elements of bidc
                    ascn=ascn+1
                    asci[ascn]= Decimal equivalent of last seven elements of bidc
                    ascn=ascn+1
                    j=j+1
            end
            Initialize k=1
            Repeat until k<ascn+1
                    ch= Character equivalent of asci[k]
                    str= Concatenate str and ch
            end
    end
End
```

Similar to DBEA, E-DBEA does not require any probability distribution information for decoding the actual data. The original information is extracted from the compressed character string by following the steps given in Figure 3.7 (b) and the necessary pseudo code is given below [93, 94].

```
Input: str
Output: arr[l]

Begin
    ls= Length(str)
    Initialize i=1
    Repeat until i< ls+1
            asci[i]= ASCII value of ith character of str
    end
```

```
        bfe[16]= Combined binary equivalent of asci[1] and asci[2]
        dif[1]= Decimal equivalent of bfe
        Initialize j=3 and dfcn=2
        Repeat until j< ls+1
                bidc[16]= Combined binary equivalent of asci[j] and asci[j+1]
                If first element of bidc is 0
                        zc= Decimal equivalent of next 13 elements
                        Initialize z=1
                        Repeat until z<zc+1
                                dif[dfcn]=0
                                dfcn=dfcn+1
                                z=z+1
                        end
                Otherwise
                        If second element is 0
                                pn= Decimal equivalent of next 12 elements
                                dif[dfcn]=pn
                                dfcn=dfcn+1
                        Otherwise
                                nn= Decimal equivalent of next 12 elements
                                dif[dfcn]=-1*nn
                                dfcn=dfcn+1
                        end
                end
                j=j+2
        end
        arr[1]=dif[1]
        Initialize k=2
        Repeat until k<cn+1
                arr[k]=arr[k-1]+dif[k]
                k=k+1
        end
End
```

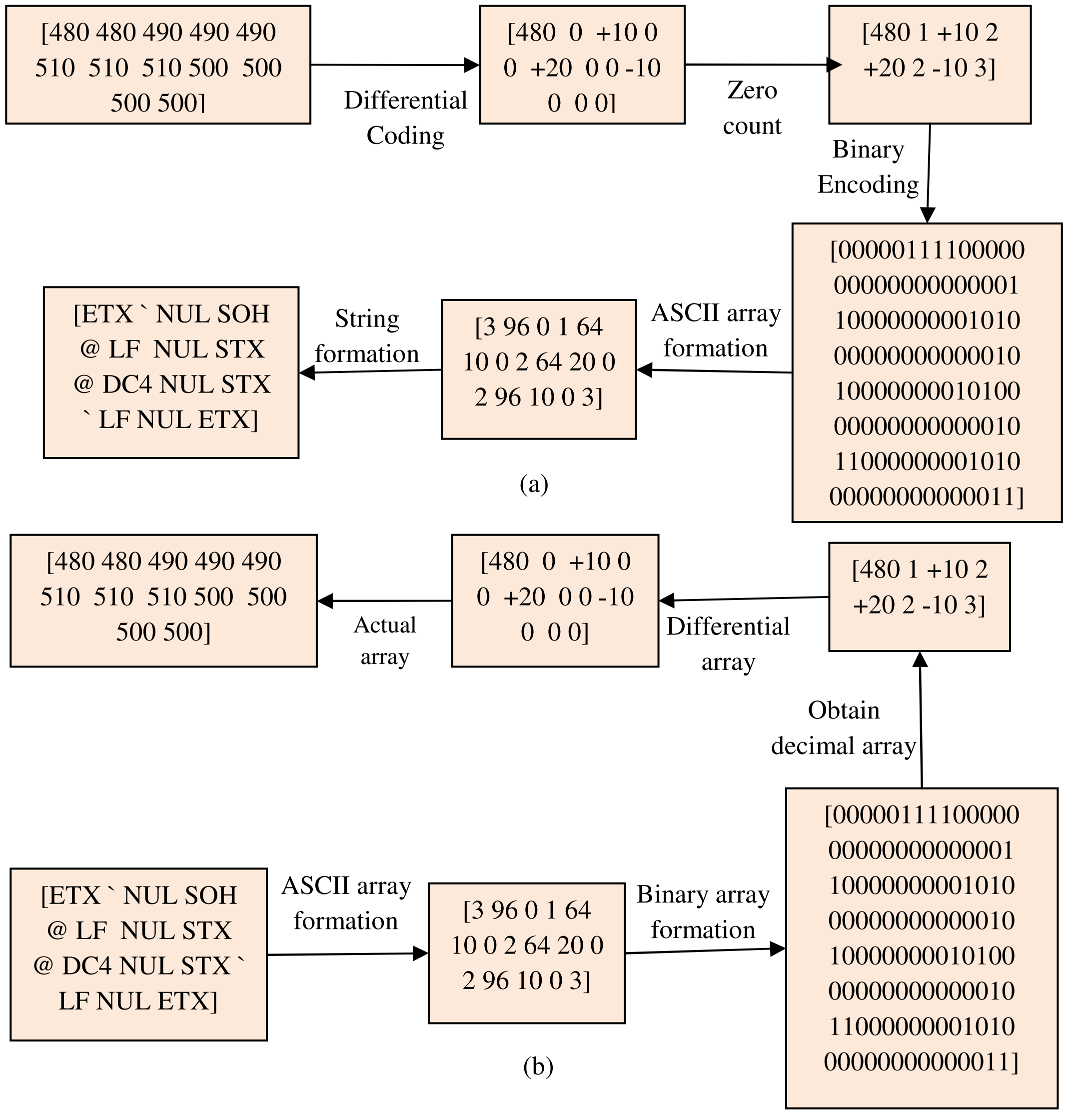


Fig. 3.7. E-DBEA compression scheme: (a) Encoding and (b) Decoding

3.5. Algorithmic complexity computation of DBEA and E-DBEA

Computation of the time complexity of any algorithm is extremely important which determines the time required to execute any algorithm. The time complexity of DBEA and E-DBEA can be computed by following the steps given below:

i. Differential Coding: For n number of elements in the data array, the time required to execute the algorithm will be a function of n and thus the time complexity of the step will be O(n).

ii. Zero count: Under worst condition, when all the elements of the array are identical, counting is done (n-1) times. So, the time complexity of the step will be O(n).

iii. Binary encoding and formation of ASCII array: Under worst condition, when all the elements of the array are different, the corresponding zero count array will have n elements. While performing binary encoding, the number of elements for DBEA and E-DBEA will be (n+1) and (2*n) respectively. For DBEA and E-DBEA, the number of elements for DBEA and E-DBEA will be same as that of the binary encoding array. This implies that the time required to execute the algorithm will be a function of n and thus the time complexity of the step will be O(n).

iv. Determination of character string: As the step involves the formation of character string corresponds to the elements of the ASCII data array. Under worst conditions, the string length for DBEA and E-DBEA will be (n+1) and (2*n) respectively. So, the time complexity of the step will be O(n).

The time complexity of the algorithm will be the maximum time complexity of the 4 steps. As the time complexity of all the steps are O(n), it is obvious that DBEA and E-DBEA will have the time complexity of O(n) which is similar to that of RLDA.

Chapter 4

DIFFERENT TYPES OF DATA TRANSFER SCHEMES FOR POWER SYSTEM OPERATIONAL DATA

4.1. Introduction

In the construction and operation of smart grids, a scalable and pervasive communication infrastructure is extremely crucial. A smart grid is characterized by the bidirectional connection of electricity and information flows to create an automated, widely distributed delivery network. It incorporates the legacy electricity grid the benefits of modern communications to deliver real-time information and enable the near-instantaneous balance of supply and demand management. Smart grid communication technologies can be grouped into five key areas: advanced components, sensing and measurement, improved interfaces and decision support, standards and groups, and integrated communications. Existing electrical utility WANs are based on a hybrid of communication technologies including wired technologies such as fibre optics, power line communication (PLC) systems, copper-wire line, and a variety of wireless technologies (i.e. data communications in cellular networks such as GSM/ GPRS/ WiMax/ WLAN and Cognitive Radio). They are designed to support some monitoring/controlling applications as SCADA, Energy Management Systems (EMS), Distribution Management Systems (DMS), Enterprise Resource Planning (ERP) systems, generation plant automation, distribution feeder automation and physical security for facilities in wide range areas with very limited bandwidth and capacity in closed networks [95].

Today's electric power grid is a complex network of networks, comprising both power and communication infrastructures, and several thousands of intelligent electronic devices (IEDs). In the smart grid environment, a communication network can be represented by a hierarchical multi-layer architecture. Classified by data rate and coverage range, this architecture comprises of Customer premises area network, i.e., Home Area Network (HAN)/ Building Area Network (BAN)/ Industrial Area Network (IAN), Neighbourhood Area Networks (NAN)/Field Area Network (FAN) and Wide Area Network (WAN). HAN/BAN/IAN applications include home automation and building automation, which are related to sending/ receiving electrical measurement data from an appliance to a controller within a customer premises. These applications do not require data to be transmitted at high frequency and all applications occur in residential/ commercial/ industrial buildings. Thus, communication requirements for HAN/BAN/IAN applications are low power consumption, low cost, simplicity, and secure communication. Communication technologies that provide data rate of up to 100 kbps with short coverage distance (up to 100 m) are generally sufficient. ZigBee, WiFi, ZWave, PLC, Bluetooth, and Ethernet are widely used to support HAN/ BAN/IAN applications [96].

In NAN/FAN applications, such as smart metering, demand response and distribution automation, data are required to transmit from a large number of customers/ field devices to a data concentrator/substation or vice versa. Therefore, these applications require communication technologies that support higher data rate (100 kbps–10 Mbps) and larger coverage distance (up to 10 km). NAN/FAN applications can be implemented over ZigBee mesh networks, WiFi mesh networks, PLC, as well as long distance wired and wireless technologies, such as WiMAX, Cellular, Digital Subscriber Line (DSL) and Coaxial Cable. For WAN applications, such as wide-area control, monitoring and protection, which require transmitting a large number of data points at much higher frequency (i.e., in a fraction of seconds) to allow stability control of a power system, communication technologies that support much higher data rate (10Mbps– 1Gbps) and provide long coverage distance (up to 100 km) are therefore required. Optical communication is commonly used as a communication medium between transmission/distribution substations and a utility control centre due to its high capacity and low latency. Cellular and WiMAX are also used due to their wide coverage range and high data throughput. Satellite communications can also be used to provide redundant communications at critical transmission/distribution substation sites as backup a communication mean in a remote location [96].

Recent developments in communication technologies have enabled reliable remote control systems, which have the capability of monitoring the real-time operating conditions and performance of electric systems. These communication technologies can be classified into four classes, i.e., PLC, Satellite Communication, Wireless Communication, and Optical Fibre Communication. Each communication technology has its own advantages and disadvantages that must be evaluated to determine the best communication technology for electric system automation. In order to avoid possible disruptions in electric systems due to unexpected failures, a highly reliable, scalable, secure, robust and cost effective communication network between substations and a remote control centre is vital. This high performance communication network should also guarantee very strict Quality of Service (QoS) requirements to prevent the possible power disturbances and outages [97].

When the communication requirements of electric system automation are considered, Internet can offer an alternative communication network to remotely control and monitor substations in a cost-effective manner with its already existing communication infrastructure. However, the Internet can’t guarantee very strict QoS requirements that the automation applications demand, since data communication in the Internet is based on ‘best effort service’ paradigm. Furthermore, when a public network like the Internet is utilized to connect the substations to a remote control centre, security concerns arise. In this context, Internet based Virtual Private Network (Internet VPN) technologies, which are transforming the Internet into a secure high speed communication network, constitute the cornerstone for providing strict QoS guarantees of electric system automation application. Internet VPN technologies offers a shared communication network backbone in which the cost of the network is spread over a large number of users while simultaneously providing the benefits of a dedicated private network. Therefore, Internet VPN technology as a high speed communication core network can be utilized to enable minimum cost and highly reliable information sharing for automation applications [97].

From the discussions, it is clear that PLC was a good choice both for HAN and NAN applications. PLC Communication (PLCC) networking technology uses existing electric power lines as its communication medium, eliminating the need to install new wires in order to implement connectivity between products plugged into the AC mains. PLCC leverages the existing power line infrastructure and provides a cost-effective approach for introducing intelligent monitoring and control to many industrial applications. Developing an efficient PLCC implementation is not without its challenges. Power lines are inherently noisy and require a robust architecture to ensure data reliability. In addition, each application and operating environment are different, requiring developers to optimize designs across a variety of factors. With the many protocol standards and modulation schemes available, developers need a flexible development platform which simplifies design, allows for optimization to environmental conditions, supports local regulations, and can be easily adjusted to conform to evolving standards [98].

In [99], a hardware setup was realised to monitor the energy consumption by using suitable sensors and Raspberry pi send energy information to the internet connected server for monitoring energy use information of individual device. Cloud storage and web hosting is used for storing real-time energy data and allows other programs to access and monitor the data. The control signal sends to raspberry pi using the same server to turn on/off the device. An integrated cloud-based data management system (CDMS) was proposed in [100], in which the asynchronous data transmission, distributed file system, and wireless network technology were used for information collection, management and sharing in large-scale egg production. The cloud-based platform can provide information technology infrastructures for different farms. The CDMS can also allocate the computing resources and storage space based on demand. CDMS consists of local farm servers and a remote cloud data centre where field data are collected from the chicken farms and is transferred to the cloud data centre asynchronously. With the improved communication infrastructure, it is possible to realize a SCADA system for managing power system operational data using cloud infrastructure.

4.2. Data transfer via power line

The conditions required for superposition to be applicable (i.e., linearity and time invariance) are not met for the majority of power line networks. A packet's signal voltage is added to the AC line voltage and causes power supply diodes to turn on and off at the packet carrier frequency, causing non-linearity. It is also assumed that, wiring capacitance dominates signal propagation effects. In real cases, it only occurs if termination or load impedance is much greater than the characteristic impedance of the wire. But, power lines are frequently loaded with impedances significantly below the characteristic impedance of the wire, under high frequency. For cases where the wire runs are less than 1/8 of a wavelength (approximately 250 meters at 100 kHz) and communication is confined to a single power phase, the presence of low impedance loads causes wire inductance to dominate. In many instances a lumped model which includes only wire inductance and low impedance loads closely approximates actual signal attenuation. Frequently the only other effect which must be considered in order to match measured values is the loss encountered when the communication signal must cross power phases. This loss, typically in the range of 5 to 25dB, is influenced by a number of variables, including distribution transformer coupling, distribution wire cross-coupling, multi-phase load impedance and circuits which are explicitly installed to reduce this loss [98].

Due to various electrical devices connected to the power supply, a considerable amount of noise is generated. Depending upon the characteristics of generated noises, they can be classified as [98]:

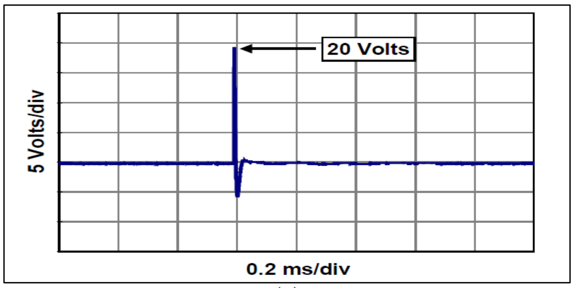
i. Impulse noise (Frequency is twice of AC line frequency): The most common impulse noise sources are triac-controlled light dimmers. These devices introduce noise as they connect the lamp to the AC line part way through each half AC cycle. When the lamp is set to medium brightness, the inrush current is at a maximum and impulses of several tens of volts are imposed on the power network. These impulses occur at twice the AC line frequency as this process is repeated in every 1/2 AC cycle. Figure 4.1 (a) shows an example of this kind of noise after the high pass filter has removed the AC power distribution frequency.

ii. Tonal noise: It can be classified as unintended and intended tonal noise.

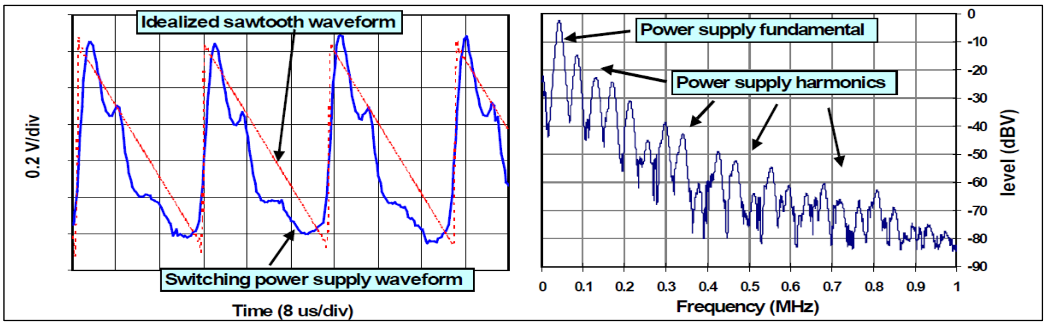
a. Unintended tonal noise: The most common sources of unintended tonal noise are switching power supplies. These supplies are present in numerous electronic devices such as personal computers and electronic fluorescent ballasts. The fundamental frequency of these supplies may be anywhere in the range from 20 kHz to >1MHz. The noise that these devices inject back onto the power mains is typically rich in harmonics of the switching frequency.

b. Intended tonal noise: Intentional tonal noise can result from devices such as power line intercoms and baby monitors. In the United States and Japan these devices generally operate at frequencies between 150 kHz to 400 kHz; injecting signals of several volts peak to peak onto the power line. A second source of intentional tonal noise results from the pickup of commercial radio broadcasts. Power wiring acts an antenna to pick up signals from these multi-thousand watt transmitters.

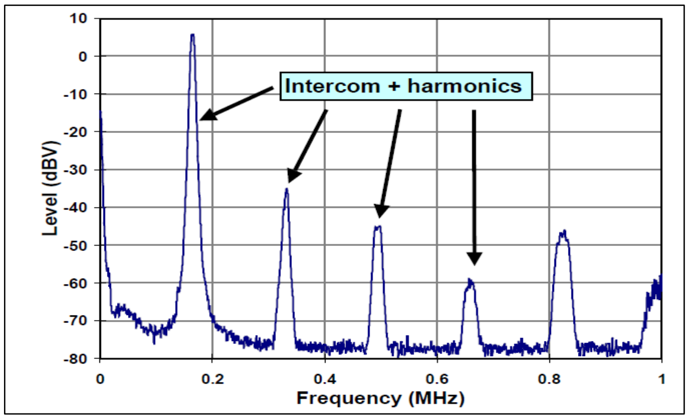
c. High frequency impulse noise: Series-wound AC motors found in different household appliances are the sources of high frequency impulse noise. Commutator arcing from these motors produces impulses at repetition rates in the several kilohertz range.



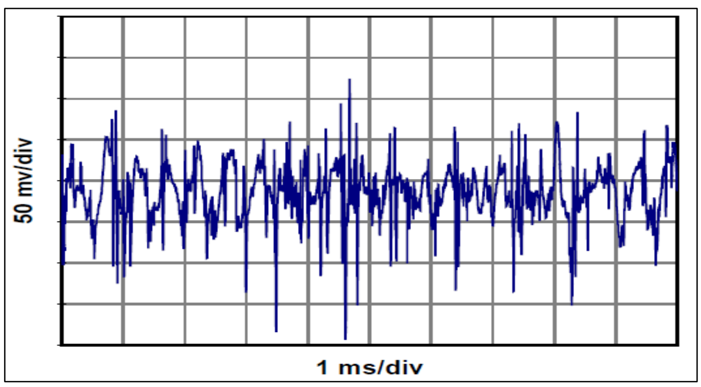
(a)



(b)



(c)



(d)

Fig. 4.1. Different type of noises: (a) Lamp dimmer impulse noise; (b) Unintended tonal noise from electric toothbrush charging stand; (c) Intentional tonal noise produced by power line intercoms and (d) High frequency impulse noise produced by vacuum cleaner

Commonly used digital communication techniques like Frequency Shift Keying (FSK) and Phase Shift Keying (PSK) suffers with serious problems while dealing with PLC communication due to power line characteristics. At high frequencies (i.e. at communication frequencies), power line characteristics vary greatly if compared with that at power frequency. The difference in characteristic impedance and load impedance at such high frequencies often results high attenuation at such communication frequencies. Apart from it, significant noise is produced by various devices connected to the system thereby degrading the performance of conventional digital communication techniques. It was found that 'Spread Spectrum Modulation Technique gives better results under power line conditions. Spread spectrum transmission is a method of signal modulation where the transmitted signal occupies a bandwidth considerably greater than the minimum necessary to send the information and some function other than the information being sent is used to increase this bandwidth. But it suffers serious problems with power line tonal interference. In the presence of common power line channel distortion, there is degradation in its performance [98].

It is clear from the previous discussions that conventional modulation techniques are not suitable for the power line environment due to the presence of high and low frequency noise in the line. With the significant increase in the use of power electronic devices, the amount of noise injected into the line will increase. To overcome the problem, a new modulation technique, Differential Code Shift Keying (DCSK) was developed and patented by Yitran Technology in 2000. It is modified version of spread spectrum technology where the data is transmitted in terms of cyclic shifting of any suitable spreading waveform. It is a highly robust communication scheme in the power line environment and many PLCC modems are based on this modulation scheme [101, 102].

4.3. Data transfer protocols in SCADA and WAMS- Role of internet for data transfer

SCADA refers to the combination of telemetry and data acquisition, which includes the collecting of the information via RTUs (Remote Terminal Unit), PLCs (Programmable Logic Controllers) and IEDs (Intelligent electronic devices), transferring it back to the central site, carrying out any necessary analysis and control and then displaying that information on a number of operator screens or displays. Master Station, Remote Terminal (RTU, PLC, IED) and the communication between them are three of the most important parts of a SCADA system. For having good communication, there must be a suitable communication protocol. Some frequently used SCADA protocols are discussed below [103].

i. Modbus RTU: It is a serial communications protocol published by Modicon in 1979 for use with its programmable logic controllers (PLCs). The reasons behind the extensive use of Modbus over other communications protocols are openly published and royalty-free, relatively easier industrial network to deploy and moving raw bits or words without placing many restrictions on vendors. It is often used to connect a supervisory computer with a remote terminal unit (RTU) in supervisory control and data acquisition (SCADA) systems.

ii. RP-570: It is a protocol used between an RTU (Remote Terminal Unit) in a substation and the FE (Front End), which is usually the SCADA software in the Control Centre. RP570 was developed by ABB in beginning of 1990 which comprises of the application layer, link layer and physical layer.

iii. Profibus: Profibus devices communicate using the standardized PROFIBUS DP (Decentralized Periphery) communication profile which defines the rules governing communication. It is based on the master/ slave concept, where a master (active communication peer) polls the associated slaves (passive communication peers) cyclically. When polled, a slave will react by sending a response frame to the polling master. A request frame comprises of output data (set point speed of a drive) and the associated response frame contains the input data (latest measured value from a sensor). In one bus cycle, the master polls, e.g., exchanges I/O data with, all associated slaves. This polling cycle is repeated as fast as possible. In parallel with this type of communication, which is described as cyclic and supports the regular exchange of input and output data between a master and its slaves, parameter data, e.g., device settings, can also be transmitted via PROFIBUS. This action is initiated by the master (typically under user program control) between I/O cycles to read and/or write slave parameter data. This type of communication is referred to as acyclic communication.

iv. Conitel: It is an asynchronous communications protocol used SCADA systems. Message blocks of this protocol are composed of 31 bits plus message synchronization "start bit" at the beginning of the first message block and an End of Message (EOM) bit at the end of each block. It may be used either in point-to-point or multi-drop configuration. Either half or full-duplex operation is possible with the protocol. Communications security is provided by a 5-bit Bose-Chaudhuri cyclic code which is included with each message block. All communication exchanges in this protocol are initiated by the host. The remote cannot initiate any exchange with the host nor can the remote directly address or communicate with another remote. The remote will return a response to the host for all valid messages sent by the host and addressed to the remote.

v. IEC 60870: It is the collection of standards produced by the IEC (International Electrotechnical Commission). It was created to provide an open standard for the transmission of SCADA telemetry control and information. It provides a detailed functional description for telecontrol equipment and systems for controlling geographically widespread processes specifically for SCADA systems. The standard is intended for application in the electrical industries, and has data objects that are specifically intended for such applications. It is also applicable to general SCADA applications in any industry. T101 or IEC 60870-5-101 (IEC101) is an international standard prepared by TC57 for power system monitoring, control and associated communications. This is compatible with IEC 60870-5-1 to IEC 60870-5-5 standards and uses standard asynchronous serial tele-control channel interface between DTE and DCE. The standard is suitable for multiple configurations like point-to-point, star, multi-dropped etc.

vi. DNP3: It was specifically developed to facilitate communications between various types of data acquisition and control systems. It plays a crucial role in SCADA systems. It is used for the communications between SCADA Master Stations or Control Centres, RTUs, and IEDs. It supports multiple-slave, peer-to-peer and multiple-master communications as well as operational modes of polled and quiescent operation.

Of the six SCADA protocols, T101 and DNP3 are the most commonly used protocols in SCADA systems. These communication protocols are standardized and recognized by all major SCADA vendors. Many of these protocols are now improved and contain extensions to operate over TCP/IP. DNP3 is widely used in North America, South America, South Africa, Asia and Australia, while IEC 60870-5-101 or T101 is strongly supported in the Europe. Table 4.1 shows the comparison of both protocols, the DNP3 and the T101 [103].

Table 4.1

Comparison of DNP3 and T101 protocols

Fields	**DNP3**	**T101**
Organization	DNP users group	IEC TC 57 WG 03
Standard	Open industry specification	IEC standard
Dominant market	North America	Europe
Architecture	4-layer architecture. Also supports 7-layer TCP/IP or UDP/UP	4-layer EPA architecture
Main coverage	Application layer (Services and Protocols)	Application layer (Services and Protocols)
Device Addressing	Link contains 16- bit source and destination address. Application layer does not contain address. 32 bit point addresses of each data type per device.	Link address could be 0, 1 or 2 bytes. Unbalanced link contains slave's address. Balanced link is point to point so link address is optional (may be included for security).
Cyclic transmission	Available but interval can't be adjusted remotely.	Eliminates static data poll message from master. Interrupted by event triggered communication request.
Open for other encoding solutions	Open for other encoding solutions like XML.	No

Introduction of the Wide-Area Monitoring Systems (WAMS) and Wide Area Monitoring, Protection and Control (WAMPAC) systems to electric power network require certain responses and adaptations from supporting information and communications technology (ICT) systems. WAMS comprises of Phasor Measurement Units (PMU) and Phasor Data Concentrators (PDC) that collect data from multiple sources (from PMUs and other PDCs). PMUs are most generally located in substations, while the location of PDCs depends on the chosen architecture of the system. It can be either centralised (one PDC covers all PMUs), or hierarchical - with more PDCs, which are subordinate to PDCs at higher levels. The PMU is used for synchronous measurement of AC voltages and currents with a common time reference. This time reference is the coordinated universal time (UTC), normally synchronized to the atomic clock time precision using a satellite global positioning system (GPS). PDC captures phasors in real time along with other data from PMUs. For data processing, time alignment of data from multiple data sources is required, which is carried out by the PDC. Adequate responses and adaptations of the ICT systems are required for introducing WAMS into the electric power systems. Different standardised formats and protocols can be used for data transfer between PMU, PDC and associated applications. It includes IEEE C37.118.2, IEEE 1344, BPA PDC Stream, IEC 61850-8-1 GOOSE (Generic Object Oriented Substation Event) based on standard IEC 61850-90-5 and IEC 61850-9-2 sampled values, also based on IEC 61850-90-5 [104].

The nature of data flow is the basic characteristic of the communication between PMU and PDC. The data sent by the PMU are needed by multiple PDC devices. From the point of view of data flow, there is a flow of data in IP packets in the direction PMU to PDC, where TCP or UDP protocol can be used for transport. In view of the nature of the traffic, where the same information from one source (PMU) can be used by several destinations (PDC devices), two mechanisms of the IP packet flow can be used for data transmission. In unicast traffic, IP packet flow is towards a single destination. But in multicast traffic, IP packet is sent to multiple destinations. WAN networks are responsible for traffic flows between PMU and PDC devices in ICT network. IP/MPLS technology is the best for use in WAMS data flows due to the maturity of technology, strong support of IETF, IEEE and ITU-T standards, widespread use and the ability to provide a wide range of network services with a single technology [104]. From the discussions, it is clear that modern day SCADA relies heavily on internet connectivity for transferring enormous volume of information between the field devices present in different substations and control centre. With the system modernization, more PMUs are being installed to extract the advantages of WAMS. This in turn requires Ethernet connectivity for developing the WAN which connects different PMUs with PDC. This implies that the role of the internet is inevitable for transferring data in modern power system.

4.4. Cloud prospective in India

In cloud computing, resources such as computing power & infrastructure, application platforms, and business processes are provided through the internet as general utilities to users in an on demand fashion. A consumer can access and use these resources and services from anywhere and anytime through internet connection. The end user may not be aware of the equipment that is being used to provide him this service. According to the National Institute of Standards and Technology (NIST, USA), US Department of Commerce, Cloud computing is defined as “a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources(e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction”. In a cloud computing environment, the traditional role of service provider can be divided as [105]:

i. The infrastructure who manage platforms in the internet cloud and lease resources accordingly to a usage-based pricing model.
ii. Service providers who rent resources from one or many infrastructure providers to serve the end users.
iii. Service providers who offer cloud services

Software as a service (SaaS), Platform as a Service (PaaS) and Infrastructure as a Service (IaaS) are among the different service models in cloud computing. SaaS is a software distribution model through which a consumer can use the provider’s applications (software) running on a cloud infrastructure. In PaaS, the service provides the consumer hardware and software infrastructure to deploy onto the cloud infrastructure consumer-created or acquired applications and tools supported by the provider. But IaaS provides the consumer ability to provision processing, storage, networks, and other fundamental computing resources where the consumer is able to deploy and run one’s own software, which can include operating systems and applications. There are also other service models like Data as a Service (DaaS), Identity and Policy Management as a Service (IPMaaS), Network as a Service (NaaS), Video as a Service (VaaS) or Hardware as a service (HaaS). A cloud system can be operated in one of the following four deployment models [105].

i. Public cloud: The cloud infrastructure is made available to the general public or a large industry group and is owned by a third party selling cloud services.
ii. Private cloud: The cloud infrastructure is operated solely for an organization managed privately.
iii. Community cloud: The cloud infrastructure is shared by several organizations and supports a specific community that has shared concerns (e.g., mission, security requirements, policy, and compliance considerations).
iv. Hybrid cloud: The cloud infrastructure is a composition of two or more cloud environments (private, community or public) that remain unique entities, but are bound together by standardized or proprietary technology enabling data and application portability.

In India, Cloud Computing offers huge potential for industries to grow and is opening up new windows of opportunities. Verticals such as retail, railways, manufacturing, banking, education and healthcare have started switching their on-premise applications to cloud services for optimised reach and performance as well as elasticity and scalability. As per Gartner report, 53 percent of organizations in India indicated they are using cloud services, with another 43 percent indicated their plans to begin using cloud services in the subsequent year. The overall cloud computing market reached $1.08 billion by the end of 2015. IT/ITeS, Telecom, BFSI, Manufacturing and Government sectors contributed largest to the cloud market in India, with nearly 78% of the total market. Few market predictions for cloud computing in India in the upcoming years are as follows [105]:

i. According to research firm Gartner, the demand for Internet of Things (IoT) devices will drive up to 1.6 billion units next year (39% percent rise from this year) due to the accelerated penetration of smart city technologies. It was also predicted that smart homes will take 21% of total demand for IoT devices in 2016 and will increase rapidly in the next five years. This will proportionally boost up the cloud services.
ii. The public cloud service market in India will grow from $838 Million in 2015 to $1.9 billion by 2018.

iii. It is also expected by Gartner that the SaaS market, in particular, will roughly double in value, between 2014 and 2020, when it will be worth $1.2 billion. By 2018, Indian spending will reach $735 million for SaaS (from $249 million in 2014), $295 million for IaaS (from $77 million), and demonstrate strong gains in sub-sectors like PaaS and Business-Process-as-a-Service as well.

iv. Social, mobility, analytics and cloud (SMAC) are collectively expected to offer a US $1 trillion opportunity in 2016. The cloud represents the largest opportunity under SMAC, increasing at a compound annual growth rate (CAGR) of approximately 30 per cent to around US $650-700 billion by 2020.

Government of India (GoI) is keen to explore a cloud based application and data access model to revolutionize its e-governance initiative. The focus of e-governance is to reduce corruption and ensure the government schemes are reaching people living in rural areas of the country. Further, e-governance services ensure quicker service delivery and eliminate the involvement of middlemen who tend to capitalize on loopholes for quick money by means of exploiting people. To build the backbone of national e-governance plan, the Department of Information Technology intends to establish a national cloud-based network, which would link all the data centres of the 29 states and 7 Union Territories (UTs) of the country. Not only would this enable the states and UTs to get their own private cloud, this plan would also assist in the timely implementation and delivery of different government to citizen and government to business services via the cloud. Once implemented, the infrastructure would help the government to share critical information across departments via common IT resources [105].

The current National e-Governance Plan (NeGP) has its own strengths as the first well organized plan of the Govt. of India for both Central and State level e-Governance projects. Twenty-seven mission mode projects (MMPs) of NeGP had seen their partial successes. However, many additions such as stakeholders needs analysis, project planning and management, process reforms and reengineering could be identified, especially in the context of technology developments such as the ubiquitous mobile phone penetration calling for mobile applications, new technologies such as Service-Oriented Architecture (SOA), grid, cloud, big data analytics, and enterprise architecture techniques for deployment into e-Governance. The various sectors where cloud adaptation was implemented are discussed below [105].

i. Banking: Indian Banking Community Cloud (IBCC) is the first Community Cloud initiative for the banking industry in the country. The theme has been to "Optimize costs while maintaining desired levels of efficiencies and security". Institute for Development and Research in Banking Technology (IDRBT) is working with the public sector and private sector banks to provide Infrastructure as a Service for non-customer facing and less critical applications.
ii. Manufacturing: Human machine interface (HMI) is an area where companies such as Jindal Steel have adopted the cloud model for their HMI applications to quickly recover their ROI. HMI refers to interfacing IT systems like ERP with manufacturing executing systems (MES) and plant automation.
iii. Telecom: The Operation Support System (OSS) and Business Support System (BSS) in Indian telecom industry are now widely keen on deploying SaaS to reduce CapEx and OpEx.
iv. Railways: Strategies have been rolled out to use the cloud for GIS management in railways, for e-ticket bookings and for automated surveillances of railway premises and storage of video logs in cloud data centres.
v. Health: Cloud Computing Innovation council of India has proposed a layout for the systematic adoption of cloud services in the Indian health sector, known as e-Health vision. e-Health vision aims to incorporate Health Information Exchange (HIE) mechanisms for successful deployment of cloud. An electronic health information exchange (HIE) allows stakeholders associated with health data to appropriately access and securely share a patient's vital medical information electronically.
vi. Education: Megh-Sikshak is a cloud-based learning management system, which is evolved from the objective of converting the traditional model of e-Learning system (eSikshak) to a SaaS model. Megh-Sikshak offers multi-lingual e-Learning services, leveraged by cloud computing capabilities and demonstrates the new model of a SaaS based e-Learning system. IIT Delhi, IGNOU and other universities have deployed their own cloud environments.

vii. Right To Information (RTI): Government's initiative to digitize its database and make more and more information available to the public domain calls for an indispensable need of adoption of cloud services in the domain of Right To Information for efficient performances.

viii. Meghraj: Department of Electronics and IT (DeitY) of the Government of India has initiated an extensive project termed as 'GI Cloud'. The 'GI Cloud' Meghraj is the Government of India's cloud computing environment that will be used by government departments and agencies at the centre and states following a set of common protocols, guidelines and standards issued by the Government of India.

ix. National eGov App Store: The eGov App Store will include the setting up of a common platform to host and run applications (developed by government agencies or private players) at National Cloud, which are easily customisable and configurable for reuse by various government agencies or departments at the central and state levels without investing effort in the development of such applications. The eGov App Store hosted on the National Cloud will be termed as the 'National eGov App Store'.

Selection of cloud will be entirely based on system requirement and cost. In Indian scenario, there are several vendors providing the cloud. There are some big companies like Microsoft, Google, IBM, Amazon, etc. which shares the maximum market. Beside that there are few comparatively small vendors like Salesforce, VMware, Oracle/Sun, etc. Of the different vendors, Microsoft is the most preferred Cloud Computing technology provider which is followed by Amazon Web Services (AWS). According to RightScale's survey of 997 respondents across multiple industries and company sizes, AWS and Microsoft Azure are on the top. They are followed by Google Cloud and IBM Cloud. The enterprise scorecard of AWS, Azure, Google and IBM after private-public cloud jousting is given in Table 4.2 [106].

Table 4.2

Enterprise scorecard of AWS, Azure, Google and IBM

Area	AWS	Azure	Google	IBM
% Adaptation	68%	58%	19%	15%
YoY growth in adaptation	15%	35%	26%	50%
% Adaptation in Beginners	47%	49%	18%	14%
% with Footprints > 50 VMs	58%	44%	17%	14%
YoY growth in Footprints > 50 VMs	14%	38%	42%	56%

4.5. IoT based data transfer and management scheme using cloud server

Internet of things (IoT) is an extremely popular domain of investigation in the recent times. With the significant rise in the internet facilities, it becomes easier to connect various devices with internet and use the information from some distant places. The rising interest in IoT can be understood by its use in different sectors like healthcare, automotive, education and sports. IoT data have distinctive characteristics that make traditional relational-based database management an obsolete solution. A massive volume of heterogeneous, streaming and geographically-dispersed real-time data will be created by millions of diverse devices periodically sending observations about certain monitored phenomena or reporting the occurrence of certain or abnormal events of interest [107]. This implies that the management of such enormous volume of information is a tedious job. In many literatures, IoT based DAS were realized where data acquired through suitable sensors were stored in the cloud environment. Storing of different data in the cloud will help with system to store, manage, and process data so that accurate decisions can be taken on time so as to increase the system performance.

In 2011, the number of interconnected devices on the planet overtook the actual number of people. Currently there are 9 billion interconnected devices and it is expected to reach 24 billion devices by 2020. According to the GSMA, this amounts to $1.3 trillion revenue opportunities for mobile network operators alone spanning vertical segments such as health, automotive, utilities and consumer electronics. Storage, ownership and expiry of the data become critical issues in IoT. The internet consumes up to 5% of the total energy generated today and with these types of demands, it is sure to go up even further. Hence data centres which run on harvested energy and which are centralized will ensure energy efficiency as well as reliability. The data have to be stored and used intelligently for smart monitoring and actuation. The vision of IoT can be either ‘Internet centric’ or ‘Thing centric’. The Internet centric architecture will involve internet services being the main focus while data is contributed by the objects. In the object centric architecture, the smart objects take the centre stage [108].

In order to realize the full potential of cloud computing as well as ubiquitous Sensing, a combined framework with a cloud at the centre appears to be most viable. It not only gives the flexibility of dividing associated costs in the most logical manner but also provides high scalability. Sensing service providers can join the network and offer their data using a storage cloud; analytic tool developers can provide their software tools; artificial intelligence experts can provide their data mining and machine learning tools useful in converting information to knowledge and finally a computer graphics designer can offer a variety of visualization tools. The cloud computing can offer these services as Infrastructures, Platforms or Software where the full potential of human creativity can be tapped using them as services. An integrated IoT and Cloud computing applications enables the creation of smart environments such as Smart Cities which must be capable to combine services offered by multiple stakeholders and scale to support a large number of users in a reliable and decentralized manner. They need to be able operate in both wired and wireless network environments and deal with constraints such as access devices or data sources with limited power and unreliable connectivity. The Cloud application platforms must be enhanced to support the rapid creation of applications by providing domain specific programming tools and environments. Beside that the platform must be upgraded for seamless execution of applications harnessing capabilities of multiple dynamic and heterogeneous resources to meet quality of service requirements of diverse users [108].

The role of cloud based IoT applications in smart city technologies was expected to increase significantly in India [105]. From the discussions, it is clear that data storage and management is one of the major challenges for implementing cloud based IoT. Data compression can be a good solution to overcome the constraints associated with data storage and data transfer. While approaching towards the formation of smart grid, IoT can play a crucial role in acquiring power system operational data and storing the information in the cloud. If real time monitoring is not desirable, a periodic compression scheme developed in [109] can be adopted instead. In the compressed DAS scheme [109], data collected by a temperature sensor over a finite time interval are compressed by Modified DBEA (M-DBEA) compression scheme and transmitted by employing serial communication. At the receiving end, the compressed information is decoded and is displayed. This method reduces the volume of information considerably and a high CR obtained for test data sets. While realising a cloud based IoT for power system applications, the temperature sensor will be replaced by suitable sensing units and internet connectivity must be incorporated to transmit the compressed information to a cloud server. As the proposed compression algorithms, DBEA and E-DBEA are computationally very simple; it is possible to implement the lossless schemes at low level microcontroller level. Though DBEA or E-DBEA do not have data management facility, they can be useful for reducing the memory requirements while realizing cloud based IoT.

Chapter 5

GENERATION SCHEDULING AND ECONOMIC LOAD DESPATCH

5.1. Introduction

In power generation and delivery, operational economics can be subdivided in two categories as economic despatch and minimum loss delivery. Economic despatch deals with the minimum cost of power generation and determines the power output of each plant for minimizing the fuel required to deliver a given load [35]. Each transmission line has a power transmission limit due to thermal, voltage or stability considerations. Traditionally a transmission line does not exceeds the limits during economically despatched generation. But, with the deregulation of the electric utility industry the transmission system is becoming increasingly constrained. This problem can be solved by optimizing the generation while enforcing the transmission lines is to combine economic dispatch with the power flow and is termed as optimal power flow (OPF). Linear programming (LP) approach is one of the most commonly used methods for solving the OPF problem [110]. As generation scheduling is the economic dispatch of power by allocating generation among different GS [111], the focus of this chapter will be primarily on economic dispatch.

The input output characteristics for any thermal unit or units that comprise the plant can be obtained from the operating data. The input can be in kilo calories per hour and the output may be in kilowatts or preferably in megawatts. In practice, the input output characteristics may not be a smooth curve and may require interpolation. Incremental fuel rate or heat rate defined as the ratio of differential change in input with respect to differential change in output can be obtained from the input output characteristics.

When the incremental fuel rate (in K-caI/KWhr) is multiplied by cost-of the fuel (in Rs per K-cal), incremental fuel cost (in Rs./KWhr) is obtained. The incremental fuel cost characteristics of a thermal unit is given in Figure 5.1 [111].

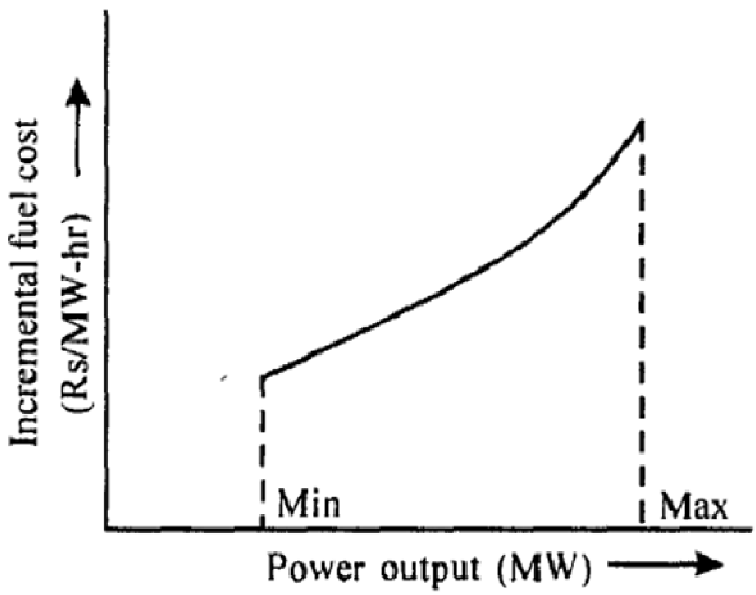


Fig. 5.1. Variation of incremental fuel cost with power output

Economic Load Despatch (ELD) problem is a challenging problem for economic operation of the system. It involves the optimum distribution of load among different GS so as to have a minimum transmission loss and low generation cost. In conventional algorithm for solving ELD problem, there is an increased complexity with the increase in the number of available GS. Beside that, there is a significant increase in computational time while dealing with large number of GS. Different stochastic techniques have evolved for solving ELD problem, but they often require a large number of iterations to obtain optimum value. Particle Swarm Optimization (PSO) is a simple and easy technique for solving linear and non-linear problems. There are several works which involve PSO based algorithms for solving ELD problem. In some literatures, Artificial Neural Network (ANN) is employed to allocate optimum generation of the GS.

As the system load demand varies with time, it becomes necessary for an electric utility to decide the ON-OFF sequence of the GS in advance. Unit Commitment (UC) Problem is the problem of deciding the selection of units required for meeting a given load demand. This problem is important for thermal GS only due to reduced operational cost and start-up times of hydel GS. Priority ordering is a simple, sub-optimal approach for solving UC problem, where the most efficient unit is loaded first to deliver the load. If load demand is high, the less efficient units are introduced subsequently. Though economic despatch and unit commitment appears to be similar, they are different. Economic dispatch distributes system load economically between different units which are available during that time. Unit commitment in contrast, selects the best combination of the units which must be available for supplying the forecasted load demand. Dynamic programming method is a good choice for solving UC problem [35, 36].

5.2. Economic Load Despatch (ELD) Problem

ELD is a constraint based optimization problem in power systems that have the objective of dividing the total power demand among the online participating generators economically while satisfying the essential constraints [112]. The objective of ELD problem is to minimize the total cost of power generation of various GS while satisfying the loads and losses in the transmission link [36]. The parameters taken into account for any ELD problem are load demand, transmission losses and generation cost coefficients. There are many traditional optimization methods to solve ELD problem which includes lambda iteration, gradient method, base point, participation factor method, Newton's method, Linear programming, and quadratic programming [113].

Generator operating cost (in Rs/hr) of i^{th} unit is often expressed by a 2^{nd} order polynomial of function P_{Gi} as given where a_i, b_i and c_i are unit constants [36].

$$C_i = \frac{1}{2} a_i P_{Gi}^2 + b_i P_{Gi} + c_i \tag{5.1}$$

Incremental fuel cost (IC) is the slope of the cost curve (C_i) and is expressed in Rs/MWh.

$$IC_i = a_i P_{Gi} + b_i \tag{5.2}$$

In order to meet a load demand P_D, generation must be shared among k different units so as to have minimum system operating cost C. The sharing should be such that the power demand can be met under the given equality and inequality constraints. As the generated power must meet the load demand and system loss (P_L), power balance equation of ELD problems is given as [85]:

$$P_D + P_L - \sum_{i=1}^{k} P_{Gi} = 0 \tag{5.3}$$

For a system comprising of k GS, if $P_G(1Xk)$ and B (kXk) is GS generation matrix and transmission loss matrix respectively, then P_L will be a function of the generation of individual GS and will be expressed as [36]:

$$P_L = P_G^T \times B \times P_G \tag{5.4}$$

The generation of each GS must be varied within its minimum and maximum limits of power generation. If P_{Gi}^{max} and P_{Gi}^{min} be the maximum and minimum limit of power generation of i^{th} GS, then the inequality constraint of each GS will be given as [85]:

$$P_{Gi}^{min} \leq P_{Gi} \leq P_{Gi}^{max} \tag{5.5}$$

For this constraint based optimization problem, Lagrangian multiplier is used and the augmented cost function is given by [85, 113]:

$$L = \sum_{i=1}^{k} C_i + \lambda \times (P_D + P_L - \sum_{i=1}^{k} P_{Gi}) \tag{5.6}$$

where λ is the Lagrangian multiplier.

For optimum real power despatch, the relation given in equation 5.7 is valid for all $i \in (1,k)$ [36].

$$\frac{\partial L}{\partial P_{Gi}} = \frac{dC_i}{dP_{Gi}} - \lambda - \lambda \frac{\partial P_L}{\partial P_{Gi}} = 0 \tag{5.7}$$

Simplifying the equation 5.7, we have

$$\frac{dC_i}{dP_{Gi}} \times L_i = \lambda \qquad (5.8)$$

where L_i is the penalty factor of i^{th} plant expressed mathematically as $1 \Big/ \left(1 - \frac{\partial P_L}{\partial P_{Gi}}\right)$.

This λ is approximately equal to the cost (in Rs. / hr) to increase the total load to be delivered by 1 MW. Penalty factor is a measure of sensitivity of any transmission system loss due to changes in P_{Gi} alone. Generating units of a particular power plant connected to the same bus will have equal access to the transmission system. So, any change in system losses must be same for any small change in the output of any one of those units. This implies that the penalty factor of the generating units of the same power plant will be equal. From equation 5.8, it is clear that the economic loading of the units is dependent on their geographical locations. For determining the penalty factors of the plants, it thus becomes extremely necessary to determine the system transmission loss as a function of individual plant loadings [35].

5.3. Solution of ELD Problem by Lambda Iteration method

The lambda Iteration method is one of the most popular traditional techniques to solve ELD problem for minimizing the cost of a GS. Although the method encounters a significant computational burden with increasing number of GS, extremely fast convergence makes it a good choice for solving ELD problem [113]. The pseudo code for solving ELD problem by using the lambda iteration method is given below.

Input: IC_1, IC_2; B; P_D; P_{G1max}, P_{G1min}, P_{G2max}, P_{G2min}

Output: P_{G1}; P_{G2}

Begin

Determine P_{G1}^{ini} and P_{G2}^{ini} by solving $IC_1 = IC_2$ and $P_{G1}+P_{G2} = P_D$ such that equation 5.5 can be satisfied

$\lambda_{ini} = a_1 * P_{G1}^{ini} + b_1$ or $a_2 * P_{G2}^{ini} + b_2$

Determine P_L^{ini} by using equation 5.4

pb= $\sum_{i=1}^{2} P_{Gi}^{ini} - P_D - P_L^{ini}$

λ= λ_{ini}

P_{G1}= P_{G1}^{ini}

P_{G2} = P_{G2}^{ini}

Repeat until |pb|<0.001 and equation 5.5 is satisfied

Initialize cnt=1

If pb<0

λ=λ*(1+(0.4/2^{cnt}))

cnt=cnt+1

Otherwise

λ=λ*(1-(0.4/2^{cnt}))

cnt=cnt+1

end

Determine $L_1 = \frac{1}{1-\frac{\partial P_L}{\partial P_{G1}}}$ and $L_2 = \frac{1}{1-\frac{\partial P_L}{\partial P_{G2}}}$

Using equation 5.8, determine updated P_{G1} and P_{G2}

pb= $\sum_{i=1}^{2} P_{Gi} - P_D - P_L^{ini}$

end

End

5.4. Solution of ELD Problem by Iterative Weight Particle Swarm Optimization (IW-PSO)

PSO is a nature inspired optimization technique evolved in 1995 for solving linear and non-linear functions which involve a population of candidates (or particles) termed as a swarm. Kennedy and Eberhart had proposed PSO, which is inspired by the collective behaviour of nature e.g. bird flocking. Each particle of the swarm is placed randomly in a search space and assigned them with random velocities. Each particle will move iteratively in the search space until all of them converge towards the global best position. In conventional PSO (C-PSO), the velocity and position of the particles are updated iteratively by the velocity and position modification rule given by equation 5.9 and 5.10 respectively [85, 114].

$$v_{ij}^{t+1} = v_{ij}^{t} + c_1 \times r_1 \times (P_{ij}^{best} - X_{ij}^{t}) + c_2 \times r_2 \times (G_i^{best} - X_{ij}^{t}) \quad (5.9)$$

$$X_{ij}^{t+1} = X_{ij}^{t} + v_{ij}^{t+1} \quad (5.10)$$

where t gives the number of iterations; i indicates the number of variables; j gives the number of particles in the swarm; c_1, c_2 being the acceleration constants; r_1, r_2 are the random numbers between 0 and 1; P_{ij}^{best} is the personal best position of j^{th} particle of i^{th} variable and G_i^{best} is the previous global best position of i^{th} variable.

The pseudo code for obtaining the optimal solution of an optimization problem by using PSO is given below.

Input: Objective function; t; Number of particles in the swarm (N); c_1, c_2; Condition of convergence (coc)

Output: Optimum solution of objective function

```
Begin
    Initialize particle positions and velocities
    Compute personal and global best of the function according to initial positions
    Check for coc
    Repeat until coc is achieved
        Update position and velocity using equation 5.9 and 5.10
        Compute fitness value
        Compute personal and global best
        If coc is achieved
            break
        Else
            continue
        end
    end
    Optimum solution of PSO will be the G_i^best
End
```

In IW-PSO, a new parameter inertia weight (w) is introduced in the velocity modification rule given in equation 5.9 with position modification rule remains unaltered [85].

$$v_{ij}^{t+1} = w \times v_{ij}^{t} + c_1 \times r_1 \times (P_{ij}^{best} - X_{ij}^{t}) + c_2 \times r_2 \times (G_i^{best} - X_{ij}^{t}) \tag{5.11}$$

The inertia weight may be fixed or can vary with iteration number. Normally, linearly decreasing inertia weight is broadly used where w is varied linearly between w_{max} and w_{min} using the given relation [85].

$$w = w_{max} - \frac{t \times (w_{max} - w_{min})}{It_{max}} \tag{5.12}$$

where t is the current iteration and It_{max} is the maximum number of iterations to be performed. Usually, w_{max} and w_{min} is considered to be 0.9 and 0.4 respectively.

Due to the simplicity and ease implementation of PSO, it was utilized for solving ELD problems in different literatures. The fitness function of ELD problem is defined to minimize the sum of generation cost function and power balance equation as given in equation 5.13. The pseudo code for solving ELD problem using IW-PSO is discussed below [85].

$$F = \sum_{i=1}^{n} (a_i P_i^2 + b_i P_i + c_i) + K \times (P_D + P_L - \sum_{i=1}^{k} P_{Gi}) \tag{5.13}$$

where K being the penalty factor

Input: a_1, b_1, c_1, a_2, b_2, c_2; B; P_D; P_{G1max}, P_{G1min}, P_{G2max}, P_{G2min}

Output: P_{G1}; P_{G2}

Begin

The various parameters of IW-PSO i.e. c_1, c_2, w_{max}, w_{min}, population size (P) and It_{max} are chosen

$P_{Gi}$is initialized with random positions between P_{Gi}^{max} and P_{Gi}^{min}

Initialize velocity of all particles randomly between $0.5 \times P_{Gi}^{max}$ and $-0.5 \times P_{Gi}^{min}$

By selecting the suitable value of K, F is computed for all particles

For each particle, the calculated value of F (equation 5.13) will be its initial personal best

The particle reaching having minimum F will be initial global best

Repeat until the difference of F between consecutive iteration is within a tolerance limit

Update w by equation 5.12

The position and velocity of each particle is updated by equation 5.10 and 5.11 respectively

If the computed positions and velocity is not within [$P_{Gi}^{\max}$, $P_{Gi}^{\min}$] and [$0.5\times P_{Gi}^{\min}$, $-0.5\times P_{Gi}^{\min}$] respectively

If $P_{Gi} < P_{Gi}^{\min}$

$P_{Gi} \equiv P_{Gi}^{\min}$

Else if $P_{Gi} > P_{Gi}^{\max}$

$P_{Gi} \equiv P_{Gi}^{\max}$

end

If $v_i < -0.5\times P_{Gi}^{\min}$

$v_i \equiv -0.5\times P_{Gi}^{\min}$

Else if, $v_i > 0.5\times P_{Gi}^{\max}$

$v_i \equiv 0.5\times P_{Gi}^{\max}$

end

end

In current iteration, if F is minimum for any particle, personal best is updated

The optimum of personal best of all particles will now be global best

Compute F and check for convergence

end

End

5.5. Solution of ELD Problem by Artificial Neural Network (ANN)

ANN is a modern computational tool which is useful for solving many complex real world problems. They have remarkable information processing having high parallelism, fault and noise tolerance, and learning and generalization capabilities. The advantage of using a neural network approach is that all the behaviour can be represented within a unified environment of a neural network and that the network is built directly from experimental data using the self organising capabilities of the neural network. It is inspired by the structure and operation of the brain. They are massively parallel systems, made up of a large number of highly interconnected, simple processing units [115, 116].

A neural network is described as the network connectivity which defines the inter connection parameters, such as threshold, transfer functions and the weights associated with its inputs. The artificial neuron is designed to mimic the first order characteristics of the biological neurons. A neuron is composed of a processing body transmitting axons and receiving synapses and dendrites. A nerve cell can be modelled as an artificial neuron (computational unit) comprising of receiving connections (dendrites), receiving sites (synopses), a processing element (cell body) and transmitting connections (axons). The cell receives the input stimuli through numerous dendrites, which are summed and relayed to the cell body. The cell transmits an impulse through the axon to other cells, when this input exceeds a certain threshold value [115]. The structure of a neuron is given in Figure 5.2

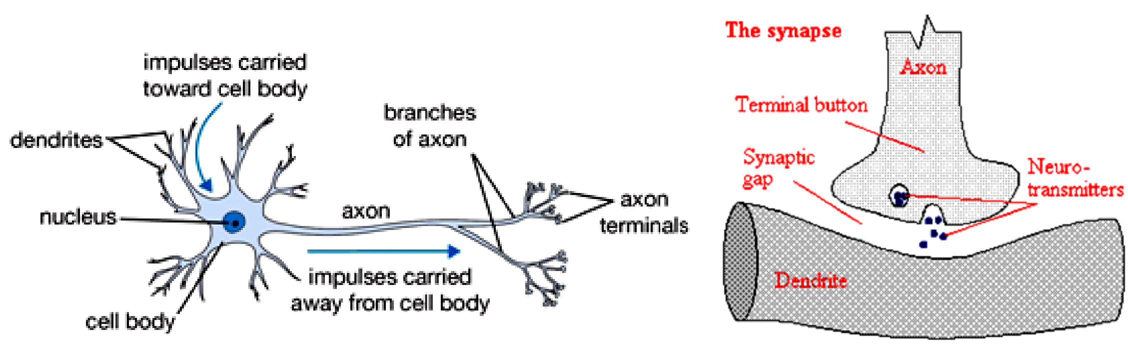


Fig.5.2. Structure of a biological neuron

Activation or activity level of a neuron is an internal state which is a function of the inputs received by it. This activation can be then sent to several other neurons in the form of a signal. Generally a neuron can send only one signal at a time, which can be broadcasted to several other neurons. If the output of neurons X_1, X_2 and X_3 be x_1, x_2 and x_3 respectively, which is received by neuron Y, the net input to Y will be $y_{in} = w_1x_1 + w_2x_2 + w_3x_3$. The activation y of neuron Y will be some function of y_{in} i.e. $y = f(y_{in})$. Some of the common activation functions are identity function, binary step function, binary sigmoid function and bipolar sigmoid function. If this signal is transmitted to some neurons Z_1 and Z_2, the inputs received by the neurons will be different as some weight, v_1 and v_2 will be multiplied with it. The simple neural network for the system being discussed is given in Figure 5.3 [116, 117].

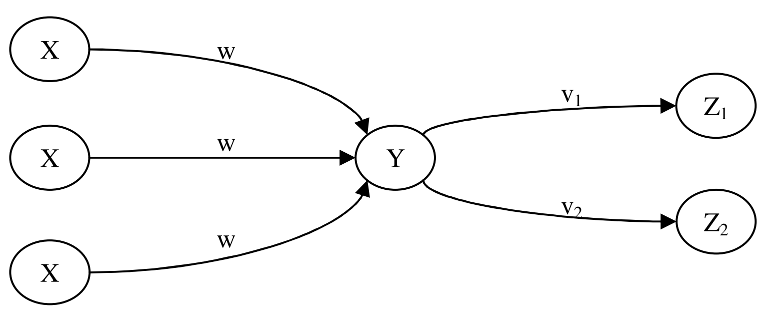


Fig. 5.3. A simple neural network

In a neural network, net architecture is the arrangement of neurons into layers and the connection pattern within and between layers. Generally neurons in the same layer have similar behaviour, same activation function and the same pattern of connections to other neurons. Neural nets can be either single or multi layered. The number of layers in a net can be defined as the number of layers of weighted interconnected links between different slabs of neurons [116, 117]. The difference between single or multi layered neural nets can be understood from Figure 5.4.

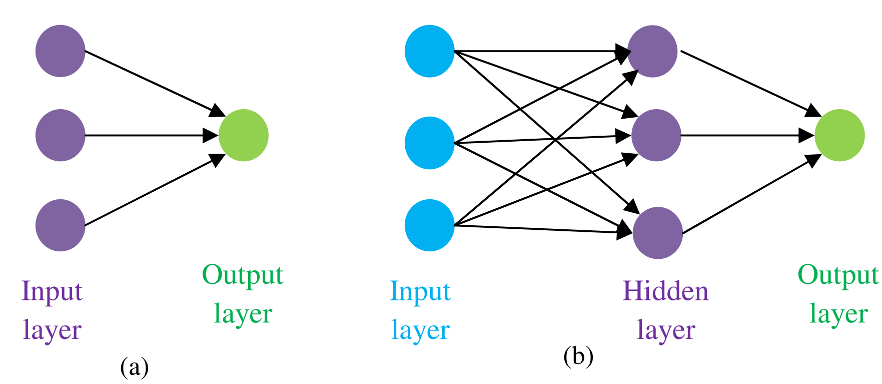


Fig. 5.4. Neural nets: (a) Single layered and (b) Two layered

The method of setting the values of the weights (training) is an important characteristic of any neural net which can be either supervised or unsupervised. In supervised training, machine is trained for every possible input and the necessary mapping between input and output is also provided. But in unsupervised training, machined is trained with a given set of input from which relationship between input and output is established for any unknown input. It is obvious that supervised training will be more accurate and faster than unsupervised training [116, 117]. From the discussions, it is clear that the direction of signal is from the input layer to the output layer. Such neural networks are called feed forward neural networks, which are very popular in pattern recognition. But in feedback neural networks, signals are bidirectional and loops are introduced into the network. This implies that the weights are updated according to the difference of actual and desired output, which will ultimately increase the system accuracy.

In [86], economic load dispatch of a 3 GS system for varying load conditions was obtained by using the lambda iteration method. The obtained result was used to train a back propagation neural network (BPNN) which is then used to predict the optimum generation of 3 GS for different load demands. The results obtained with BPNN based method was nearly same as that obtained with the lambda iteration method. In the developed setup, a feed forward ANN with user defined hidden layers is realized and is trained with the output data obtained with the lambda iteration method for the given system. Of the different available training function, Levenberg-Marquardt back propagation is used to update the weight and bias values. Levenberg-Marquardt algorithm was designed to minimize the sum of square error functions, as expressed in equation 5.13 where e_k is the error in k^{th} pattern and e is the vector containing e_k [118].

$$E = \frac{1}{2}\sum k \left(e_k\right)^2 = \frac{1}{2}\|e\|^2 \tag{5.13}$$

When the difference between the previous and present weight vector is small, the error vector can be expanded to 1^{st} order of Taylor series as expressed in equation 5.14 [118].

$$e(j+1) = e(j) + \frac{\partial e_k}{\partial w_i}(w(j+1) - w(j)) \tag{5.14}$$

The equation 5.13 can be expressed in terms of equation 5.14 as given in equation 5.15 [118].

$$E = \frac{1}{2}\|e\|^2 = \frac{1}{2}\left\| e(j) + \frac{\partial e_k}{\partial w_i}(w(j+1) - w(j)) \right\|^2 \tag{5.15}$$

Minimizing the error function with respect to new weight vector, the relation can be expressed as:

$$w(j+1) = w(j) + \left(Z^T Z\right)^{-1} Z^T e(j) \tag{5.16}$$

where $(Z)_{ki} = \frac{\partial e_k}{\partial w_i}$

For sum of square error function, Hessian will be expressed as given in equation 5.17 [118].

$$H_{ij} = \frac{\partial^2 E}{\partial w_i \partial w_j} = \sum \left\{ \frac{\partial e_k}{\partial w_i} \times \frac{\partial e_k}{\partial w_i} + e_k \times \frac{\partial^2 e_k}{\partial w_i \partial w_j} \right\} \tag{5.17}$$

Neglecting the 2nd term, Hessian can be given as H= $Z^T Z$.

Updating of the weights therefore involves the inverse Hessian or an approximation thereof for nonlinear networks. As the Hessian is based on first order derivatives with respect to the network weights, it is relatively easy to compute and can be easily accommodated by back propagation. Although the updating formula could be applied iteratively to minimize the error function, this may result in a large step size, which would invalidate the linear approximation on which the formula is based [118].

5.6. Unit Commitment (UC) Problem

As the total system load varies throughout the day and reaches a different peak value from one day to another, it becomes necessary for an electric utility to decide which generators to start up and when to connect them to the network and the sequence in which the operating units should be shut down and for how long in advance. Unit commitment is the computational procedure for making such decisions and a unit when scheduled for connection to the system is said to be committed. Normally, commitment of fossil-fuel units having different production costs, are considered due of their dissimilar efficiencies, designs, and fuel types. Although there are many factors which determines when units are scheduled on and off to satisfy the operational needs of the system, economics of operation, etc. are of major importance [35].

The solution of the unit commitment problem is complicated due to its dynamic nature. If 10 units are available for scheduling within any one-hour interval, theoretically a total of 1023 (= $2^{10} - 1$) combinations can be listed. Then, the total number of candidate combinations becomes 1.726 X 10^{72} (= 1023^{24}) for linking each prospective combination of any one hour to each prospective combination of the next hour of the day. This enormously large number of combinations is unrealistic to handle. The multistage decision process of the unit commitment problem can be reduced dimensionally by the practical constraints of system operations and by a search procedure based on the following observations [35]:

i. The daily schedule has N discrete time intervals or stages, the durations of which are not necessarily equal. Stage 1 precedes stage 2, and so on to the final stage N.
ii. A decision must be made for each stage k regarding which particular combination of units to operate during that stage. This is the stage k sub-problem.
iii. To solve for the N decisions, N sub-problems are solved sequentially in such a way that the combined best decisions for the N sub-problems yield the best overall solution for the original problem.

Dynamic programming (DP) can be useful for computing unit commitment table if load increases slightly in finite size steps [36]. A problem is decomposed into a series of smaller problems in DP which are solved subsequently to develop an optimal solution to the original problem. The DP algorithm developed for solving unit commitment problem, examines every possible state in every interval. If the states are found infeasible, they are rejected instantly. As a large number of feasible states may exist for an average size utility, the requirement of execution time will stretch the capability of even the largest computers. Therefore, many techniques were proposed where some sort of simplification and approximation can be included in the fundamental dynamic programming algorithm [119]. The steps followed in DP approach are given below:

i. Two units are considered randomly at the starting
ii. The collective outputs of the two units are assembled in the form of discrete load levels
iii. The most economical combination of the two units for all the load levels is determined. This economic operation may be to run either a unit or both units with a certain load sharing between the two units at each load level
iv. Most cost-effective cost curve for the two units is obtained in discrete form and is treated as cost curve of the single equivalent unit
v. The third unit is added and the cost curve for this three unit combination is obtained
vi. This procedure is repeated until all the existing units are considered

5.7. Parameters to be considered for the management of power system operational data

Power system operational data can be either generation scheduling information or parameter monitoring information [94]. Generating stations (GS), Distribution Company (DC) and Load Despatch Centre (LDC) are responsible for managing generation scheduling information. For DCs, the drawal schedule comprises of the amount of power export and import from (or to) other DC or power traders, allocated schedule by the LDC and actual drawal by the company(s) for all 96 time blocks. On the other hand, the generation schedule of GSs comprises of notional availability (possible generation when favourable condition is met), actual availability (generation possible in the present circumstances) and scheduled generation for all 96 time blocks [39, 94]. The different fields to be considered for managing generation scheduling information are discussed below.

i. Date and month information must be included to extract the exact file from a large database containing yearly information associated with generation scheduling.
ii. As some practical generation scheduling information is comprised of floating numbers, normalization information is also required for handling these data by using DBEA and E-DBEA.
iii. Modification of generation schedule might be required due to sudden tripping of transmission lines or inaccurate load forecasting. These modifications might result in multiple versions of the same data in a day. So, it becomes necessary to include version information as well.
iv. Type of data for GS can be notional availability, actual availability and scheduled generation. Similarly for DC, type of data includes amount of power export and import, allocated schedule by the LDC and actual drawal. For managing generation scheduling information, this information must also be included.

Management of parameter monitoring information is of more importance in the recent times. Due to the increased sampling frequency, there is a significant rise in the volume of information. Analysis of practical data indicates that the following fields must be considered for managing parameter monitoring information.

i. As monitoring will be in real time, it becomes necessary to include the time information during data management. This will enable to extract the data collected over the finite time duration from the large volume of such data.
ii. Similarly to generation scheduling information, date and month information must also be included for extracting the desired data.
iii. Similarly to generation scheduling information, normalization information must also be included. This information is more crucial in this type of information as parameters like frequency; system voltage, etc. will be some float values which require normalization. In contrast, power information are normally integer values which often do not require normalization.
iv. Similarly to generation scheduling information, information regarding the type of data, i.e. active power, system frequency, energy billing etc. must also be provided.

v. As power transfer though a power line is bidirectional, the associated data can be positive or negative. As DBEA or E-DBEA can't handle negative first element, information about the polarity of first element must be included so that the exact data array can be extracted at the decoding end.

Keeping in view the existing power system operation, several technological advantages can be obtained if a cloud based power system operational data management system can be realized. Some of the advantages of the proposed system over existing system are enlisted below.

i. In the existing system, all the data are stored in .xls file day-wise or version-wise. If separate files containing compressed data corresponds to different GS (or DC) or different monitoring parameter are maintained, reduced memory will be required for store such data and there will be a considerable reduction in system cost as well [72].

ii. In cloud computing, computer resources are shared among different systems and thus can be scaled up and down according to the requirement. This will reduce hardware requirements and increases the system flexibility.

iii. The objective of power industry is to maximize the electricity generation efficiency so as to have reduced generation cost. This can be achieved by considering unit constants, forecasted load demand and loss coefficients. In the existing system, LDC receives the next day availability and load forecasting information via internet and obtains next day generation and load allocation which are conveyed by internet. Data handling would be much easier if separate files are maintained for each GS or DC and are shared by cloud environment [50, 51].

iv. If a cloud based data management system can be realised for managing parameter monitoring data, there will not only be a considerable reduction in memory requirement, but also includes the provision of selecting a particular data (say the data array containing some abnormal data) from a large chunk of data without any regular monitoring.

Chapter 6

PERFORMANCE ANALYSIS AND REALIZATION OF PRACTICAL SETUP

6.1. Development of DBEA compressed DCSK for PLCC

PLCC is still in operation for transferring data through power lines. Conventional modulation techniques are not suitable for the power line environment due to different noise present in the power line. With the increasing number of electronic devices, the noise increases drastically and information loss becomes more prominent. DCSK modulation scheme, patented by Yitran Technologies was developed for data transfer through power line and is a robust communication scheme in the presence of power line noise. The DCSK modulation scheme was developed in MATLAB environment for obtaining the superchirp corresponds to an input character string at the encoding end [120]. From the superchirp, the actual character string is decoded at the decoding end by comparing the shifted chirps with a non-shifted reference chirp. As data compression will result in reduced length of information, reduced energy will be required for transmitting the compressed data. So, in the subsequent works, Basic Arithmetic Coding [121] and Huffman Coding [122] were used to reduce the length of superchirp. The comparative analysis of the three works is included in [123] to determine their advantages and disadvantages under certain circumstances. As DBEA is an effective compression algorithm for power system applications, a DBEA compressed DCSK modulation scheme is realized in MATLAB. At the encoding end, a large data array is compressed by using DBEA and the corresponding superchirp is obtained. From this superchirp, the actual data array is decoded at the decoding end. As DBEA can be embedded at low level microcontrollers, it is also possible to realize a compressed Data Acquisition System (DAS) employing power line as the communication medium between the field devices and control room. The block diagram of the proposed DBEA compressed data transfer system is given in Figure 6.1 [124].

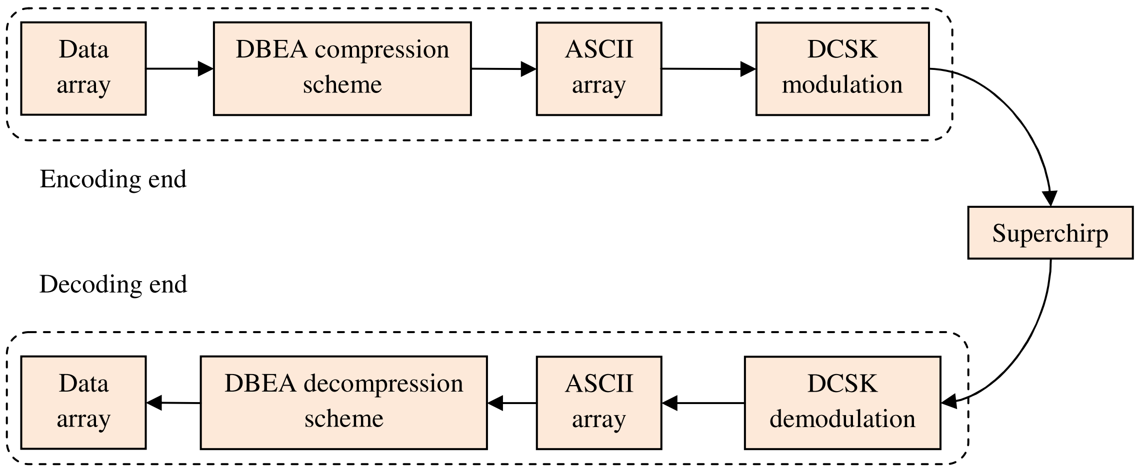


Fig. 6.1. Block diagram of DBEA compressed DCSK modulation scheme

6.1.1. DCSK Modulation scheme for PLCC

In this modulation scheme, any spreading waveform of constant length (chirp waveform in this case) is rotated either in clockwise or counter clockwise direction depending upon the data value and this shifted chirp waveform is termed as Unit Symbol Time (UST). At the receiving end, a series of rotation of the UST in reverse direction and simultaneous comparison with the non-shifted reference waveform is performed to determine the amount of rotation. This will give the information of the transmitted data. There are two possible alternatives of DCSK communication scheme. In the first one, the data is conveyed in the amount of rotation of the chirp only. But alternatively, data can also be conveyed in the differential shift between two consecutive chirps. For the sake of simplicity, the first alternative is considered for data transfer. For the ramp waveform divided in 10 equal sections, given in Figure 6.2 (a). For transmitting symbol '4' and '7' employing DCSK (Figure 6.2 (b) and 6.2 (c)), the actual waveform is rotated by 4 and 7 sections respectively in the counter clockwise direction. Some of the advantages of this DCSK modulation scheme include higher transmission reliability, fast and simple synchronization, immediate recovery after severe fading and higher data throughput rate [120-124].

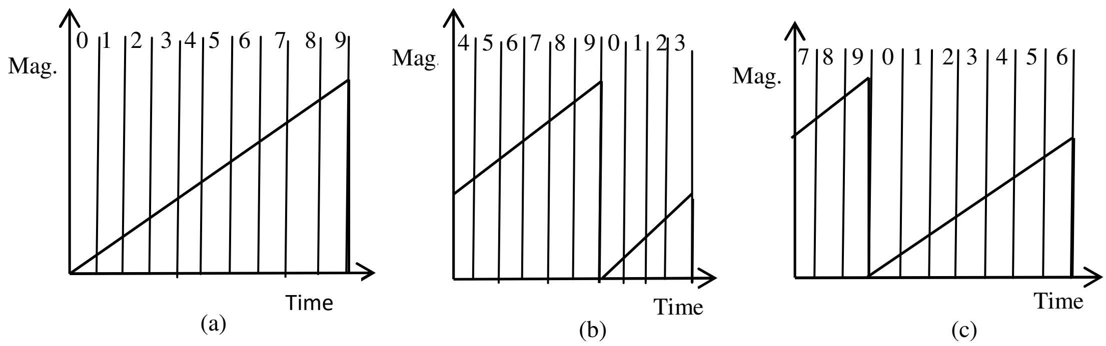
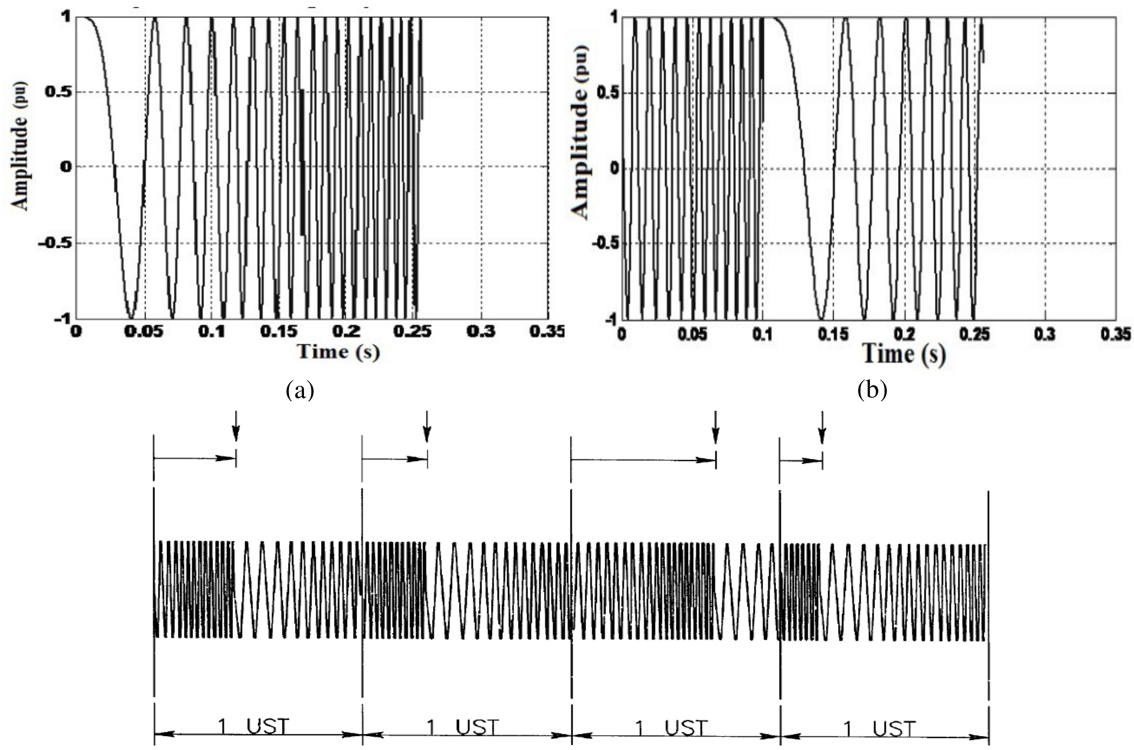


Fig. 6.2. Simple illustration of DCSK modulation scheme (a) Non- shifted waveform; (b) Waveform for transmitting symbol '4' and (c) Waveform for transmitting symbol '7'

The waveform required to perform DCSK modulation can be any waveform having suitable auto correlation properties and chirp waveform is most commonly employed. These chirp waveforms are frequency modulated signal of constant length, which is rotated according to the data value. This rotated chirp waveforms are termed as 'Unit Symbol Time' (UST). There are two alternatives for designing this system. In the first arrangement, the data is conveyed in the amount of rotation of the individual chirp. Alternatively, data may be conveyed in the differential shift between two consecutive chirps. The first case is considered in this case owing to its simplicity. A non-shifted chirp waveform and a shifted chirp waveform (UST) are given in Figure 6.3 (a) and 6.3 (b) respectively. At the receiving end, the extent of rotation is determined by comparing the shifted waveform with the non-shifted waveform. This is done by a series of rotation of the shifted waveform and simultaneous comparisons with the non-shifted waveform. For transmitting a string with n characters using DCSK modulation scheme, the corresponding superchirp will be a combination of n shifted chirp waveforms (USTs).At the receiving end, the superchirp is divided in individual USTs which are then compared with the reference non-shifted chirp waveform to decode the number array. Figure 6.3 (c) gives a superchirp waveform comprising of four USTs and the arrow indicates the extent of rotation of individual USTs. If d is the number of significant digits of final frequency (ff), fi is the initial frequency and D is the number of divisions of the chirp waveform, sampling time (tdel) and UST length (tf) of the waveform can be calculated by equation (6.1) and (6.2) respectively [120-124].

$$\text{tdel} = 10^{-(d+2)} \text{ sec} \qquad (6.1)$$

$$\text{tf} = (D * t' - \text{tdel}) \text{ sec, where } t' = 1/(\text{ff} - \text{fi}) \qquad (6.2)$$



(c)

Fig. 6.3. Chirp waveform based DCSK: (a) Non-shifted chirp; (b) Clockwise shifted chirp and (c) Superchirp [101]

These chirp waveforms are frequency modulated signal of constant length, which is rotated according to the data value.

Input: ch: Input character; fi, ff: Initial and final frequency of chirp waveform

Output: r(t): Non-shifted reference chirp; u(t): Shifted chirp (UST)

Begin

    asci= ASCII value of ch

    Chirp length= 255.99 X10^{-n} sec. (n being the no. of significant bits of ff)

    r(t) of desired length with given fi &ff is produced

    Divide r(t) in 256 equal sections

    Initialize cnt=0

    Repeat until cnt!=asci

        Rotate the r(t) 1 unit counter clockwise

        cnt= cnt+1

    end

    u(t)= Waveform obtained by rotating r(t) asci times

End

At the receiving end, the transmitted waveform is obtained and continuously compares with the non-shifted chirp signal as used at the transmitting end to determine the amount of rotation. The pseudo code of decoding the modulated signal at the receiving end employing the DCSK communication scheme is given below [120-124].

Input: r(t); u(t)
Output: ch

```
Begin
    Split u(t) in 256 equal sections
    Initialize cnt=0
    Repeat until r(t) and u(t) are same
        Rotate u(t) 1 unit clockwise
        cnt= cnt+1
    end
    ch= Character equivalent of cnt
End
```

For string transmission using DCSK communication scheme, the output will be a superchirp comprising a no. of chirps rotated corresponding to the characters in the string. The pseudo code of transmitting a character string employing the DCSK communication scheme is as below [120-124].

Input: str: Input character string; fi; ff
Output: r(t); s(t): Superchirp

```
Begin
    l= Length of str
    Initialize i=1 and s(t)= NULL
    Repeat until i ≤ l
        u(t)= UST corresponds to i^th character of str
        s(t)= Merge s(t) and u(t)
    end
End
```

At the receiving end this received superchirp is sub-divided in the corresponding chirps. Then the amount of rotation of the chirps is determined by continuously comparing with a non-shifted chirp signal. From this information, the different characters of the character string are obtained. The pseudo code employed at the demodulating end is given below [120-124].

```
Input: r(t); s(t)
Output: str

Begin
    l= Length of s(t)/ length of r(t)
    Split s(t) in l USTs
    Initialize i=1 and str= NULL
    Repeat until i ≤ l
        ch= Character corresponds to i^th UST
        str= Concatenate str and ch
    end
End
```

6.1.2. DCSK based DBEA Compressed data transfer

The algorithms employed for transferring the compressed information through power line [123] are based on entropy based compression algorithms where the character string is compressed by Basic Arithmetic [121] and Huffman Coding [122]. The CR obtained with the algorithms is not high either and varies typically between 1.5 and 3. Analysis of practical power system monitoring data indicates that the difference between two consecutive data elements do not vary widely during stable grid conditions. This implies that DBEA is capable to compress majority of the data array formed during steady state grid conditions [93, 94]. As DBEA can reduce the volume of information drastically for repetitive data array, there will be a significant reduction in the superchirp length for transferring such information. Unlike the previous chapter, this section describes the communication prospect of transferring compressed information collected over a finite time interval through power lines.

Implementation of DBEA for compressing unit commitment and parameter monitoring data gives good results for the majority of data. As real testing of DBEA is successful between two PCs connected through serial cable [94], it is obvious that DBEA compressed data array transfer will be possible by using the DCSK communication scheme. As PLCC is still in operation for data transfer in power system application, a DBEA compressed data transfer scheme employing the DCSK communication scheme is developed and tested offline in MATLAB environment. At the encoding end, a large data array associated with parameter monitoring data is compressed to a character string by using DBEA. The superchirp corresponds to this character string is obtained which can be transmitted over power lines. This superchirp and reference chirp being obtained is stored in the computer for decoding purposes. At decoding end, the actual character string is obtained after splitting the superchirp to individual chirp and is compares with the reference non-shifted waveform to extract the information contained in it. Actual array is the then decoded from the compressed information by using DBEA scheme. The pseudo code of the algorithm being employed at the encoding end is discussed below [124].

```
Input: arr[l]; fi, ff
Output: s(t); r(t)

Begin
        Initialize i=1
        dif[1]=arr[1]
        i=i+1
        Repeat until i<(n+1)
                dif[i]=arr[i]-arr[i-1]
                i=i+1
        end
        Formation of mod[m] by replacing successive zeros by their count
        Binary encoding of individual elements of mod[m] is done.
        Initialize j=1 and str= NULL
        Repeat until j<(m+1)
                asci[j]= Decimal equivalent of binary encoded mod[i]
```

```
        j=j+1
    end
    r(t) is produced according to fi and ff
    Define s(t)= NULL and initialize k:= 0
    Repeat until k<m
        UST= Rotate r(t) counter clockwise (or clockwise) according to asci [i]
        s(t)= Merge UST with s(t)
        k=k+1
    end
End
```

The pseudo code to obtain actual data array from superchirp at the decoding end is given below [124].

```
Input: s(t), r(t)
Output: act [n]: Decoded data array having n elements

Begin
    n is calculated by comparing s(t) and r(t)
    Initialize i= 1
    Repeat until i<m+1
        Obtain (i+1)th UST from s(t)
        Initialize cnt= 0
        Rotate UST clockwise (or vice versa) by 1 unit and compare with r(t)
        If both are identical
            asci[i]=cnt
        Otherwise
            cnt= cnt+1
        end
        k=k+1
    end
    modi[1]= 256*asci[1]+ asci[2]
    Initialize j=3
    Repeat until j<m+1
```

```
        x= asci[j]
        If x< 128
            modi[j-1]=x
        Else if x> 192
            modi[j-1]=192- x
        Otherwise
            modi[j-1]= x-128
        end
        j=j+1
    end
    dif[1]= modi[1]
    Initialize k= 2 and cnt= 2
    Repeat until k<m+1
        y= modi[k]
        If y< 128
            dif[k] to dif[k+ modi[j]] will be zero
            k=k+1
            cnt=cnt+ modi[j]
        Else if y> 192
            dif[k]= -1*y
            cnt= cnt+1
        Otherwise
            dif[k]= y
            cnt= cnt+1
        end
        k=k+1
    end
    act[1]= dif[1]
    Initialize i1= 2
    Repeat until i1<n+1
        act[i1]= dif[i1]+ dif[i1-1]
    end
End
```

6.1.3. Obtained result

The results obtained by DBEA are satisfactory for the majority of the parameter monitoring and unit commitment data. The CR varies over a wide range owing to the fact that CR varies exponentially with the number of changes in the data array. This implies that the length of superchirp and hence the number of USTs required to form the superchirp varies with the data array. When the generation scheduling information of Barh super thermal GS on 23/7/16 is compressed with DBEA, the character string comprises of seven characters. While executing the proposed algorithm with the same sample array in MATLAB, the superchirp corresponds to the compressed character string will have seven USTs in it. Superchirp corresponds to the sample generation scheduling information will be as given in Figure 6.4 when the initial and final frequency is 0 Hz and 100 Hz respectively.

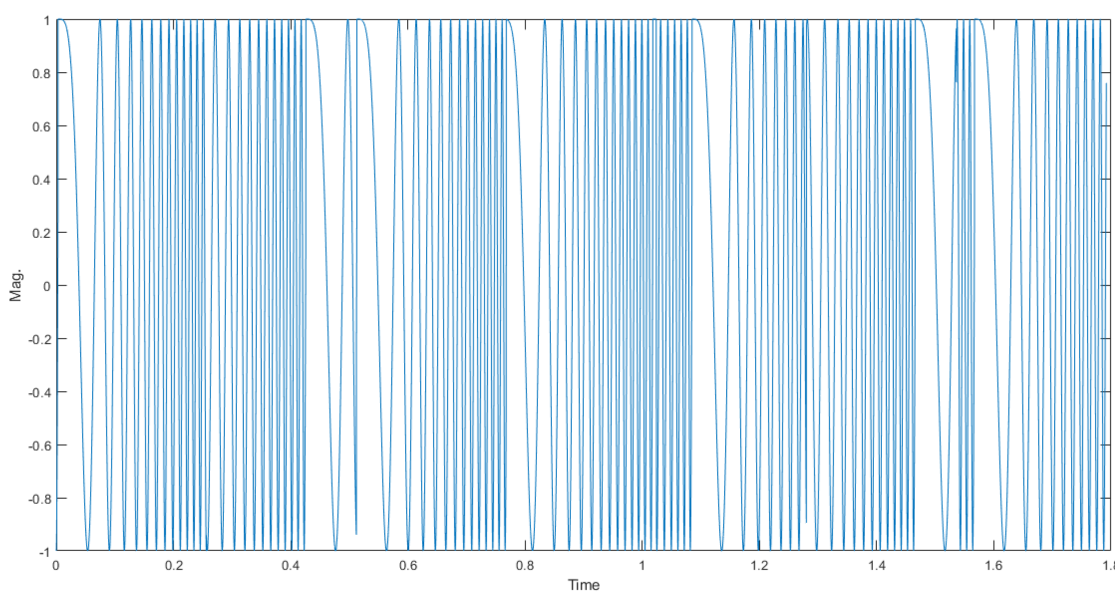


Fig. 6.4. Superchirp obtained with proposed algorithm while compressing the generation scheduling information array with initial and final frequencies 0 Hz and 100 Hz respectively

The unit commitment information of Barh super thermal GS over a week [51] are compressed by DBEA and the corresponding superchirp were obtained by DCSK based communication scheme. Few unit commitment information data of GS operating under West Bengal Power Development Corporation Limited (WBPDCL) [50] are also compressed and the corresponding superchirp is obtained. The compression ratio and number of USTs in the superchirp being obtained for those data samples is given in

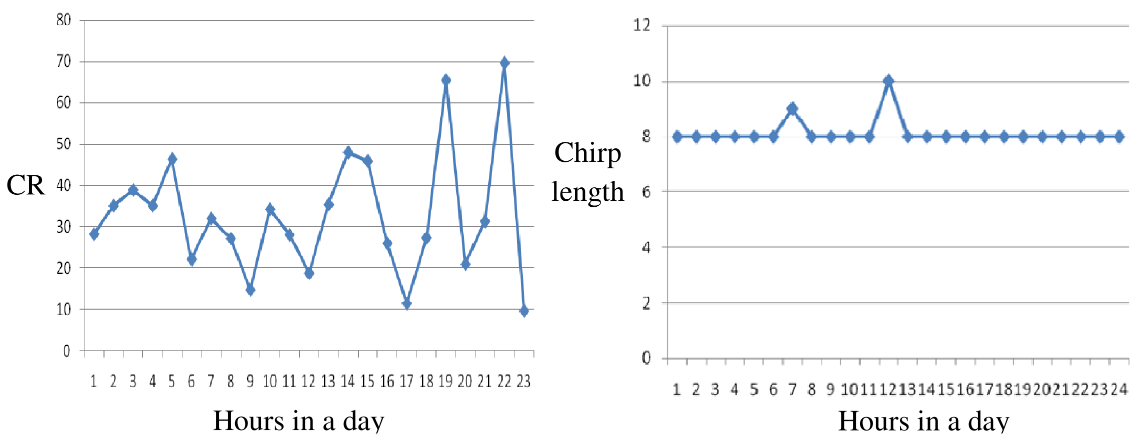
Table 6.1. It is clear from the figure that higher will be the number of USTs in the superchirp, lower will be the CR and vice versa. With increased number of changes in the data array, the number of characters in the encrypted string will increase, thereby increasing the number of USTs and reducing the CR.

Table 6.1

Performance analysis for practical unit commitment data

Sl. No.	Date	Name of GS	Block size (Bytes)	CR	No. of UST	Entropy before comp.	Entropy after comp.	Execution time (in sec)	
								At encoding end	At decoding end
1	06/03/ 2015	BTPS	288	15.93	15	1.99	2.708	1.525	0.083
2	24/03/ 2015	STPS		72	4	1.59	1.386	1.422	0.053
3	17/04/ 2015	KTPS		32	9	1.91	2.197	0.925	0.052
4	20/05/ 2015	STPS		14.4	20	1.63	2.996	1.677	0.059
5	28/05/ 2015	BkTPP (1 to 3)		48	6	0.98	1.792	2.797	0.066
6	02/06/ 2015	SgTPP		18	16	2.21	2.773	2.147	0.154
7	09/06/ 2015	BkTPP (4 and 5)		28.8	10	2.13	2.303	1.023	0.055
8	13/06/ 2015	SgTPP		13.1	22	2.18	3.091	1.637	0.065
9	22/07/ 2016	NTPC, Barh		96	3	1.59	1.099	0.784	0.045
10	23/07/ 2016	NTPC, Barh		41.14	7	1.68	1.946	1.465	0.087
11	24/07/ 2016	NTPC, Barh		96	3	1.59	1.099	0.634	0.061
12	25/07/ 2016	NTPC, Barh		96	3	1.59	1.099	1.401	0.07
13	26/07/ 2016	NTPC, Barh		96	3	1.59	1.099	1.15	0.06
14	27/07/ 2016	NTPC, Barh		96	3	1.59	1.099	1.321	0.059
15	28/07/ 2016	NTPC, Barh		48	10	1.64	2.303	1.071	0.067

From the previous discussions, it is clear that DBEA based DCSK scheme is giving pretty good result when there is a huge repetition of data is encountered. In order to avoid any additional or reduced drawl to avoid the UI charges, both GS and transmission companies tend to stick to its forecasted drawl conditions. This implies that the parameter monitoring data do not vary extensively with time and thus DBEA based compression can give a pretty good compression ratio for majority of the testing data obtained from WBSLDC. The hourly data associated with data transferred through a 400 kV line and generation data of a hydel generating station over an entire day is compressed by DBEA. The variation of CR obtained with hourly power transfer data over the day is given in Figure 6.5 (a). Figure 6.5 (b) gives the variation of the number of USTs in the superchirp obtained while compressing hourly generation data by DBEA.



(a) (b)

Fig. 6.5. Variation of (a) CR of power transfer data through a 400kV line during entire day and (b) Chirp length for generation data of a Hydel GS

6.2. Comparison of Lambda Iteration, IW-PSO and ANN for solving ELD problem

The generation scheduling algorithms discussed in Chapter 5 was tested with IEEE 5-bus system data. For IEEE 5-bus system, the cost coefficients, limit of generation of two GS and transmission loss matrix are given as:

$a_1= a_2 =0.005$; $b_1= 3.51$; $b_2= 3.89$; $c_1=44.4$; $c_2=40.6$; $30\text{MW} \leq P_{G1}, P_{G2} \leq 120$ MW, $B = 10^{-4}\begin{bmatrix} 3.49 & 0.86 \\ -0.55 & 3.71 \end{bmatrix}$.

While testing the lambda iteration algorithm for the GS of IEEE 5-bus system with load demand varying between 60MW and 220MW, the variation of MW generation of two GS is given in Figure 6.6. It is clear from the figure that GS1 will take the maximum load to supply the load demand over the entire range. It is also clear from the figure that there is an increase in transmission load with increase in load demand.

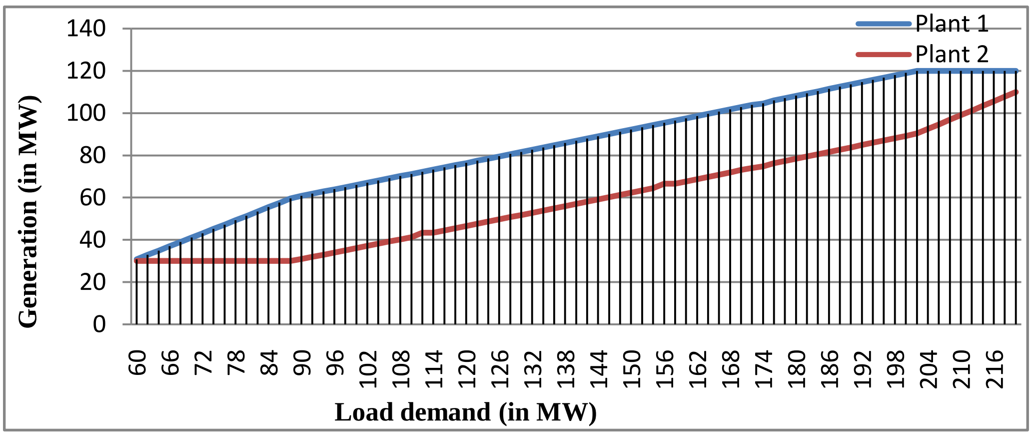


Fig. 6.6. Variation of generation of two GS of IEEE 5-bus system with variation of load demand as computed by λ-iteration method

When IW-PSO algorithm was tested with IEEE 5-bus system parameters, the generation variation of two GS with the variation of load demand for particle size of 30 and 50 is given in Figure 6.7. The IW-PSO parameters are initialised as $c_1=c_2=2$, $w_{max}=0.9$, $w_{min}=0.4$, particle size= 30 and Maximum number of iterations=1000.

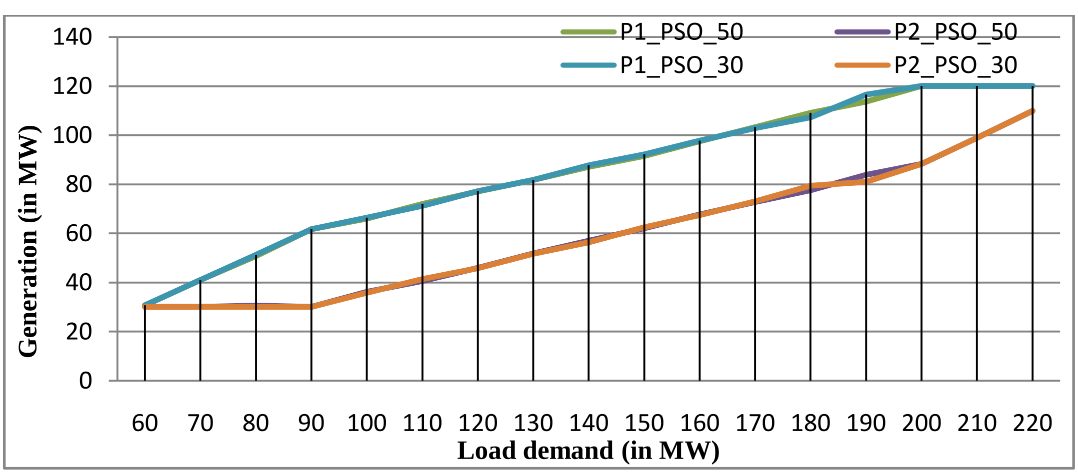


Fig. 6.7. Variation of generation of 2 GS of IEEE 5-bus system with load demand while tested with IW-PSO based algorithm with particle size of 30 and 50

The comparison of the results being obtained with the lambda iteration algorithm and IW-PSO based algorithm with varying load demand is given in Table 6.2. When the particle size is increased to 50, the execution time increases but there is a little variation in the results being obtained.

Table 6.2
Comparison of the results obtained with lambda iteration algorithm and 30 particle size IW-PSO based algorithm for varying load demand

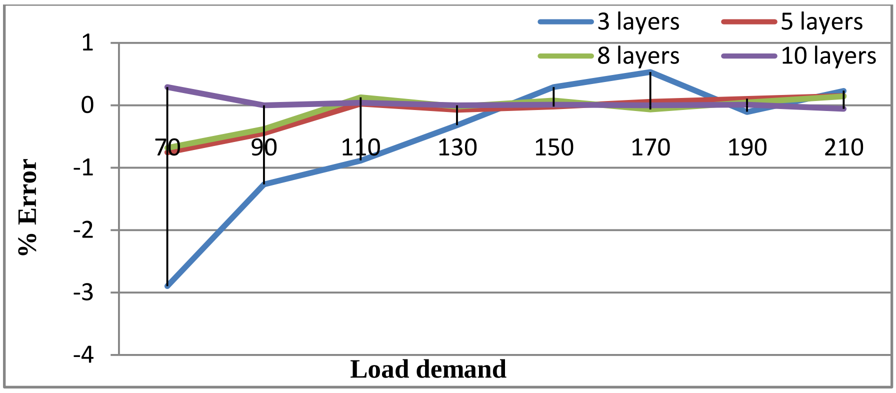
Sl. No.	**Load demand (MW)**	**Generation of Plant 1 (MW)**		**Generation of Plant 2 (MW)**		**Epoch required**	
		with λ iteration	**With IW-PSO**	**with λ iteration**	**With IW-PSO**	**For λ iteration**	**For IW-PSO**
1	60	30.83	30.69	30	30	18	8
2	70	41.08	40.98	30	30	17	240
3	80	51.37	51.34	30	30	18	278
4	90	60.84	61.72	30.86	30	16	275
5	100	66.01	66.33	36.06	35.76	16	391
6	110	71.21	71.14	41.28	41.35	17	252
7	120	76.4	76.49	46.49	46.46	10001	437
8	130	81.67	81.72	51.78	51.74	18	507
9	140	86.93	86.45	57.07	57.54	18	428
10	150	92.22	92.14	62.37	62.44	18	293
11	160	97.53	97.67	67.69	67.56	17	383
12	170	102.86	102.87	73.04	73.04	15	347
13	180	108.22	107.19	78.41	79.43	17	302
14	190	113.61	116.44	83.8	81.02	16	335
15	200	119.01	120	89.21	88.24	16	232
16	210	120	120	99.09	99.03	18	192
17	220	120	120	110	109.97	10001	219

The 10 hidden layers feed forward ANN using Levenberg-Marquardt back propagation training function was trained with the 41 generation data of the two GS given in IEEE 5-bus system obtained with the Lambda-iteration method. The ELD solution obtained with the trained neural net with some untrained load demand is given in Table 6.3. The variation of percentage errors of the solution of the ELD problem for IEEE 5-bus system is given in Figure 6.8. The errors are computed for different number of hidden layers and are plotted with respect to the variation of load demand. From the figure, it is clear that the neural net with 10 hidden layers gives extremely good results.

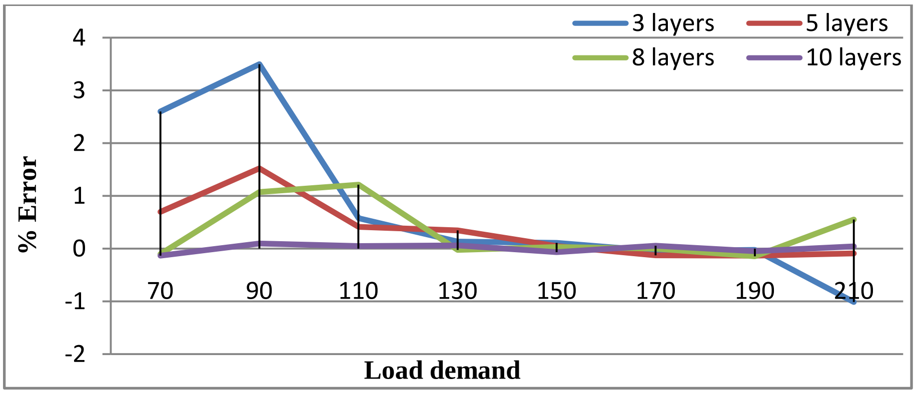
Table 6.3

Comparison of the results obtained with 10 hidden layers feed forward ANN and λ iteration method with varying load demand

Sl. No.	Load demand (MW)	Generation of Plant 1 (MW)		Generation of Plant 2 (MW)		Epoch required	
		With lambda iteration	With 10 Layer ANN	With lambda iteration	With 10 Layer ANN	For lambda iteration	For 10 Layer ANN
1	70	41.08	41.2	30	29.96	17	11
2	90	60.84	60.84	30.86	30.89	16	
3	110	71.21	71.24	41.28	41.3	17	
4	130	81.67	81.67	51.78	51.81	18	
5	150	92.22	92.23	62.37	62.33	18	
6	170	102.86	102.86	73.04	73.08	15	
7	190	113.61	113.62	83.8	83.76	16	
8	210	120	119.93	99.09	99.13	18	



(a)



(b)

Fig. 6.8. Variation of percentage error with load demand obtained with different number of layers (a) For plant 1 and (b) For plant 2

6.3. Comparison of DBEA, E-DBEA and RLDA for compressing Power System operational data

DBEA and E-DBEA are novel low computational compression algorithms developed for compressing large repetitive data array to a character string of much reduced size. RLDA, developed for compressing load profile data was found to be a good choice for compressing power system operational data as well [92].

6.3.1. Experimental setup

DBEA, E-DBEA and RLDA routines were developed in MATLAB environment and tested offline with unit commitment data of various state and national GS [50, 51] and hourly system monitoring information collected from WBSLDC. RLDA is developed and tested with unit commitment and parameter monitoring data for comparing the performance with DBEA and E-DBEA. All the available works are based on offline compression of data. To overcome this limitation, DBEA and E-DBEA are tested in real time between two systems. Real time testing setup of DBEA and E-DBEA between two computers is developed in MATLAB environment using serial port instructions. The experimental setup for online testing of the algorithm is given in Figure 6.9. One system resembles virtual GS/ substation and transfers compressed information to the LDC/data centre through the suitable communication channel (RS232 in this case). The system at the virtual LDC/data centre decodes the information to obtain the original data. During real time testing, it was found that an additional character with ASCII value 10 is added with the actual string while transferring it through RS232 cable. This additional terminating character must be removed from the string before decoding the information so that any false interpretation can be avoided. Although data transfer through serial cable is not among the state of art communication technique, this technique was used just to validate the ability of the proposed algorithms for real time data transfer. As all the modern communication techniques are based on serial communication, it is expected that the proposed real time test bench will work successfully when connected with those communication channels for data transfer.

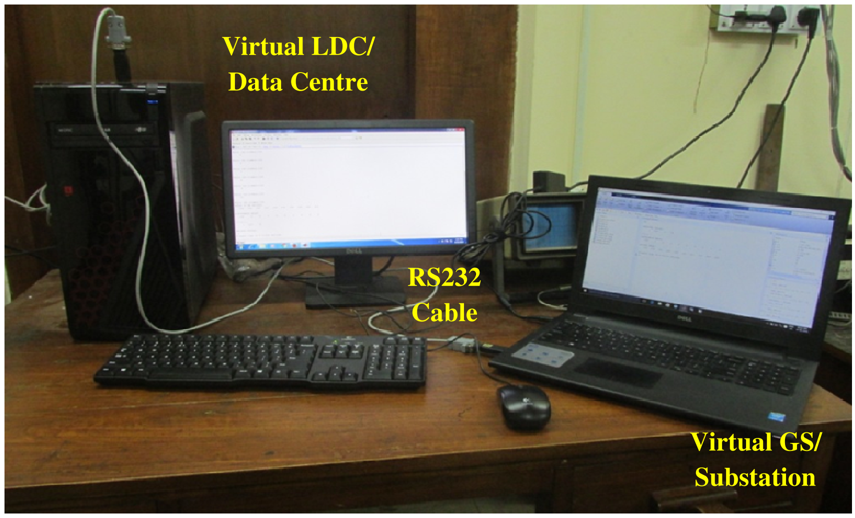


Fig. 6.9. Experimental set-up for real time testing of DBEA and E-DBEA

While testing DBEA in the experimental setup given in Figure 3.8, it was found that the algorithm was not giving the desired results in some instants. But the algorithm was giving the desired results while offline testing where the character string being obtained is stored in the HDD of the PC at the encoding end in .txt format. This file is extracted at the decoding end and is decoded to obtain the actual data array. Surprisingly, this problem doesn't occur while real time testing of E-DBEA. While investigating the problem, it was found that the problem will arise when the ASCII value of the character exceeds 127.It was due to the fact that serial communication does not allow transferring of extended ASCII characters obtained with DBEA compression scheme. To overcome the problem, 7-bit DBEA was developed where the output string will have characters with ASCII values varying between 0-127. Due to the presence of identity bit (s), it is not possible to remove a bit from the beginning. So, while removing the bit from the end, the range reduces further. This observation was considered while approaching towards extended versions of DBEA.

The simplest mode of data transfer between two computers can be performed via serial /USB port. The work will be extended to develop a test bench to test the algorithms between two computers with internet connectivity. For practical applications, the computer networking protocol must be incorporated and requires DTP programming at both ends. The computer at the encoding end will encode the updated data array using DBEA or E-DBEA and transfer the encrypted file through the internet.

The computer at the other end will continuously look for any new file availability at a predefined time interval and decode the file to obtain the updated data array whenever it is available.

6.3.2. Obtained result

In DBEA, string size will be same as the number of elements in the modified array which again is dependent on the number of changes in the array to be compressed. With increased number of changes in the data array, number of characters in the string will increase and thereby reducing the compression ratio significantly. The algorithm is tested with various unit commitment data containing 3-digit or 4-digit elements only. The CR obtained by DBEA varies with the number of changes in the data array and is given in Figure 6.10.

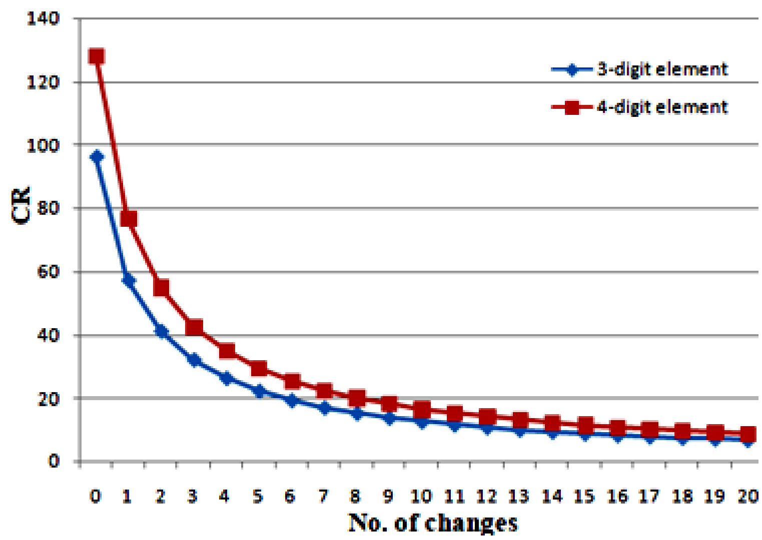


Fig. 6.10. Variation of CR for DBEA with the number of changes

DBEA and E-DBEA is tested with unit commitment data of various GS and the results are given in Table 6.4. It is clear from the table that, though DBEA fails for few data sets, E-DBEA can successfully compress all data sets. The average CR obtained by compressing the unit commitment information by DBEA, E-DBEA and RLDA for the GS considered in Table 6.4 is 26.99, 15.93 and 15.08 respectively.

Table 6.4

CR obtained by DBEA, E-DBEA and RLDA for unit commitment data

GS	Date	Block size (Bytes)	No. of changes	CR with DBEA	CR with E-DBEA	CR with RLDA
BTPS	06/03/15	223	6	15.93	7.97	10.67
KTPS	06/03/15	288	7	-	10.29	11.44
STPS	24/03/15	288	0	96	72	32.42
SgTPP	24/03/15	288	10	14.4	7.58	9.74
BkTPP (1 to 3)	31/03/15	288	8	19.2	11.08	11.3
STPS	02/04/15	288	16	12.52	6.26	8.87
BTPS	08/04/15	238	10	11	5.79	8.4
SgTPP	17/04/15	288	19	8.47	4.24	7.12
KTPS	17/04/15	288	3	32	18	23.17
BkTPP (1 to 3)	25/04/15	288	10	16	8.47	10.4
STPS	25/04/15	288	8	-	12	12.56
BTPS	05/05/15	216	9	10.8	5.68	8.94
BkTPP (4 to 5)	05/05/15	288	4	28.8	16	18.33
SgTPP	12/05/15	258	20	-	3.91	6.16
BTPS	12/05/15	192	7	12	6	8.61
STPS	20/05/15	288	13	15.16	7.58	9.45
STPS	28/05/15	288	6	19.2	16	16.98
BkTPP (1 to 3)	28/05/15	288	2	48	28.8	27.07
SgTPP	02/06/15	288	7	18	9.6	12.63
STPS	02/06/15	288	0	96	72	32.42
BTPS	09/06/15	192	1	38.5	24	27.53
BkTPP (4 to 5)	09/06/15	288	4	28.8	16	18.41
SgTPP	13/06/15	288	12	13.1	6.86	9.09
BkTPP (1 to 3)	13/06/15	288	3	32	18	20.15

The result obtained by compressing the unit generation data of a hydro GS in the month of December 2016 is given in Table 6.5. From Table 6.4 and 6.5, it is clear that the CR obtained by E-DBEA is approximately half of that obtained by DBEA. But for large repetitive data array, E-DBEA will perform better than DBEA as it can compress a maximum of 8191 consecutive zeros contained in the differential array in merely two characters. In comparison, bit-string output of RLDA is very large for a majority of data sets and thereby CR will be lower. It is also visible from Table 6.5 that RLDA fails when all elements of the array is zero.

Table 6.5

String size obtained by DBEA, E-DBEA and RLDA for monitoring data

Hours	Actual size (Bytes)	String size with DBEA (Bytes)	String size with E-DBEA (Bytes)	Binary string size with RLDA (Bits)
1	720	8	4	-
2	720	8	4	-
3	720	8	4	-
4	1224	12	16	253
5	1440	9	8	229
6	1440	9	8	234
7	1440	9	8	233
8	1440	8	4	115
9	1440	8	4	115
10	1440	10	12	232
11	1440	8	4	119
12	1286	12	14	241
13	720	8	4	-
14	720	8	4	-
15	720	8	4	-
16	720	8	4	-
17	1284	11	12	239
18	1629	15	22	267
19	2160	8	4	123
20	2160	8	4	123
21	2160	8	4	123
22	1818	13	16	261
23	1440	8	4	117
24	1359	9	8	228

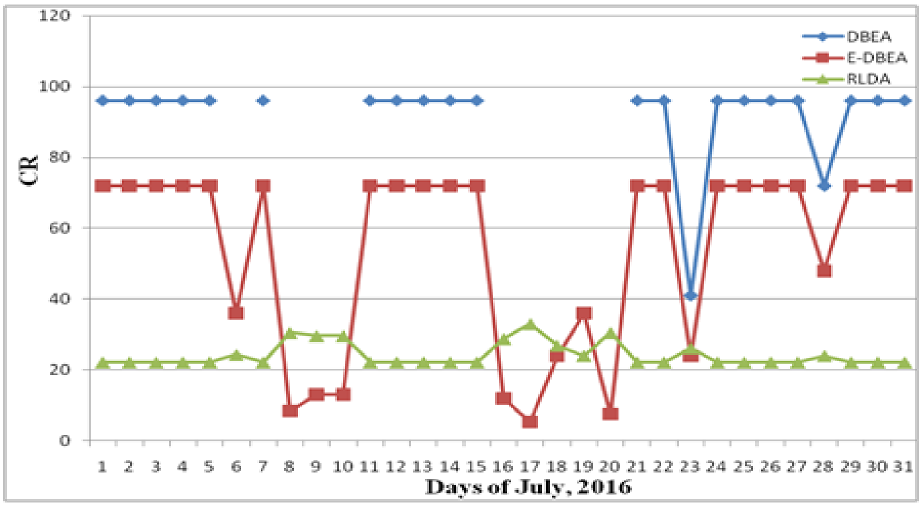
The variation of execution time for DBEA and E-DBEA with the variation of input array size is given in Table 6.6. From the table it is clearly visible that the execution time of DBEA and E-DBEA is extremely low (even less than 0.1s) which is primarily due to extremely low computation involved in it.

Table 6.6

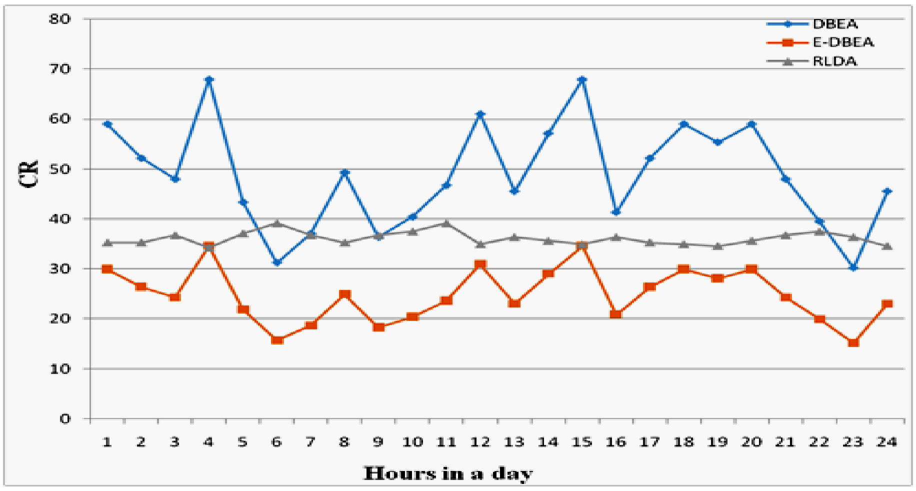
Effect of array size on the performance of DBEA and E-DBEA sample slow varying repetitive and non- repetitive data sets

No. of elements	**Type of array**	**DBEA**			**E-DBEA**		
		CR	**Execution time at encoding end (ms)**	**Execution time at decoding end (ms)**	**CR**	**Execution time at encoding end (ms)**	**Execution time at decoding end (ms)**
10	Repetitive	10	11.76	22.21	7.5	8.74	0.82
20		20	5.18	0.63	15	4.86	0.97
30		30	34.39	7.24	22.5	43.56	8.02
40		40	2.51	0.51	30	1.82	0.46
50		50	1.85	1.5	37.5	1.16	0.53
60		60	1.68	0.57	45	1.47	0.46
70		70	2.07	0.57	52.5	1.22	0.51
80		80	1.68	0.68	60	1.33	0.62
90		90	1.64	0.8	67.5	2.09	0.73
100		100	1.93	0.72	75	1.47	1.79
10	Non- repetitive	2.73	20.27	12.48	1.5	26.54	3.4
20		2.86	9.17	9.94	1.5	20.61	9.1
30		2.9	2.35	1.49	1.5	10.8	3.61
40		2.93	2.58	0.67	1.5	3.25	5.66
50		2.94	2.8	0.5	1.5	4.46	6
60		2.95	3.2	0.66	1.5	7.55	5.83
70		2.96	3.88	0.62	1.5	4.33	8.62
80		2.96	4.38	1.01	1.5	4.96	7.79
90		2.97	3.82	1.08	1.5	7.43	10
100		2.97	3.58	0.85	1.5	5.85	10.41

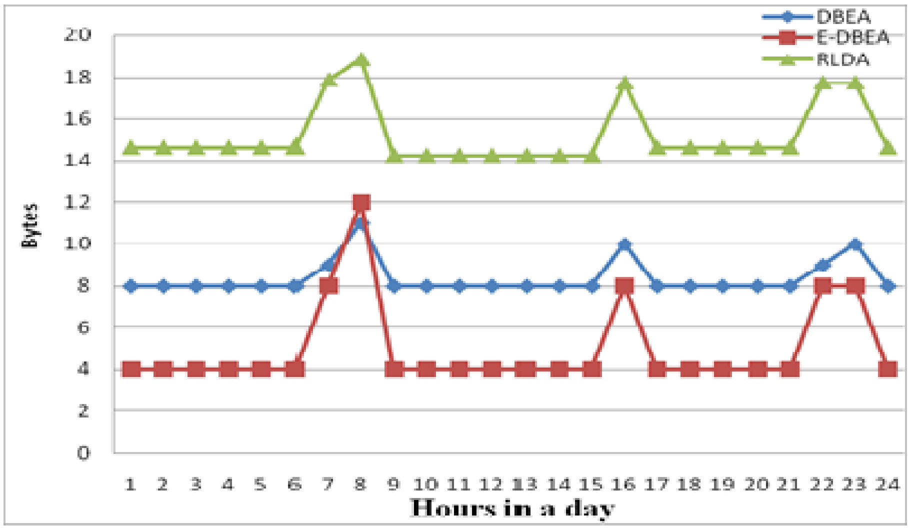
The unit commitment data associated with Barh GS [51] operating under National Thermal Power Corporation (NTPC) during July 2016 is compressed by using DBEA, E-DBEA and RLDA and the comparison is given in Figure 6.11 (a). As discussed previously, DBEA fails with some practical data sets with difference exceeding the threshold which results the discontinuity in the figure. Figure 6.11 (b) gives the comparison of the compression ratio achieved by DBEA, E-DBEA and RLDA while compressing system frequency information of 2nd December, 2016.Variation of output data size (in Bytes) obtained with DBEA, E-DBEA and RLDA for generation data of Crescent GS and Renuka GS on 2nd December, 2016 is given in Figure 6.11 (c) and 6.11 (d) respectively. It is clear from the figure that although DBEA is apparently more effective than E-DBEA, but the latter gives much better CR when large repetitive monitoring information is encountered. But for less-repetitive monitoring data, DBEA give better results than E-DBEA. It is clear from the figure that DBEA (or E-DBEA) are extremely effective techniques for compressing practical power system operational data. RLDA also gives good pretty good results with the majority of data sets, but the simplicity of DBEA and E-DBEA makes it superior than RLDA.



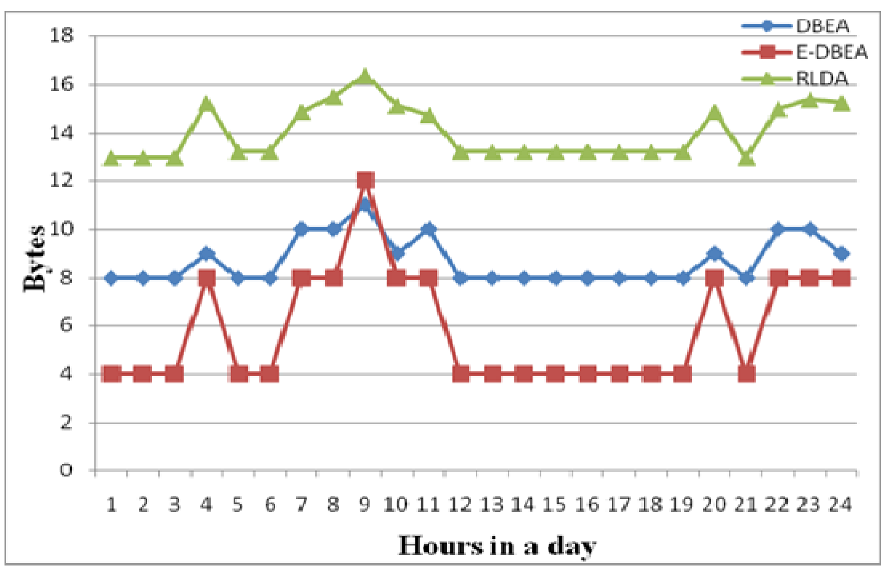
(a)



(b)



(c)



(d)

Fig. 6.11. Comparison of CR for (a) Unit commitment data and (b) Frequency monitoring data and output byte size for (c) Generation monitoring data of Crescent Power GS and (d) Generation monitoring data of Renuka GS

Entropy (Shannon entropy) or bits per symbol is also an important parameter of any data compression algorithm which is the expected value (or average) of the information contained in each message. If the probability distribution of n different symbols is p, entropy will be expressed as $-\Sigma\ p_i*\log(p_i)$ for i varying from 1 to n. In Figure 6.5 (a), the variation of CR obtained with hourly power transfer data for a 400kV transmission line is given. The entropy of the same data is computed before and after DBEA and E-DBEA and is noted in Table 6.7.

Table 6.7

Entropy before and after compression

Hours	**Block size (Bytes)**	**Entropy before compression**	**Entropy after compression**	
			DBEA	**E-DBEA**
01:00-02:00	1440	2.91	1.47	2.12
02:00-03:00	1440	2.92	1.73	2.23
03:00-04:00	1440	2.70	3.18	2.88
04:00-05:00	1440	2.42	2.79	2.82
05:00-06:00	1440	2.6	2.1	2.4
06:00-07:00	1778	3.01	1.7	2.23
07:00-08:00	2160	2.87	2.71	2.75
08:00-09:00	2160	2.9	1.76	2.25
09:00-10:00	2160	3.03	1.69	2.27
10:00-11:00	2160	2.77	1.37	2.1
11:00-12:00	2160	2.79	1.81	2.29
12:00-13:00	2160	2.88	1.54	2.19
13:00-14:00	2160	2.63	2.69	2.78
14:00-15:00	2160	2.71	2.16	2.51
15:00-16:00	2160	2.46	2.64	2.73
16:00-17:00	2160	2.83	1.3	2.08
17:00-18:00	2160	2.97	1.58	2.25
18:00-19:00	2160	2.77	1.48	2.14
19:00-20:00	2160	2.36	2.74	2.74
20:00-21:00	2160	2.83	1.73	2.31
21:00-22:00	2160	2.81	1.67	2.23
22:00-23:00	2160	2.23	2.99	2.85
23:00-24:00	2008	3.09	1.64	2.28

For Table 6.7, the average entropy of the input data before compression was 2.76, i.e., each symbol of input data can be represented by 2.76 bits. After compressing this information by DBEA and E-DBEA, the entropy of the compressed data is computed to be 2.02 and 2.41 respectively. The reduced entropy of any compression algorithm is desirable for better utilization of any communication link as for a constant baud rate; higher samples will be delivered per second when entropy is low. For a link speed of 100 baud, DBEA and E-DBEA can deliver an average of 49.49 samples per second and 41.49 samples per second respectively. If the uncompressed data could be transmitted, the average delivery rate would fall to 36.23 samples per second. Generally entropy before compression is higher than entropy after compression, but with few exceptions. This explanation will be clear by considering a couple of examples. The input of the proposed compression algorithm is a 1-D decimal array. If five decimal values, {1, 2, 4, 5, 0} are present in the input array with probability distributions of {0.333, 0.333, 0.121, 0.153, 0.06} respectively, entropy before compression will be 0.627. Suppose, the output character string obtained with DBEA comprises of 11 non-repetitive characters, i.e. probability distribution of each character is (0.09091), then entropy after compression will be 1.041 which is greater than the entropy before compression. This is the reason of considering average entropy computed before and after compression. Some exceptions were observed in Table 6.7 which may be possible when the characters contained in the output character string are large and highly non-repetitive in nature.

So far, discussions were confined to practical data arrays having a considerable extent of repetition. But obviously, it is important to check the performance of the algorithms under worst case where the achievable CR will be at its lowest limits. From the structure of the algorithm, it is clear that CR with reduction in repetition of data array due to increased number of elements in the zero count array. The results obtained with DBEA and E-DBEA while compressing two sample data array containing 10 elements is given in Table 6.8 which will be used to proceed towards the answer. Two sample, non-repetitive data array containing 96 elements are also considered and the obtained results are given in Table 6.8. It is clear from the table that the CR obtained with DBEA and E-DBEA is lower than that of entropy after compression for second and fourth sample data array. It is also visible from the first two sample data array that entropy after compression ratio varies with the type of sample data array but CR remains unchanged. The reason being that CR is the ratio of input array size and output string size, which remains unaltered if the digits of element are varied. But entropy is dependent on the probability distribution of different characters in the data array and in the output character string. This implies that entropy will vary according to the content of data array and of character string. For the first sample array, the symbols being encountered will vary between 0 and 9. On the other hand, the symbols being encountered in the second sample array will vary between 0-5 and 7. Due to this reason, entropy before compression is lower for the second array. In the first sample array, data array varies uniformly and thus the differential array will have repetitive elements. For the second sample array, differential array will have lesser repetition due to which entropy increases significantly. With the increase in the number of elements in a non-repetitive data array, it was found that the CR obtained by DBEA increases. But the CR obtained with E-DBEA remains unchanged. So, considering the minimum number of elements in the data array be 10, CR obtained with DBEA and E-DBEA is limited to 2.73 and 1.5 respectively.

Table 6.8

CR and entropy before and after compression for sample non-repetitive data array containing different number of elements

Sample data array	**Entropy before compression**	**Entropy after compression**		**CR**	
		DBEA	**E-DBEA**	**DBEA**	**E-DBEA**
[100, 110, 120, 130, ……. 190]	2.37	0.866	1.469	2.73	1.5
[100, 105, 102, 125, 135, 137, 124, 110, 100, 105]	2.347	3.278	3.141	2.73	1.5
[100, 110, 120, ….. 1040, 1050]	2.917	0.166	1.084	3.03	1.53
[100, 101, 103, 106, ….. 4565, 4660]	3.197	5.816	4.293	3.53	1.78

The variation of CR obtained with DBEA and E-DBEA while compressing non-repetitive data array of varying length is given in Table 6.9.

Table 6.9

CR obtained with small non-repetitive data array of varying array length

Sl. No.	**Array length**	**CR**	
		DBEA	**E-DBEA**
1	10	2.727	1.5
2	20	2.857	1.5
3	30	2.903	1.5

6.4. Performance analysis of large data array compression by Basic Arithmetic and Huffman Coding based algorithm

The results given in Table 6.10 are obtained when scheduling information containing 1, 2 and 3-digit elements are compressed by both Basic Arithmetic Coding and Huffman Coding based algorithm separately. It is obvious that the higher compression ratio can be achieved only when 3-digit elements are present in the data array. But for 1-digit element, compression ratio will be at its minimum and can be less than unity for Huffman Coding. It implies that size of input data array will be lower than the size of encrypted data string.

Table 6.10

CR obtained with two algorithms while compressing scheduling array containing 1, 2 and 3-digit elements

Type of algorithm	1 digit element	2 digit element	3 digit element
Basic Arithmetic Coding	0.635	1.262	1.905
Huffman Coding	1.11	2.229	3.343

Analysis of practical generation scheduling data of KTPS clearly indicates that it is primarily 3-digit data array. This implies that a constant compression ratio is obtained with those data by both the algorithms. Bandel Thermal Power Station (BTPS) is among the oldest generating stations of Eastern India and have higher incremental fuel cost in comparison to other advanced generating stations. It was also observed that BTPS have the highest failure rate during duration March 2015 to February 2016.The scheduling information of BTPS during 7th February, 2016 to 16th February, 2016 is compressed by proposed Entropy based compression algorithms and the obtained results are given in Table 6.11. The variation of CR obtained with the proposed algorithms while compressing the unit commitment information of KTPS during 1st February, 2018 and 15th February, 2018 is given in Figure 6.12.

Table 6.11

CR obtained with proposed algorithms while compressing scheduling array of BTPS

Sl. No.	Date	Block size (Bytes)	Huffman Coding	Basic Arithmetic Coding
1	07/02/16	288	1.905	3.343
2	08/02/16	286	1.877	3.32
3	09/02/16	192	1.262	2.229
4	10/02/16	192	1.262	2.229
5	11/02/16	262	1.72	3.042
6	12/02/16	288	1.905	3.343
7	13/02/16	252	1.654	2.925
8	14/02/16	192	1.262	2.229
9	15/02/16	268	1.753	3.099
10	16/02/16	288	1.905	3.343

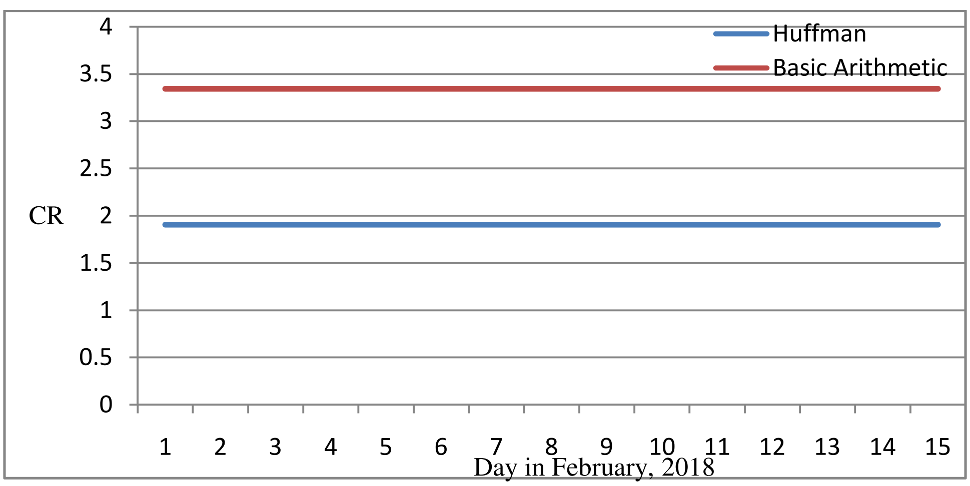


Fig. 6.12. Variation of CR obtained with Huffman and Basic Arithmetic based compression algorithm

It is obvious that compression of parameter monitoring data is of much higher importance than compressing scheduling data. It is due to the fact that conventional SCADA can acquire data value typically at every 5 seconds. It implies that the hourly data array will have 720 elements in it. Complete analysis of the performance of the proposed algorithms is only possible when such bulk volume of information can be compressed. To serve the purpose, hourly MW generation data of India Power and Haldia Energy Limited over an entire day is compressed by the proposed algorithms and the obtained results are given in Figure 6.13 (a) and 6.13 (b) respectively. While compressing power transfer data of Berhampur- Jirat 400kV line over an entire day, the variation of CR is given in Figure 6.13 (c). The successful testing of the proposed algorithms clearly indicates its effectiveness for compressing very large data arrays.

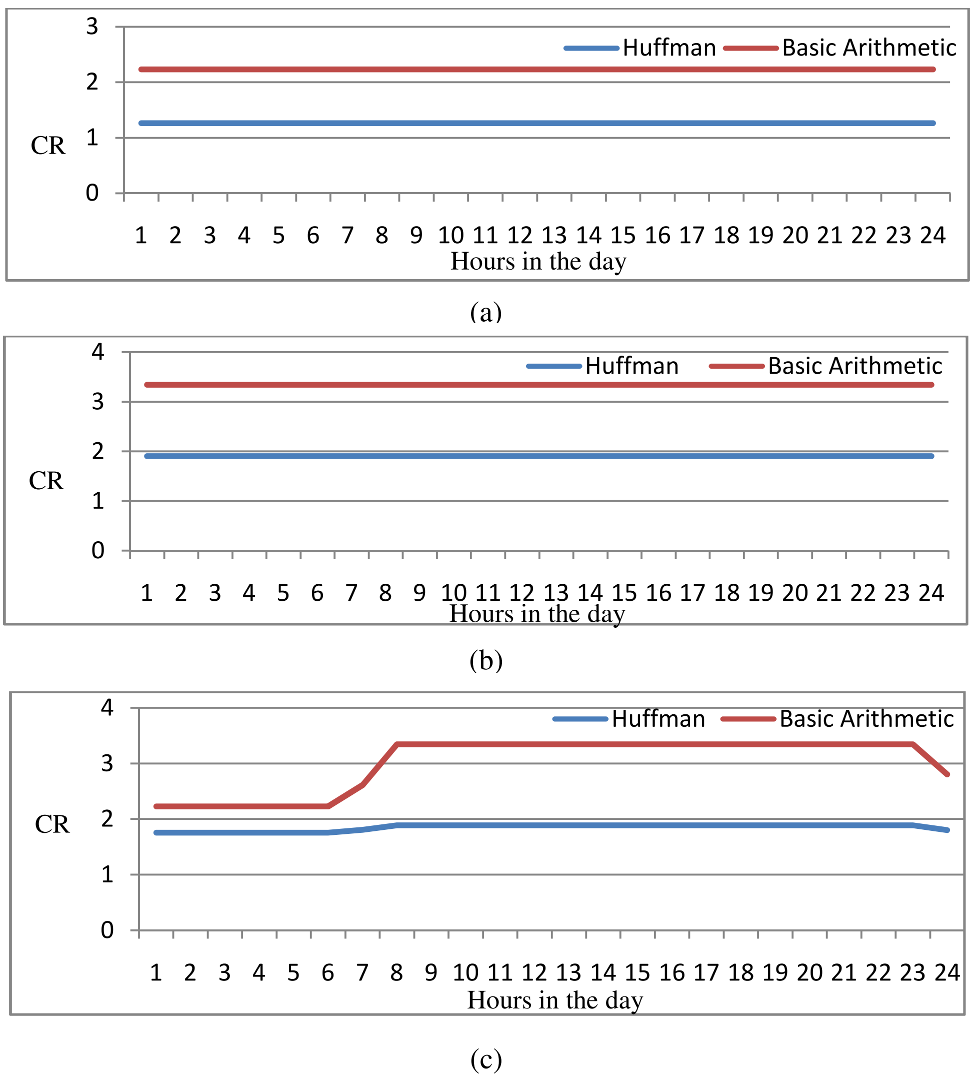


Fig. 6.13. CR obtained with hourly monitoring data: (a) For India Power generation; (b) HEL Power generation and (c) Power transfer data of Berhampur- Jirat 400kV line

6.5. Development of secured large data array compression algorithms

6.5.1. RSA encrypted DBEA compression scheme

Data encryption plays a critical role in communication to maintain the security during data transmission. Due to the simplicity of DBEA (or E-DBEA), it becomes necessary to incorporate suitable data encryption algorithm followed by compression before transferring the information through the suitable communication channel. The block diagram of the system will be as given in Figure 6.14.

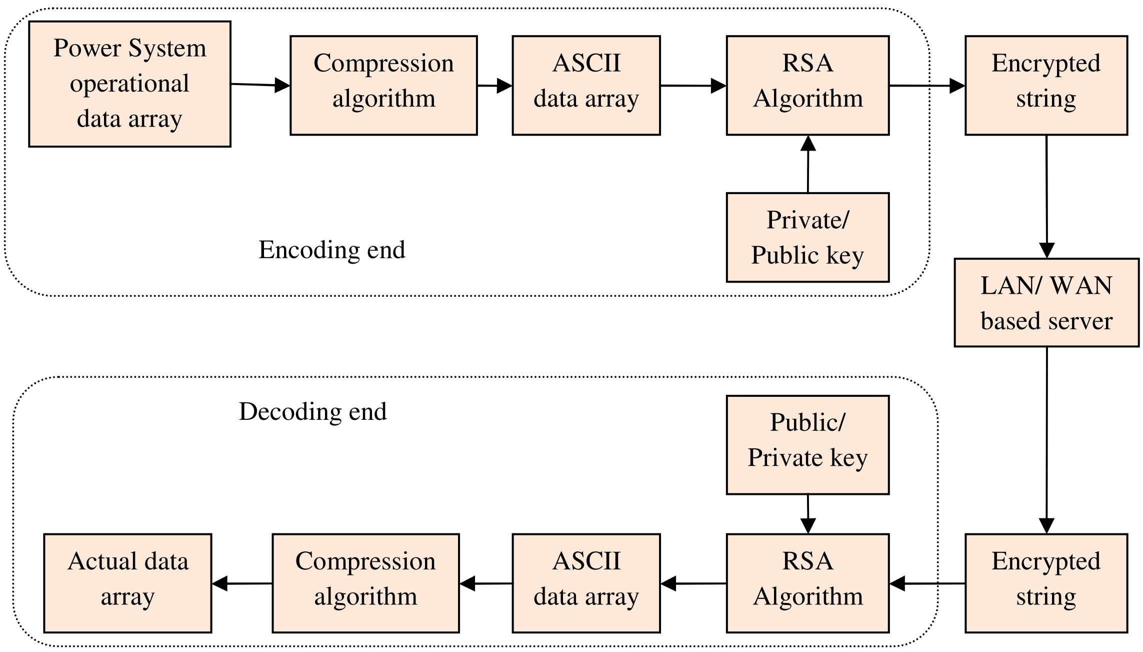


Fig. 6.14. Block diagram of a compressed, encrypted data management system

Though there are several available data encryption algorithms, the selection of the proper algorithm should be done judiciously as the following conditions can be satisfied.

i. Security can be ensured if separate keys are used in encoding and decoding end.
ii. The output of DBEA (or E-DBEA) is a character string containing ASCII or extended ASCII characters. The security algorithm must encrypt the information, suitably such that the output will be an encrypted character string of ASCII or extended ASCII characters.

Based on these observations, it was found that RSA algorithm can be a good choice for such requirements. As it employs separate keys at encoding and decoding end, the first condition can be met. In RSA algorithm, a number between 0 and the modulus value (n) and be encrypted to another number between 0 and n. This implies that proper selection of n can satisfy the second condition as well. This is the reason behind the selection of the RSA algorithm for encrypting the character string obtained with DBEA or E-DBEA.

6.5.1.1. Data Security and Cryptography: RSA Algorithm

The science and study of secret writing is termed as cryptography where a plaintext (a human readable message) is transformed into ciphertext (encrypted or encoded information). Encipherment or encryption is the process of transforming plaintext into ciphertext. The reverse process of transforming ciphertext into plaintext is called decipherment or decryption. Both encipherment and decipherment are controlled by a cryptographic key or keys. On the other hand, data security is the science and study of methods of protecting data in computer and communications systems. Cryptography can be an important tool for providing data security. Modern cryptography protects data transmitted over high-speed electronic lines or stored in computer systems to serve the following two objectives [125].

i. Secrecy or privacy: To prevent the unauthorized disclosure of data
ii. Authenticity or integrity: To prevent the unauthorized modification of data

Cryptographic algorithms can be classified according to the number of keys that are employed for encryption and decryption as [126]:

i. Secret Key Cryptography: This method employs a single key for both encryption and decryption. The sender uses the key to encrypt the plaintext and sends the ciphertext to the receiver. The receiver applies the same key to decrypt the message and recover the plaintext. As a single key is used for both encryption and decryption, it is also called symmetric encryption.
ii. Private Key Cryptography: In this method, two different keys, one key for encryption and another for decryption are used. As different keys are used for encryption and decryption, it is called asymmetric encryption. It is primarily used for authentication, non-repudiation, and key exchange.
iii. Hash functions: Hash functions (also called message digests and one-way encryption) are the algorithms which do not use any key. According to the plaintext, a fixed-length hash value is computed by the algorithm. So, it becomes impossible to recover the contents or length of the plaintext. These algorithms are typically used to provide a digital fingerprint of a file's contents, used often to ensure that the file has not been altered by an intruder

or virus. They are also commonly employed by many operating systems to encrypt passwords. Hash functions, then, provide a mechanism to ensure the integrity of a file.

In symmetric encryption, a single key is used by the encryption and decryption algorithm. In these encryption techniques, security depends only on the secrecy of the key, not on the secrecy of the algorithm. But in asymmetric algorithms, different keys are used in encrypting and decrypting end such that the decryption key can be determined only if the encryption key and cryptographic algorithm are known. In some asymmetric algorithms like RSA algorithm, either of the key can be used for encryption, while the other key is used for decryption. Public key encryption scheme comprises of plaintext, encryption and decryption algorithm, public and private keys and ciphertext. Plaintext is the readable data or message fed to the algorithm as an input. On the other hand, ciphertext is the output scrambled message which is a function of the plaintext and key. It implies that the content of the output will be different for different key. In asymmetric encryption, a set of keys, namely private and public key is used at the both ends. Private Key is kept confidential by the user and the public key is shared by the user with the other users. The basic block diagram of any public key cryptography is given in Figure 6.15 [127, 128].

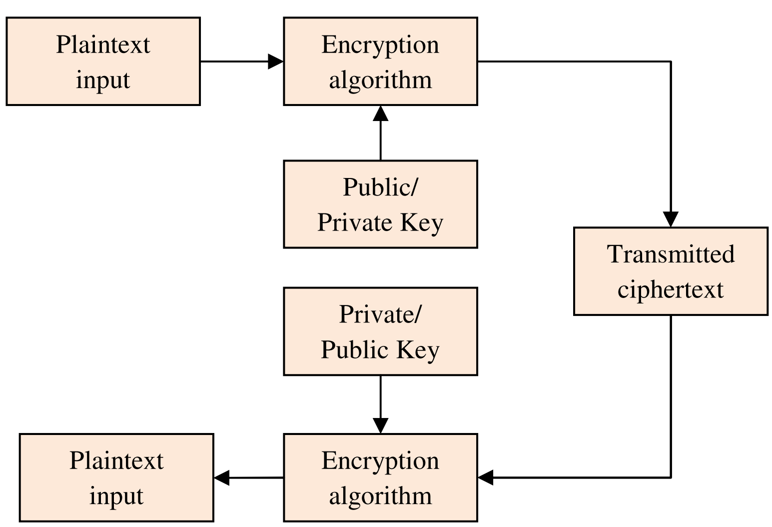


Fig. 6.15. A Public Key Cryptography system

In RSA scheme, plaintext and ciphertext are integer values varying between 0 and n-1 for some value of n. For some plaintext block M and ciphertext block C, encryption and decryption can be done by performing relations C= M^e mod n and M= C^d mod n. This clearly indicates that the value of n should both at sending and receiving end. The sender should know the value of e whereas the receiver requires the value of d. This implies that the public key and private key can be given as PU= {e, n} and PR= {d, n} respectively. The keys can be generated from two selected prime numbers following the given pseudo code [127].

Input: p, q
Output: PU, PR

```
Begin
        n= p*q
        Φ= (p-1)*(q-1)
        Select a prime number e such that 1< e< Φ
        The relation d*e= 1 (mod Φ) must be satisfied
        d is computed by using Extended Euclidean Algorithm's Table method
        PU= {e, n}
        PR= {d, n}
End
```

After determination of public and private key, encryption and decryption of any plaintext and ciphertext block is possible by solving the relation C= M^e mod n and M= C^d mod n by using modular exponentiation discussed below [127, 129].

Input: M (or C); PU (or PR)
Output: C (or M)

```
Begin
        bin(1, l)= Binary equivalent of e (or d)
        Initialize result= 1, base= M (or C) and modu= n
        Initialize i= l
        Repeat until i > 0
                If bin (1, i) ==1
                        result= (result*base) modulus modu
                end
```

```
            base= (base* base) modulus modu
            i= i-1
      end
      C= result
End
```

6.5.1.2. RSA encrypted DBEA Algorithm

The output of DBEA is a character string corresponds to the ASCII data array obtained after performing binary encoding. In RSA encryption, both plaintext and ciphertext are integer values and thus it is possible to encrypt the elements present in the ASCII data array before forming the character string. This not only reduces system complexity, but also includes the advantages of asymmetric encryption after compressing any large data array significantly. The steps followed in the proposed RSA secured DBEA system is given in Figure 6.16 [129].

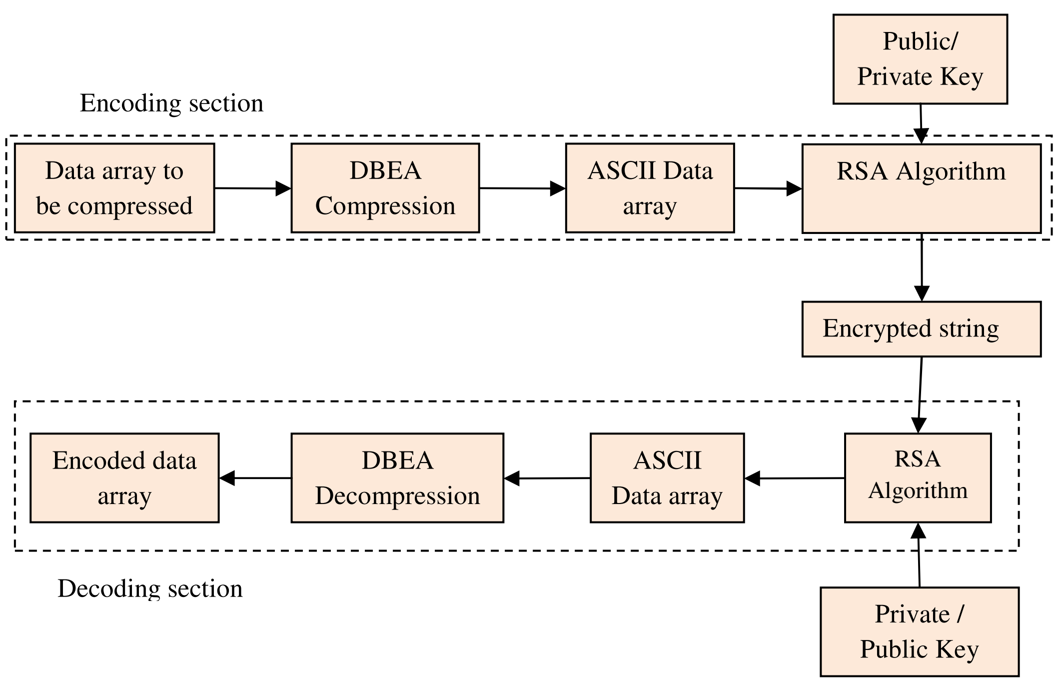


Fig. 6.16. Block diagram of RSA encrypted DBEA algorithm

The pseudo code for forming an encrypted string from a large data array is given below [129].

Input- arr[l]; PU (or PR)

```
Output- str: Encrypted character string

Begin
        asci[p] (p≤l) corresponds to arr[l] is obtained by using DBEA compression scheme.
        Initialize i= 1
        Define str= NULL
        Repeat until i>p+1
                enc= RSA encrypted number corresponds to asci[i]
                ch= Character equivalent of enc
                str= Concatenate str and ch
                i= i+1
        end
End
```

The actual data array is decrypted from the encrypted string by following the given pseudo code.

```
Input- str; PR (or PU)
Output- arr[p]

Begin
        l= Length(str)
        Initialize i= 1
        Define asci= NULL
        Repeat until i>l+1
                ch= str(i)
                dec= ASCII equivalent of ch
                asci[i]= dec
                i= i+1
end
        Determine arr[p] (p>l) corresponds to asci[l] using DBEA decompression
End
```

While testing the proposed DBEA compressed, RSA encrypted algorithm with a sample data, the various steps followed during encoding are illustrated in Figure 6.17 (a). At the decoding end, the encrypted string is converted to the corresponding ASCII values which are then decrypted by using RSA algorithm. DBEA is applied to the resultant array to obtain the actual data array. The steps followed at the decoding end are illustrated in Figure 6.17 (b). The result clearly indicates that a data array containing 30 characters is compressed in merely 6 characters by virtue of DBEA compression scheme. The characters are also in encrypted form so that data security can be ensured during compressed data transfer.

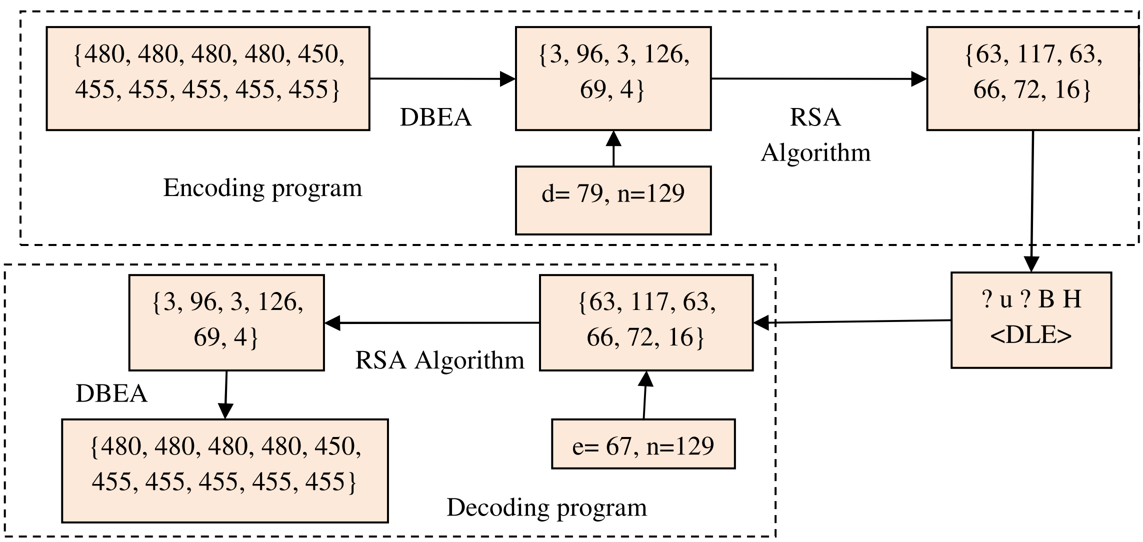


Fig. 6.17. RSA secured DBEA compression scheme for compressing a data array

6.5.2. Combined DBEA Encrypted (C-DBEAE) compression scheme

The simplicity of DBEA and E-DBEA encourages the development of a combined algorithm where DBEA or E-DBEA is selected according to the data array. This will result in the highest possible CR while compressing power system operational data. In this adaptive algorithm, differential array corresponds to the input array is obtained and based on its elements, DBEA or E-DBEA is selected for compressing that data array. The respective functions are responsible to compress the data array to a character string. For decoding such compressed string, it is obvious that some identity information must be included in the string to distinguish between DBEA and E-DBEA. In C-DBEAE, data compression is followed by encrypting the information by using RSA algorithm to form an encrypted character string. In this work, a compressed, encrypted data transfer system is realized with PCs at either ends as given by the block diagram in Figure 6.18. At the encoding end, data array associated with power system is compressed and encrypted to form a character string at discrete time intervals. This encrypted string will be stored in the cloud and is accessed by the PC at receiving end. It will continuously check for any new received file and decode the file only when any new file is obtained. This feature will enable to operate the system with the asynchronous data transfer.

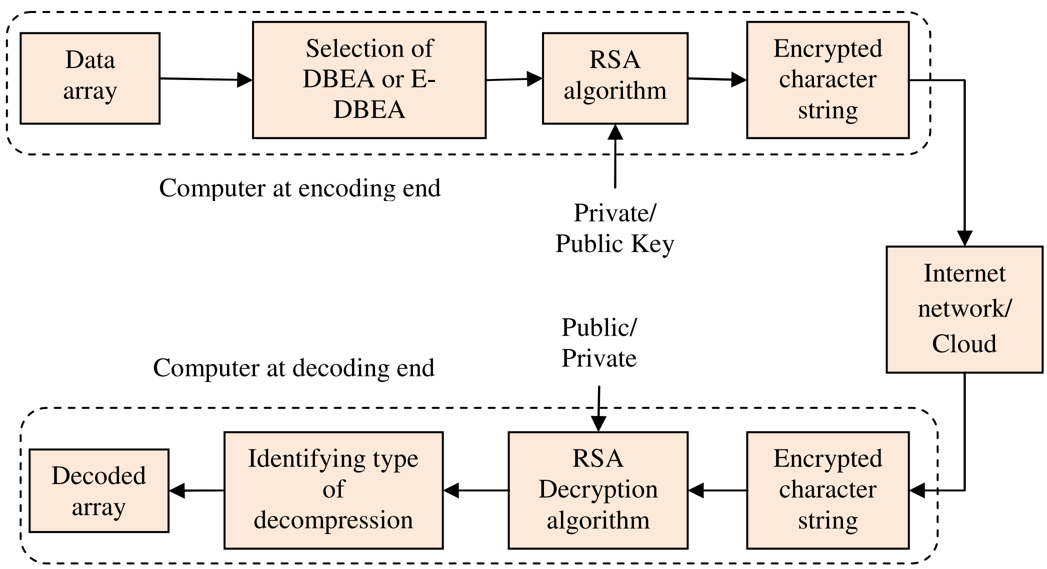


Fig. 6.18. Block diagram of cloud based encrypted data transfer system

The pseudo code for compressing a large data array to an encrypted character string using C-DBEAE is described below.

Input- arr(l); PU (or PR)

Output- str

Begin

    Define zero difference array, dif(l)

    dif(1)= arr (1)

    Initialize i= 2 and status= 0

    Repeat until i< l+1

        dif(i)= arr(i) – arr(i-1)

        If modulus(dif(i))< 30 and status= 0

```
            status= 1
        Otherwise
            status= 2 when status ≠ 2
        end
        i= i+1
    end
    Define zero ASCII array asci and encrypted array enc
    If status= 1,
        enc(1)= 68
        Determine asci by DBEA compression scheme
    Else if status= 2
        enc(1)= 69
        Determine asci by E-DBEA compression scheme
    end
    len=Length(asci)
    Initialize j= 2
    Repeat until j< len+1
        enc(j)= Encrypted number corresponds to arr (j-1) and PU (or PR)
        j= j+1
    end
    str=NULL
    Initialize k= 1
    Repeat until k< len+1
        ch= Character equivalent of enc (j)
        str= Concatenate str and ch
        k=k+1
    end
End
```

At receiving end, received string is decrypted initially by RSA to obtain an actual ASCII array. This array is then decompressed either by DBEA or E-DBEA to obtain actual data array. The pseudo code for extracting a large data array from an encrypted character string using C-DBEAE is given below.

```
Input- str; PR (or PU)
Output- act(l)

Begin
        l= Length(str)
        Initialize i= 1
        Repeat until i<l+1
                asci(i)= ASCII value corresponds to ith character of str
        end
        If asci(1)==68
                Initialize j= 2
                Repeat until j<l+1
                        decr(j-1)= Decrypted number corresponds to asci(j) and PR/ PU
                        j=j+1
                end
                act(l)= DBEA decompressed decr[]
        Else if asci(1)==6
                Initialize j= 2
                Repeat until j<l+1
                        decr(j-1)= Decrypted number corresponds to asci(j) and PR/PU
                        j=j+1
                end
                act(l)= E-DBEA decompressed decr[]
        end
End
```

6.5.2.1. Experimental setup

File sharing between two computers connected via Local Area Network (LAN) is simple, but extending the same for Wide Area Network (WAN) is a challenging task. Development of dedicated client- server system for online testing of the algorithm not only increases system cost, but also introduces traffic congestion when a large number of clients send requests to server. In contrast, cloud based services are not only available at lower cost, but are also more reliable and consistent than client- server system. For online testing of the developed prototype, Dropbox is employed to share encrypted files between two computers connected through internet. Dropbox is a file hosting service operated by American company Dropbox, Inc. It is based on freemium business model, where users are offered a free account with a set storage size, with paid subscriptions available that offer more capacity and additional features. Initially, Amazon's S3 storage system was used by Dropbox to store user files. But between 2014 and 2016, they gradually moved away from Amazon to use their own hardware, "Magic Pocket". Advanced Encryption Standard (AES) 256 bit encryption is used for synchronization and data storage in Dropbox [130].

This implies that encrypted file will be stored in the cloud and individual systems will act like clients which either update the file (Encoding end) or decrypt and decode the file (Decoding end). The data array to be compressed at the encoding end will select DBEA or E-DBEA according to the content of difference array and is encrypted by RSA to form character string and is stored as a text file in Dropbox. When system at the decoding end access same Web or server, text file stored in respective Dropbox will be decoded to obtain actual data array only when any updated file is obtained. If test file is not updated, decoding program will not decode the string and “No change” will be displayed in the MATLAB command window. In case of no internet connectivity at encoding or decoding end, updating of the file will not be possible and previous string will be loaded again and again. Hardware realization for online test bench with two computers connected through internet is given in Figure 6.19.

(a)

(b)

Fig. 6.19. Hardware realization of cloud based compressed encrypted data transfer system (a) at encoding end and (b) at decoding end

6.5.2.2. Obtained results

In this proposed algorithm, based on the difference between consecutive data value, DBEA or E-DBEA is selected and an additional character is included before the encrypted string based on which suitable decoding algorithm can be selected at decoding end. This implies that the compression ratio achieved by the proposed algorithm will be slightly lower than that obtained by DBEA where slow varying data array is encountered. But data arrays with abrupt changes, the compression ratio will be even below that achieved by E-DBEA. Adaptability is the advantage of the proposed algorithm which enables its application for compressing any power system operational data and average compression ratio is achieved by C-DBEAE is much higher than that obtained with E-DBEA. Variable length normalization is also included in this combined algorithm to make it suitable flexible. Unit commitment information of two different generating stations operating under ERLDC during 1^{st} – 15^{th} July, 2016 is compressed by C-DBEAE and the obtained result is given in Table 6.12.

Table 6.12

Results obtained with C-DBEAE while compressing unit commitment data

Date	**Teesta**			**FSTPP 3**		
	Actual size (Bytes)	**Encrypted string size (Bytes)**	**CR**	**Actual size (Bytes)**	**Encrypted string size (Bytes)**	**CR**
01/07/16	384	9	42.67	384	9	42.67
02/07/16	384	9	42.67	384	5	76.8
03/07/16	384	9	42.67	384	5	76.8
04/07/16	384	9	42.67	384	5	76.8
05/07/16	279	39	7.15	384	5	76.8
06/07/16	96	5	19.2	384	9	42.67
07/07/16	248	23	10.78	384	9	42.67
08/07/16	384	9	42.67	384	5	76.8
09/07/16	384	13	29.54	384	5	76.8
10/07/16	384	13	29.54	384	5	76.8
11/07/16	384	13	29.54	384	5	76.8
12/07/16	384	13	29.54	384	5	76.8
13/07/16	384	13	29.54	384	5	76.8
14/07/16	384	15	25.6	384	5	76.8
15/07/16	384	15	25.6	384	5	76.8

It is clearly visible that the achievable CR is lower for a Hydel generating station (Teesta) in comparison to a super-thermal power plant (FSTPP 3). Average compression ratio obtained Teesta hydel plant and FSTPP 3 during 01/07/16 and 15/07/16 are 29.96 and 69.97 respectively. This is due to the fact that generation of any hydel plant is dependent on water availability which is highly unpredictable in nature and is dependent on rainfall. So, there are increasing number of changes in final unit commitment information and thereby having a lower compression ratio. If there is some sudden change in the generation, E-DBEA will be employed and compression ratio reduces further. Generally, thermal power plant serves as base load generating stations and their generation do not vary widely. Due to reduced number of changes in data array, higher compression ratio is achieved with unit commitment information of FSTPP 3. Comparative analysis of compression ratio obtained with DBEA, E-DBEA and proposed algorithm while compressing unit commitment information of Barh super thermal generating station over the entire month of July, 2016 is given in Figure 6.20. It is clear from the figure that C-DBEAE gives better results than E-DBEA, except few cases and is more consistent than DBEA which fails for certain data sets.

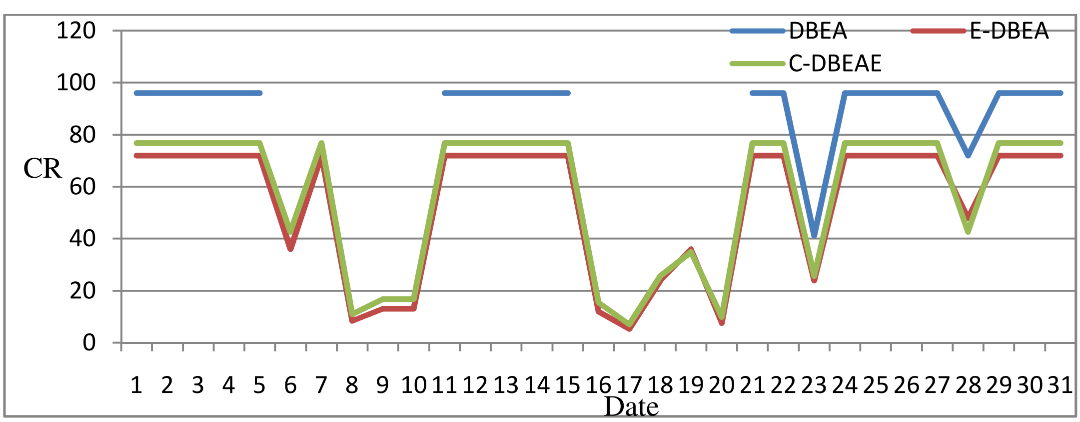


Fig. 6.20. Comparison between CR obtained with DBEA, E-DBEA and C-DBEAE of unit commitment information of Barh Super thermal generating station during July, 2016

In comparison to unit commitment data, system monitoring data comprise of a much larger volume. To justify the effectiveness of C-DBEAE for compressing such bulk volume, it becomes important to test the algorithm with that data as well. While compressing 15 minute time block practical system frequency data and power generation data of the Rammam Hydroelectric project, the obtained results given in Figure 6.21. Though DBEA is employed for compressing both data arrays, a lower CR is obtained with frequency monitoring data. This is due to high fluctuations occurring in such data. In contrast, power output of the 50MW generator of hydel generator will remain constant for the entire day (96 time blocks). In comparison to Teesta hydel generation data, CR being obtained while compressing Rammam Hydroelectric project is much higher. The reason is the increased fluctuations occurring in Teesta generation data which results in a significant drop in CR when repetition in data array reduces.

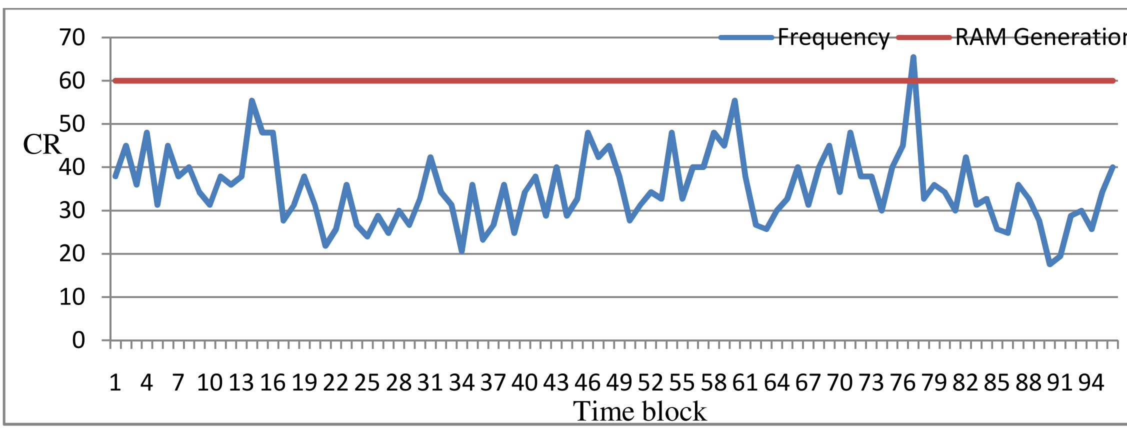


Fig. 6.21. CR obtained with frequency monitoring and hydel power generation data over an entire day

The average string length obtained with C-DBEAE over 96 time blocks for frequency and power transfer data are 21.7 and 26.4 respectively. It implies that compression ratio will be higher for system frequency monitoring information. This is obvious as system frequency tends to vary within a very small range after application of Availability Based Tariff (ABT) in the Indian power system and there is a large repetition in the data array. For any data compression algorithm, Entropy (Shannon entropy) or bits per symbol is very important parameter which is the expected value (or average) of the information contained in each message. It is defined as the negative of the logarithm of the probability distribution of the possible events or messages. Comparison of entropy obtained with DBEA, E-DBEA and C-DBEAE while compressing hourly power transfer data of Berhampur- Jirat 400kV line over an entire day is given in Table 6.13.

Table 6.13

Entropy before and after compression of hourly power transfer data of Berhampur- Jirat 400kV line

Time block (in hrs)	**Block size (Bytes)**	**Entropy before compression**	**Entropy after compression**		
			DBEA	**E-DBEA**	**C-DBEAE**
01-02	1440	2.91	1.47	2.12	1.475
02-03	1440	2.92	1.73	2.23	1.73
03-04	1440	2.70	3.18	2.88	3.179
04-05	1440	2.42	2.79	2.82	2.786
05-06	1440	2.6	2.1	2.4	2.1
06-07	1778	3.01	1.7	2.23	1.697
07-08	2160	2.87	2.71	2.75	2.706
08-09	2160	2.9	1.76	2.25	1.755
09-10	2160	3.03	1.69	2.27	1.632
10-11	2160	2.77	1.37	2.1	1.373
11-12	2160	2.79	1.81	2.29	1.813
12-13	2160	2.88	1.54	2.19	1.539
13-14	2160	2.63	2.69	2.78	2.691
14-15	2160	2.71	2.16	2.51	2.16
15-16	2160	2.46	2.64	2.73	2.636
16-17	2160	2.83	1.3	2.08	1.221
17-18	2160	2.97	1.58	2.25	1.58
18-19	2160	2.77	1.48	2.14	1.482
19-20	2160	2.36	2.74	2.74	2.74
20-21	2160	2.83	1.73	2.31	1.734
21-22	2160	2.81	1.67	2.23	1.671
22-23	2160	2.23	2.99	2.85	2.992
23-00	2008	3.09	1.64	2.28	1.64

Comparison of compression ratio achieved by DBEA, E-DBEA and C-DBEAE while compressing hourly power transfer data of Berhampur- Jirat 400kV line over an entire day is given in Figure 6.22. This clearly indicates that the proposed algorithm gives better result than E-DBEA for parameter monitoring data as well.

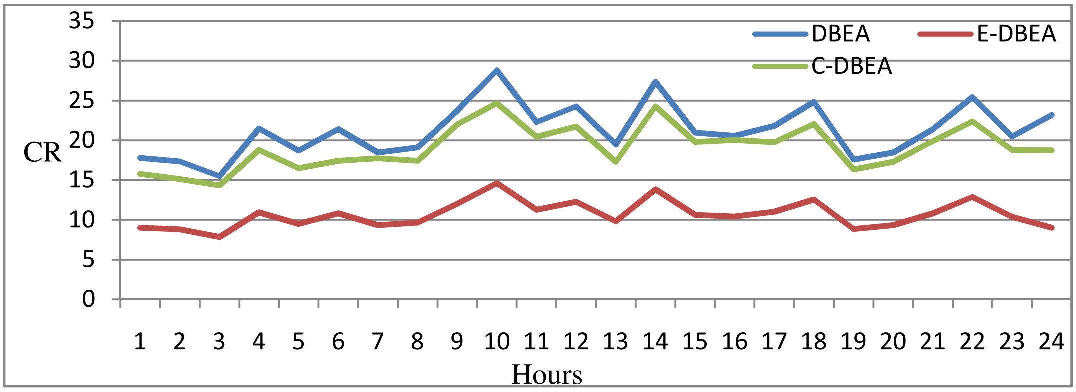


Fig. 6.22. Comparison of CR obtained by DBEA, E-DBEA and C-DBEAE for power system data transfer data over an entire day

6.6. Power System operational data management using Smart DBEA Encrypted (S-DBEAE) algorithm

In C-DBEAE, an attempt was made to compress power system operational data by DBEA or E-DBEA, selected according to the data array and the information is encrypted with the RSA algorithm. This enables C-DBEAE to achieve highest possible CR from a given set of data arrays. Due to encrypted information exchange between two PCs connected through internet communication will be secured. Though the inclusion of the additional identity bits decreases its performance with respect to DBEA, but there is a significant improvement in the data handling capacity. But the developed test bench was meant for real time testing of the algorithm between the two PCs only. But while approaching towards practical system, it is highly probable that the system has to deal with several versions of data and all such versions should be stored for future references. The compressed, encrypted string obtained with C-DBEAE can't take care of these situations and thereby demanding a much smarter algorithm. Smart-DBEA Encrypted (S-DBEAE) [131] algorithm is a good alternative to handle such cases which also possesses the features of C-DBEAE. By using S-DBEAE, it is possible to store multiple compressed, encrypted character strings corresponds to different versions of the data array in a single file and extract precisely one particular data array from the stored chunk of compressed, encrypted data. The CR obtained with S-DBEAE will be lower than that obtained with C-DBEAE as multiple identity characters are included before the encrypted string to extract the information at decoding end.

6.6.1. Smart DBEA Encrypted (S-DBEAE) compression algorithm

Analysis of practical generation scheduling data indicates that few additional parameters should incorporate with the compressed, encrypted string to extract a particular data array from a large set of data array present in the cloud in the form of a compressed, encrypted file. The parameters considered during the development of S-DBEAE are:

i. Date information: It is a known fact that date can be any integer between 1 and 31 i.e. it can be represented as 5-bit information.

ii. Month information: According to the calendar, month will be an integer between 1 and 12 i.e. it can be represented as 4-bit information.

iii. Normalization information: Many practical data array can have float numbers. For compressing such data array normalization is followed by differential coding. At the decoding end, this information must be provided to obtain the float array. In the proposed system, it is possible to take care of the fractional array correct up to 3 decimal places. This implies that this information can be represented as 2-bit information.

iv. Version information: Any practical scheduling information will be in multiple versions typically varying between 5 and 25. This information is also contained in S-DBEAE in the form of 7-bit information, i.e. the system can handle maximum up to 127 versions.

v. Encrypted string length information: For extracting the exact encrypted string from the large text file, information regarding the string length is also included with the encrypted string. In S-DBEAE, this will be 7-bit information, i.e. the system can handle an encrypted string containing maximum up to 127 characters.

vi. Data type information: Scheduling data for a GS can be either next day availability information or unit commitment information. Based on the content of a single bit, this information is also extracted at decoding end. For data associated with DC, data type can be either next day forecasting information or next day schedule information which again can be distinguished by 1-bit information.

vii. Compression type information: In C-DBEAE, distinction between DBEA and E-DBEA is done according to the first character of the string output. In S-DBEAE, multiple information is required for decoding and thus a single bit is used to differentiate between DBEA or E-DBEA at the decoding end.

Summarising the discussions, it is clear that S-DBEAE can be considered as a C-DBEAE with three additional identity bits. The combination of different information included before the compressed, encrypted character string while compressing generation scheduling information is given in Table 6.14.

Table 6.14

Identity character fields while compressing generation scheduling information

Identity character	No. of bits	Fields considered (No. of bits)
First	7	Date (5) Round (2)
Second	7	Month (4 bits) Type of data (2) Type of compression(1)
Third	7	Version (7)
Fourth	7	Encrypted string length (7)

For compressing parameter monitoring information, a different information pattern is followed. Different parameters considered during the development of S-DBEAE for compressing parameter monitoring data are given below. The combination of different information included before the compressed, encrypted character string while dealing with parameter monitoring information is given in Table 6.15.

i. Date information: Obviously, this information can be represented as 5-bit information as considered in the previous case.

ii. Month information: As discussed previously, this information can be represented as 4-bit information.

iii. Time block information: In M-DBEA parameter monitoring information collected over a finite time interval is compressed and the compressed information is transmitted over the suitable communication channel. In S-DBEAE, parameter monitoring information collected over a time block (of 15 minutes duration) is compressed and is updated in the cloud. There are 96 time blocks in an entire day and hence 7-bit information is required to represent the information.

iv. Type of data information: In the present form, three different types of data array, power generation data array, power transfer data array and frequency monitoring data array are considered. This information can be represented by 2-bit of information.

v. Polarity of first element: Analysis of power transfer data indicates that the elements such data array can be negative. When the first element of any data array is negative, neither DBEA nor E-DBEA can compress such data array. This problem can be solved by multiplying the entire array with -1. In order to handle such case, one bit is included to distinguish the polarity of first element.

vi. Type of compression information: To distinguish between DBEA or E-DBEA, one bit is included with the encrypted information.

vii. Encrypted string length information: Information regarding the string length is also included with the encrypted string in the form of 7-bit information.

Table 6.15

Identity character fields while compressing parameter monitoring information

Identity character	**No. of bits**	**Fields considered (No. of bits)**
First	7	Date (5) Type of data (2)
Second	7	Month (4) Type of compression (1) Polarity of first element (1)
Third	7	Time block (7)
Fourth	7	Encrypted string length (7)

The pseudo code followed at encoding end in S-DBEAE scheme for power system monitoring data management is given below.

Input: arr[l]; Last updated file (txt); PU (or PR); Version (vr); Type of data (td); data normalization (round); Time block (tb)
Output: Updated file (utxt)

```
Begin
        Multiply each element of arr[l] with 10^round and round it to nearest integer
        According to arr[l], DBEA or E-DBEA is selected by setting or resetting field tc
        If arr[1]<0
                pol=1
                arr= Multiply each element of arr[] by -1
        Otherwise
                pol=0
        end
        iftc==1
                asci[p]= Compression of arr[l] by using DBEA
        else if tc==0
                asci[p]= Compression of arr[l] by using E-DBEA
        end
        Initialize i=1
        encr=NULL
        Repeat until i<p+1
                num= Encrypted number corresponds to asci (i) and PU (or PR)
                ch=Character equivalent of num
                encr= Concatenate encr and ch
        end
        Read date (dt) and month (mnt) information from system clock
        Based on dt, td, mnt, tc, pol, tb and p, four identity characters is determined
        menc= Concatenate identity characters with encr
        Load txt from cloud
        utxt= Concatenate txt and menc
End
```

At decoding end, the actual monitoring array is obtained by following the pseudo code is given below.

```
Input: utxt; PR (or PU); dt; mnt; tb; td
Output: arr(l)

Begin
        First and third identity characters are computed
        utxt is loaded and these characters were searched
        Based on mnt information, the selection is more specific
        If found
                Determine tc and pol information from the second character
                p= ASCII value of fourth identity character
                str= String of p characters starting from fifth character of utxt
                enc(p)= ASCII array corresponds to str
                asci(p)= Decrypt enc[p] element wise by using PR (or PU)
                If tc==1
                        arr[l]= Decompress asci(p) by using DBEA
                If tc==0
                        arr[l]= Decompress asci(p) by using E-DBEA
                end
                If pol==1
                        arr[l]= Multiply each element of arr[] by -1
                end
        Otherwise
                Stop execution
        end
End
```

For the management of generation scheduling data, the process will be similar, except the fields considered in the formation of identity elements. The necessary pseudo code for managing generation scheduling at the encoding end is given below.

Input: arr[l]; Last updated file (txt); PU (or PR); Version (vr); Type of data (td); data normalization (round); Time block (tb)

Output: Updated file (utxt)

```
Begin
    Multiply each element of arr[l] with 10^round and round it to nearest integer
    According to arr[l], DBEA or E-DBEA is selected by setting or resetting field tc
    iftc==1
        asci[p]= Compression of arr[l] by using DBEA
    else if tc==0
        asci[p]= Compression of arr[l] by using E-DBEA
    end
    Initialize i=1
    encr=NULL
    Repeat until i<p+1
        num= Encrypted number corresponds to asci (i) and PU (or PR)
        ch=Character equivalent of num
        encr= Concatenate encr and ch
    end
    Read date (dt) and month (mnt) information from system clock
    Based on dt, td, mnt, tc, vr, tb and p, four identity characters is determined
    menc= Concatenate identity characters with encr
    Load txt from cloud
    utxt= Concatenate txt and menc
End
```

The pseudo code for decoding generation scheduling data from the compressed, encrypted file present in the cloud is given below.

Input: utxt; PR (or PU); dt; mnt; vr; td

Output: arr(l)

```
Begin
    First and third identity characters are computed
```

```
        utxt is loaded and these characters were searched
        Based on mnt information, the selection is more specific
        If found
                Determine tc and td information from the second character
                p= ASCII value of fourth identity character
                str= String of p characters starting from fifth character of utxt
                enc(p)= ASCII array corresponds to str
                asci(p)= Decrypt enc[p] element wise by using PR (or PU)
                If tc==1
                        arr[l]= Decompress asci(p) by using DBEA
                If tc==0
                        arr[l]= Decompress asci(p) by using E-DBEA
                end
        Otherwise
                Stop execution
        end
End
```

6.6.2. Experimental setup

For online testing of the prototype developed in the laboratory, Dropbox is employed to share encrypted file between different computers connected through internet. The encrypted file will be stored in the cloud and individual systems will update the file or decrypt, and decode a selected portion of the file. The proposed system is realized with three GS, one LDC and one DC uses Dropbox as common file for execution. The steps followed at the GS end are given below.

Step 1: At time t_1, next day availability information (td=0) of GS stored in the PC is compressed by S-DBEAE and text file with filename GS1 (or GS2) is updated in the cloud.

Step 2: At time t_3, next day scheduling information (td=1) of GS is extracted from the updated text file with filename GS1 (or GS2) by decompressing with S-DBEAE.

Step 3: If a change in ongoing day availability information is required, step 1 can be performed manually at some time t. Similarly, for decoding ongoing day scheduling information, step 2 should be performed manually at some time t.

Similar to the GS, DC must require handle next day forecasting and dispatch information. This is possible by following the steps given below.

Step 1: At time t_1, next day forecasting information (td=0) of DC stored in the PC is compressed by S-DBEAE and text file with filename DC1 is updated in the cloud.
Step 2: At time t_3, next day dispatch information (td=1) of DC is extracted from the updated text file with filename DC1 by decompressing with S-DBEAE.
Step 3: If a change in ongoing day forecasting information is required, step 1 can be performed manually at some time t. Similarly, for decoding ongoing day scheduling information, step 2 should be performed manually at some time t' (> t).

The steps to be followed at the LDC end to perform generation scheduling will be as given in the following steps.

Step 1: At time t_2 (> t_1), next day availability information of GS1 and GS2 are extracted from text files GS1 and GS2 respectively by decompressing with S-DBEAE.
Step 2: Next day load forecasting is extracted from text file DC1 by decompressing with S-DBEAE. Based on these decoded information, generation scheduling (using IW-PSO) of two GS is done and next day despatch schedule is finalized.
Step 3: Next day scheduling information (optimal loading information) of two GS and despatch schedule of DC are compressed by S-DBEAE and are updated in the respective files.
Step 4: If a change in ongoing day scheduling information is required, steps 1, 2 and 3 are performed manually at some time t.

The necessary hardware setup comprises of four PCs resembling two GS, one DC and one LDC. The experimental setup developed for realizing the proposed system is given in Figure 6.23.

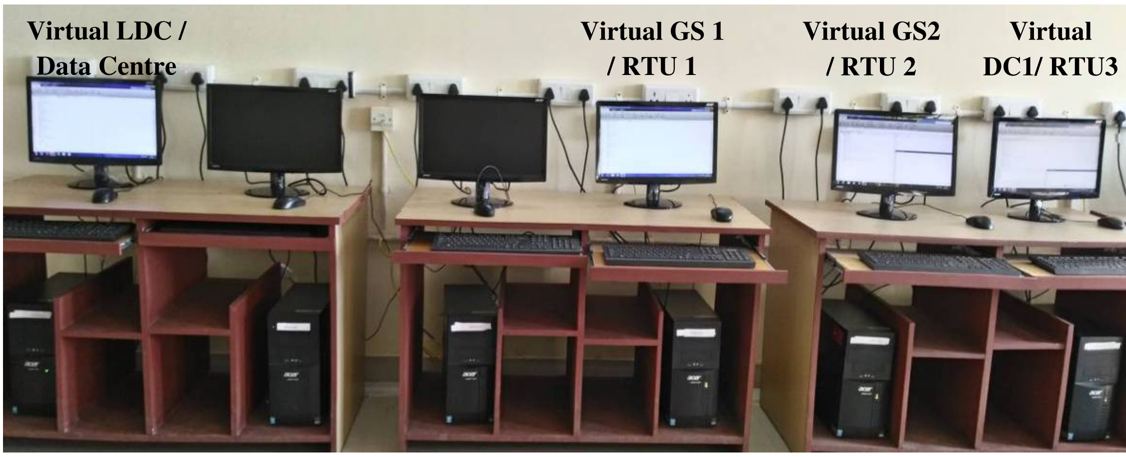


Fig. 6.23. Hardware setup for realization of proposed system using Dropbox

6.6.3. Obtained results

Due to the inclusion of few additional characters, S-DBEAE will have a lower CR than C-DBEAE. But the data management facility of the algorithm makes it superior from the previously developed works. Performance of S-DBEAE is monitored for generation scheduling information of the various units which gives an encouraging result. As C-DBEAE and S-DBEAE utilizes DBEA or E-DBEA for compressing a large data array to a character string, the comparison of the performance of the S-DBEAE is compared with DBEA and E-DBEA. Comparison of compression ratio obtained with DBEA, E-DBEA and S-DBEAE while compressing the generation scheduling information of Haldia Energy Limited (HEL), Santaldih Thermal Power Station (STPS), and Kolaghat Thermal Power Station (KTPS) operating under WBSLDC during 1st and 21st July, 2018 is given in Figure 6.24 (a), (b) and (c) respectively. The comparison of CR obtained with DBEA, E-DBEA and S-DBEAE while compressing generation scheduling information of Barh Super Thermal Generating Station (Barh) for the entire month of July 2016 is given in Figure 6.24 (d).

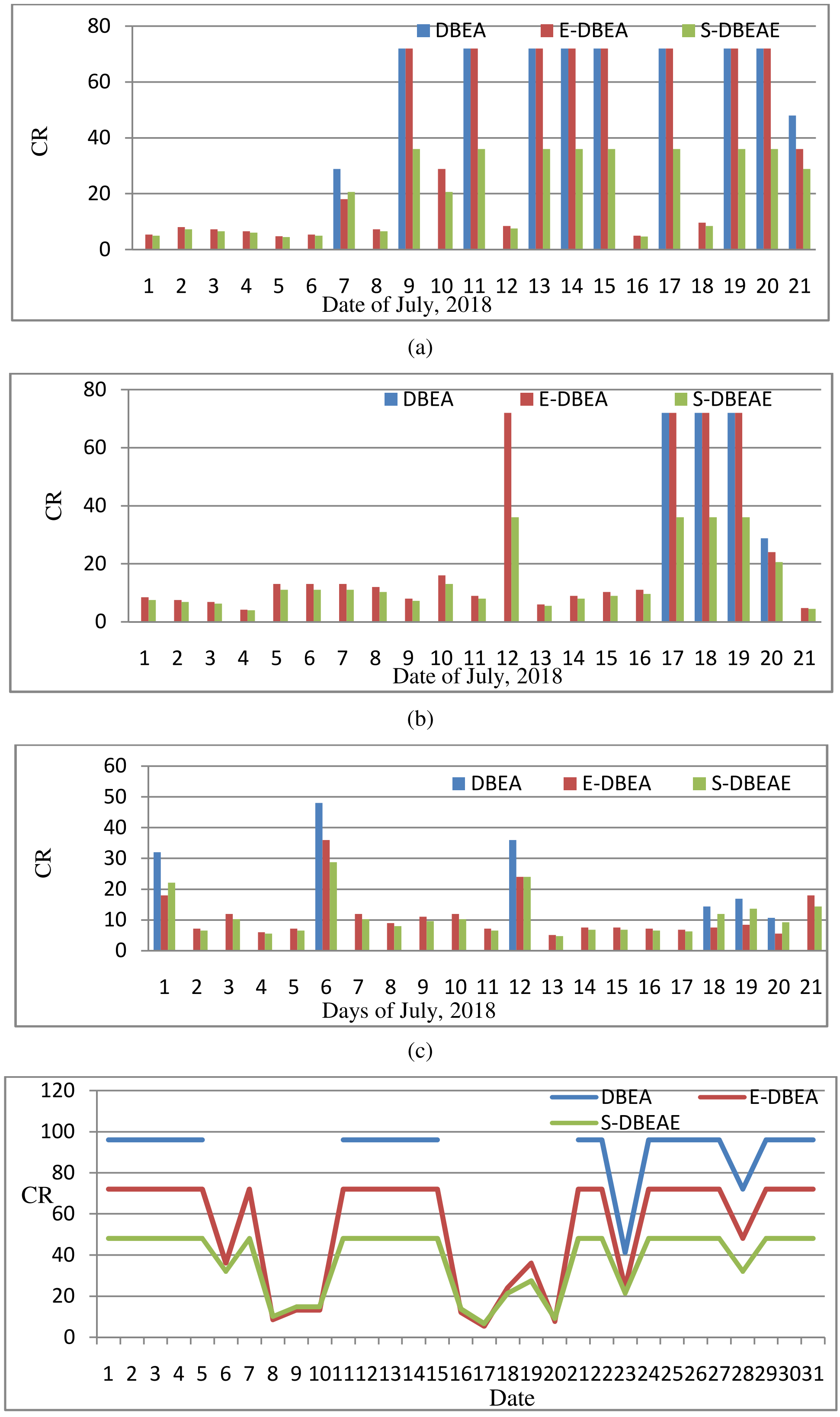


Fig. 6.24. Variation of CR obtained with DBEA, E-DBEA and S-DBEAE for generation scheduling information

As the volume of parameter monitoring information is much larger size than that of generation scheduling information, it is important to check the effectiveness of S-DBEAE with such enormous volume of data. SCADA data were collected from WBSLDC for testing purposes, which gives good results with S-DBEAE as well. Variation of CR for different system monitoring parameters at different time blocks is given in Figure 6.25. Though a varying CR is obtained with frequency and power transfer over Berhampur- Jirat 400kV line, it is much higher than 10. As the operational cost of hydel GS is zero, it is preferred to operate at its full load during the entire day until sufficient water is available. This implies that CR will remain constant throughout as given in the figure.

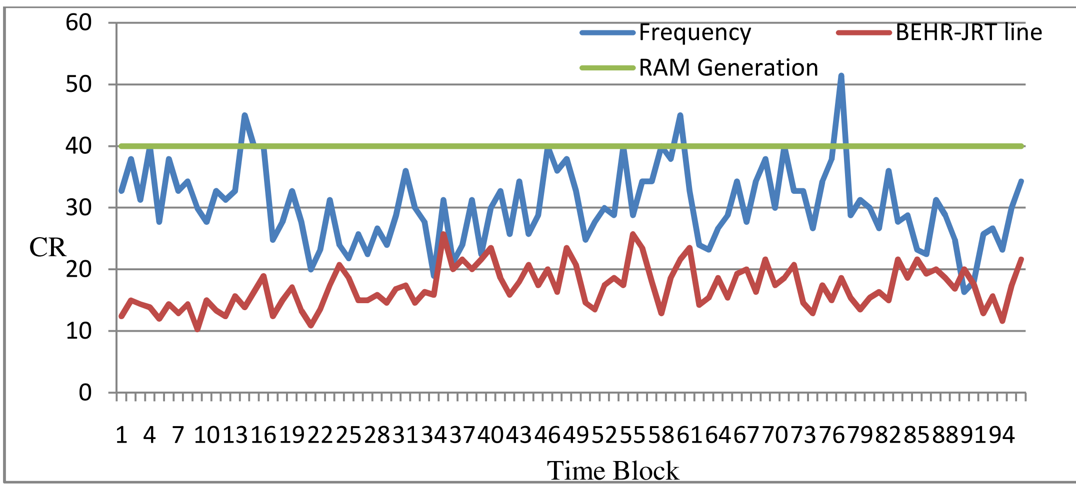


Fig. 6.25. Variation of CR obtained S-DBEAE for practical monitoring information

It is clear from the figure that S-DBEAE is effective for compressing practical power system operational data as well. In order to meet grid standards, frequency deviation is extremely rare. On the other hand, there is more variations in power transfer information as system load is varying in nature. As CR obtained by DBEA or E-DBEA increases significantly with repetition of elements in the data array, higher CR is achieved by frequency monitoring data in comparison to power transfer data. The comparison of CR obtained with DBEA, E-DBEA and S-DBEAE while compressing generation monitoring data of Santaldih Thermal Power Station (STPS) operating under WBSLDC is given in Figure 6.26 from where it is clear that S-DBEAE is a good alternative data management algorithm for power system operational data.

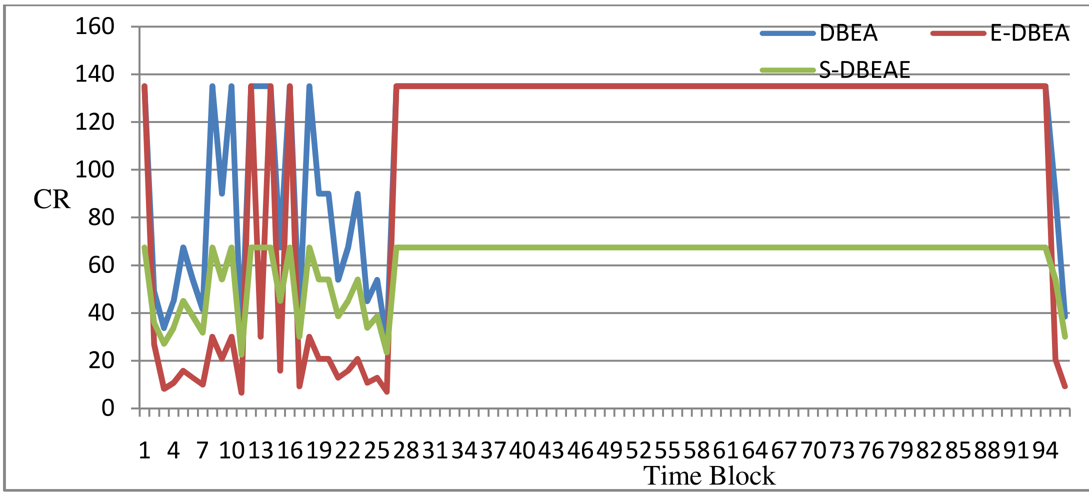


Fig. 6.26. Variation of CR obtained with DBEA, E-DBEA and S-DBEAE while compressing generation monitoring information of STPS

6.7. Performance analysis of power system operation data transfer under cloud server environment

For managing power system operational data, S-DBEAE was discussed elaborately in section 6.6. For storing different data associated with power system operation, separate text files were maintained in the cloud environment. This implies that three files corresponding to DC, GS1 and GS2 must be updated in the cloud for managing generation scheduling data. On the other hand, three files corresponding to RTU1, RTU2 and RTU3 are required for managing parameter monitoring data. The file is updated at discrete time instants as described in section 6.6.2. The block diagram of the developed system is given in Figure 6.27. At the encoding end, the encrypted character string corresponds to a data array is obtained by using S-DBEAE and is appended with the text file present in the cloud and the updated file is stored in the cloud. From this updated file, desired information is selected before and is encrypted before decoding to extract the actual data. Unit commitment information of two hydel GS (Chukha and Teesta) and two coal based GS (Barh and FSTPPIII) available in [77] over 1st-15th July, 2016 are compressed by the proposed algorithm. The results are given in Table 6.16 from where it is clear that the volume of data reduces by more than 90% for all the test results.

Table 6.16

Result obtained with S-DBEAE while compressing unit commitment information of various GS operating under E-RLDC during 1st-15th July, 2016

Sl. No.	GS Name	Uncompressed data size (Bytes)	Compressed file size (Bytes)	Compression ratio
1	Teesta	5231	252	20.76
2	CHPC	4608	280	16.46
3	FSTPP III	5760	132	43.64
4	BARH	5760	190	30.32

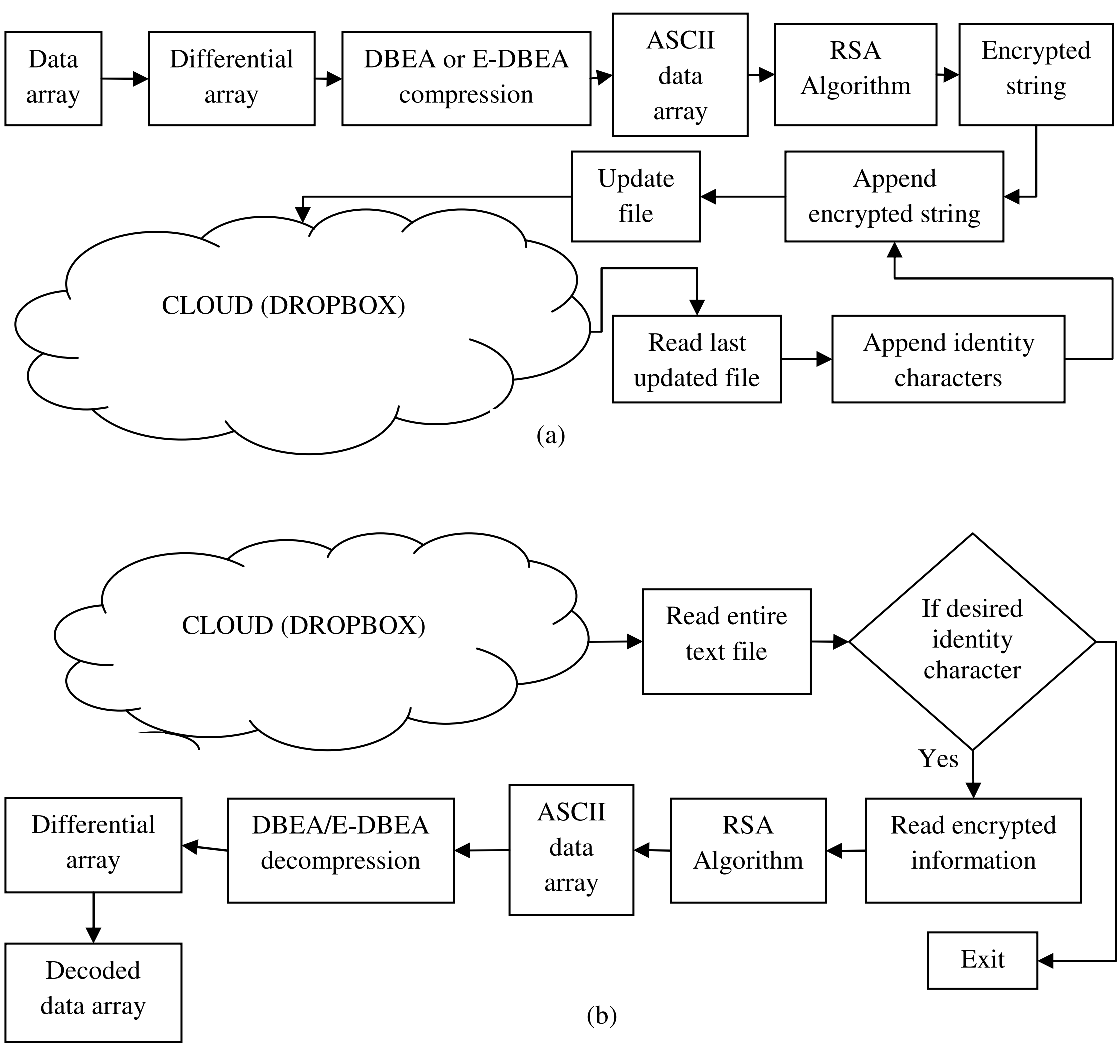


Fig. 6.27. Block diagram of S-DBEAE compression scheme (a) At encoding end and (b) At decoding end

While investigating the performance of performance of S-DBEAE while managing generation scheduling information of seven thermal generating units operating under WBSLDC during 1st July 2018 and 21st July 2018, the obtained results were encouraging. The seven units include one unit each of HEL, STPS, KTPS and two units of Bakreswar Thermal Power Plant (BkTPP) and Sagardighi Thermal Power Plant (SgTPP). The results being obtained are given in Table 6.17 which clearly describes the advantage of the proposed algorithm.

Table 6.17

Result obtained with S-DBEAE while compressing unit commitment information of various GS operating under WBSLDC during 1st-21stJuly, 2018

Sl. No.	GS Name	Uncompressed data size (Bytes)	Compressed file size (Bytes)	CR
1	HEL	6048	303	19.96
2	STPS	6048	427	14.16
3	KTPS	6048	554	10.92
4	BkTPP (1-3)	6048	472	12.81
5	BkTPP (4-5)	6048	464	13.03
6	SgTPP-I	6012	398	15.11
7	SgTPP-II	6048	438	13.81

Due to the bulk volume of parameter monitoring data, management of these information are of high importance. Comparison of uncompressed data size and encrypted file size of generation monitoring data of different hydel and thermal GS collected over an entire day is given in Table 6.18.

Table 6.18

Result obtained with S-DBEAE with parameter monitoring information during an entire day

Sl. No.	Parameter	Uncompressed data size (Bytes)	Compressed file size (Bytes)	CR
1	Frequency	69210	2370	29.16
2	Power transfer data of Behr-Jirat 400kV line	47353	2825	16.76
3	Power generation data of RAM hydel GS	34560	864	40

Chapter 7

CONCLUSION AND FUTURE SCOPE OF RESEARCH

7.1. Conclusion

The literature survey indicates that there are several works on compression of electrical signals for smart grid applications. But there are very few works on compression of power system operational data. The objective of the thesis is to develop low computational algorithms for compressing slow varying and repetitive data array associated with the power system operational data. During the initial phase of the work given in chapter 2, Huffman and Basic Arithmetic Coding based compression algorithms were developed for compressing large data array. The CR obtained with these algorithms is extremely low and thus require significant improvement. In chapter 3, some similarity between RLDA [10] and Basic Arithmetic Coding based algorithms [12] was observed. While testing the performance of RLDA for compressing practical power system operational data, the following points were observed:

i. The CR obtained with RLDA varies over a wide range
ii. While compressing a zero data array by RLDA, the output binary string will be ‘1’ and will be independent of the array length. This implies that the decoding of the actual data array is not possible.
iii. As RLDA involves Binary Arithmetic Coding, significant computation is involved in it. With the increase in the length of the data array, there is a considerable increase in computation and thus the scalability of the algorithm is dependent on the software environment.
iv. The provision of data management is not available with RLDA.

v. The algorithm can only be tested offline due to the requirement of probability distribution table for decoding the actual data array. For practical use, real time testing is important which is not possible in the present form.

The majority of these limitations is overcome by using DBEA, which is a differential coding based low computational algorithm. It is capable to compress any slow varying repetitive data array to achieve high CR. It is also scalable and can compress extremely large data array successfully. As the algorithm does not require probability distribution information at decoding end, real time testing of the algorithm is performed successfully. But obviously DBEA requires some modifications due to its following limitations:

i. DBEA fails when the difference between two elements exceeds 63.
ii. The output of DBEA will be a character string with ASCII values varying between 0-255. In many communication schemes, transfer of extended ASCII characters is not possible. When DBEA is modified to produce string comprises of ASCII characters only, the difference limit is reduced to 31.

Table 7.1

Comparison of the results obtained with DBEA and E-DBEA for compressing generation scheduling information

Date	No. of changes			String length obtained with 7-bit DBEA			String length obtained with E-DBEA		
	BTPS	KTPS	STPS	BTPS	KTPS	STPS	BTPS	KTPS	STPS
01/02/18	18	16	4	-	-	10	54	56	16
02/02/18	26	14	12	-	-	-	74	44	32
03/02/18	16	12	0	-	-	4	58	40	4
04/02/18	10	10	2	-	-	8	38	36	12
05/02/18	13	14	8	-	-	-	42	48	24
06/02/18	13	6	4	-	-	9	44	22	14
07/02/18	15	9	0	-	-	4	52	36	4

Based on these observations, E-DBEA was developed where all the elements of modified array are represented as the 14-bit binary which is then divided into 7-bit binary from which ASCII array is determined. This implies that the character string obtained with E-DBEA is comprised of ASCII characters only. This will also extend the range of zero count and difference between two consecutive elements. But during this process, the number of characters in the character string doubles and thus CR reduces by about 50%. The comparative analysis of the results obtained with DBEA and E-DBEA while compressing generation scheduling information of 3 GS operating under WBPDCL over 1st – 7th February, 2018 is given in Table 7.1. Though the performance of the E-DBEA is poorer than that for DBEA, their importance for compressing practical data can be established from the table.

Data security is a matter of concern during data transfer. In [129], a compressed, encrypted algorithm, DBEAE is proposed where DBEA compression is followed by encrypting the character string by RSA algorithm. The RSA algorithm is selected due to the fact that this algorithm can ensure practically realizable character string by proper selection of codeword at encrypting end. From Table 7.1, it is clear that DBEA can fail for many practical data sets. This observation results in the formation of C-DBEAE, where DBEA or E-DBEA is selected according to the data array. It is followed by encryption of the compressed character string is encrypted by using RSA algorithm. The information regarding the type of compression algorithm being employed is provided in an additional character before the encrypted string. The comparison of the performance of C-DBEAE with DBEA or E-DBEA while compressing generation scheduling information of SgTPP I and SgTPP II during 1st – 7th February, 2018 is given in Figure 7.1.

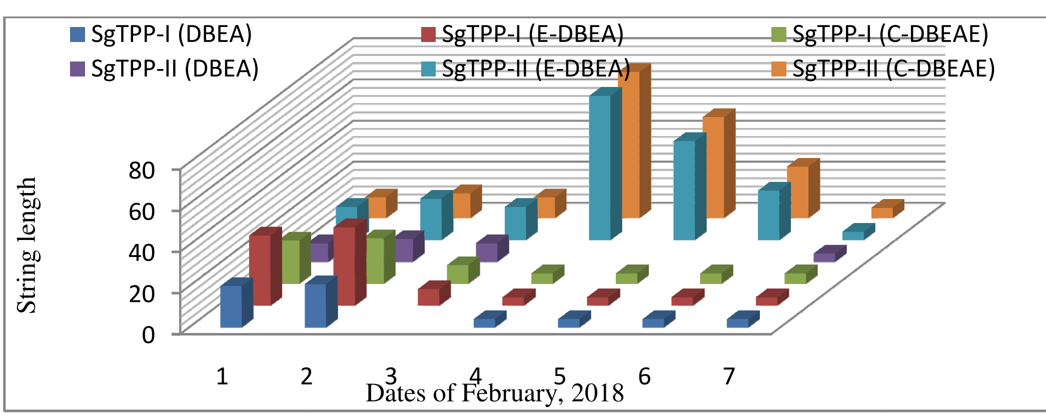


Fig. 7.1. Comparison of string length obtained with DBEA, E-DBEA and C-DBEAE for two GS

The algorithms discussed so far are useful for compressing the information at encoding end and extracting the same information at decoding end. But in many cases, multiple versions and types of data sets are available which should be distinguished correctly at decoding end. Management of such data sets are is not possible by DBEA, E-DBEA, M-DBEA or C-DBEAE. This results in the development of S-DBEAE which is similar to C-DBEAE except that it includes some additional information regarding the type of compression, version, time, type of data set etc. This enables S-DBEAE for managing large amount of information corresponds to a particular data set in a single text file. Feature-wise comparison of DBEA, E-DBEA, C-DBEAE and S-DBEAE is given in Table 7.2.

Table 7.2

Feature-wise comparison DBEA, E-DBEA, C-DBEAE and S-DBEAE

Features	**DBEA**	**E-DBEA**	**C-DBEAE**	**S-DBEAE**
CR	Highest	Generally lower than DBEA	Lower than DBEA	Lower than C-DBEAE
Data security	Not included	Not included	Included	Included
Data Management	Not possible	Not possible	Not possible	Possible
Real time testing between	Two PCs connected through serial cable	Two PCs connected through serial cable	Two PCs connected through internet using Dropbox based cloud	Four PCs connected through internet using Dropbox based cloud
Time Complexity	O(n)	O(n)	O(n)	O(n)
Advantages	Very high CR, Simplest	More data array handing capability	Highest possible CR for data, Data security is provided	Data management facility, Secured algorithm
Disadvantages	Failure with many practical data, No data security	Lower CR, No data security	Lower CR than DBEA, Increased computational burden	Lower CR than C-DBEAE, Increased computational burden

From Table 7.2, it is clear that S-DBEAE is superior to C-DBEAE due to its data management feature. In the generation scheduling data management system, next day availability information of GS and load forecasting information of DC is chosen arbitrarily. The GS constraints and B-loss matrix is considered to be that of IEEE 5-bus system and generation scheduling are done accordingly. After successful real time testing between the PC based laboratory setup of the virtual power system, it is possible to extend the principle for practical applications. The comparison of CR obtained with C-DBEAE and S-DBEAE obtained while compressing the power transfer data of Berhampur- Jirat 400kV line over an entire day is given in Figure 7.2.

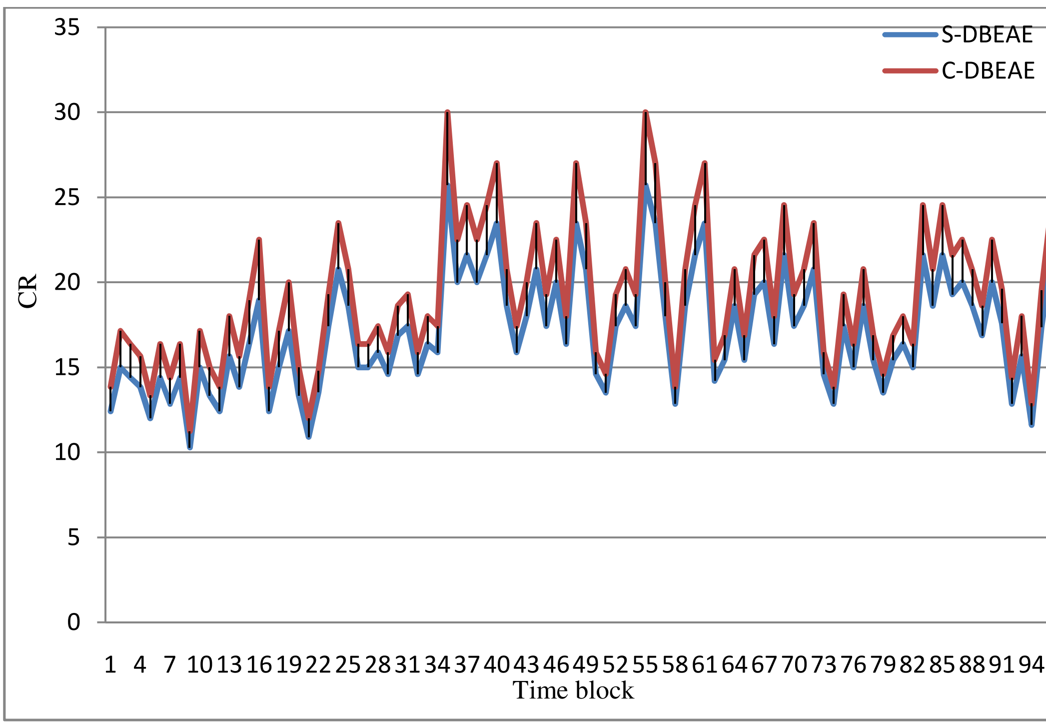


Fig. 7.2. Comparison of C-DBEAE and S-DBEAE while compressing power transfer data

From the beginning, the performance of DBEA, E-DBEA, C-DBEAE or S-DBEAE for practical power system operational data were analysed. It was clear from the discussions that these algorithms give high CR for the majority of data sets due to their repetitive nature. But obviously, there is some limiting condition beyond which compression is not possible. The lower CR is obtained when a non-repetitive data array is compressed with these algorithms. Practical analysis indicates that the elements of practical data array generally vary between 1 and 4. Based on this observation, the CR obtained with these algorithms was determined while testing them with non-repetitive data arrays with varying array length. The results being obtained for such data array are given in Figure 7.3.

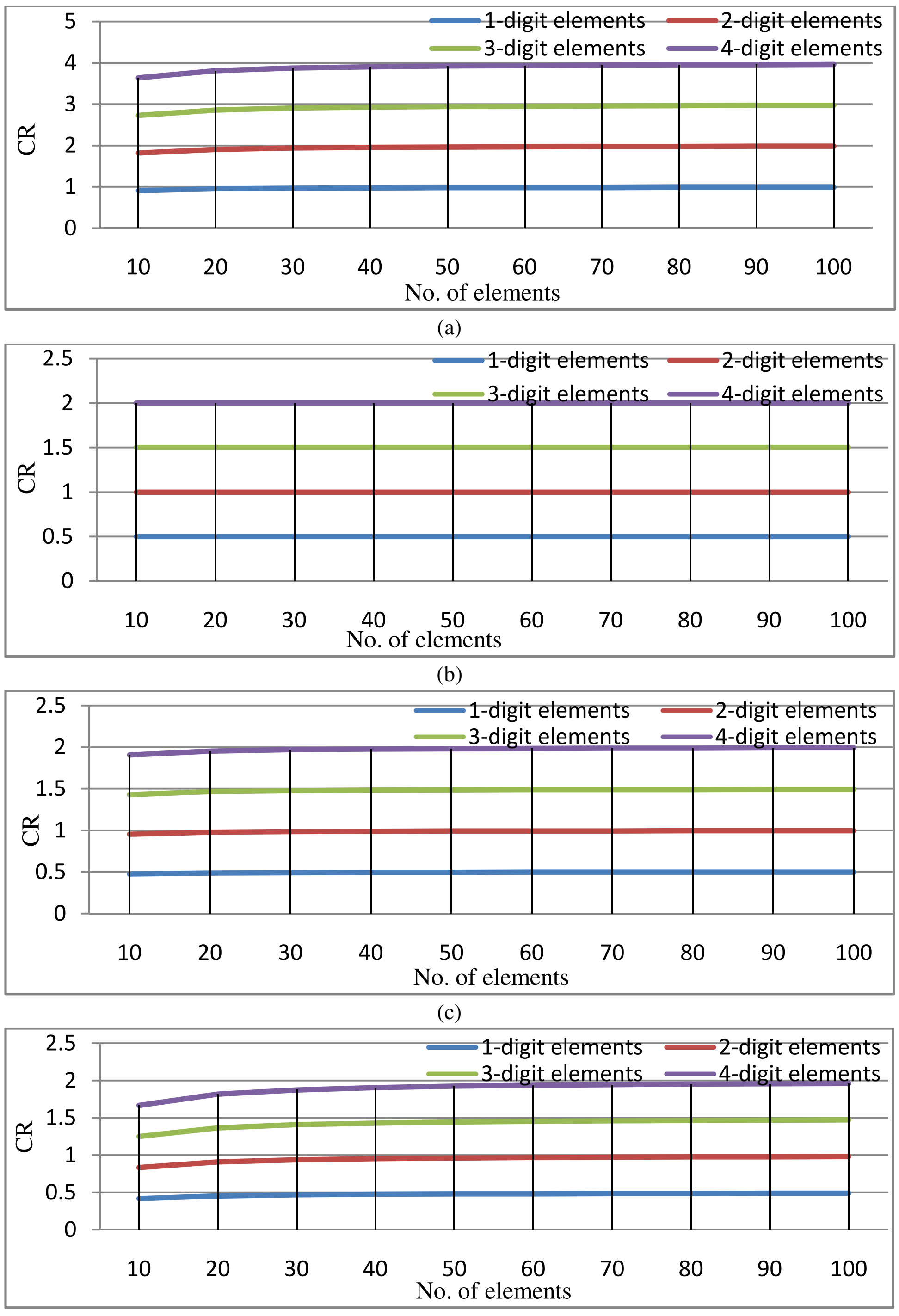
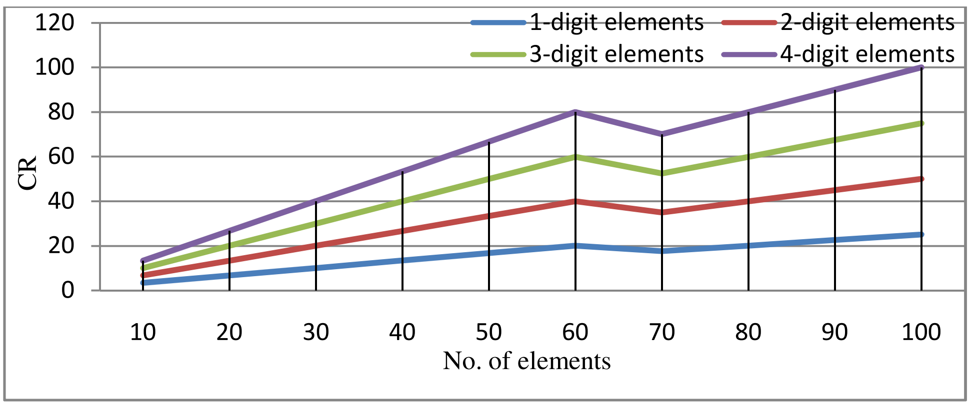
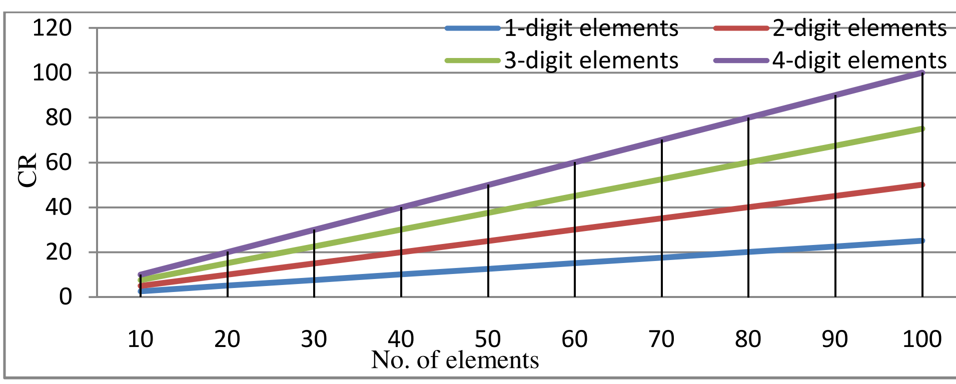


Fig. 7.3. Variation of CR with number of elements for non-repetitive data array obtained with (a) DBEA; (b) E-DBEA; (c) C-DBEAE and (d) S-DBEAE

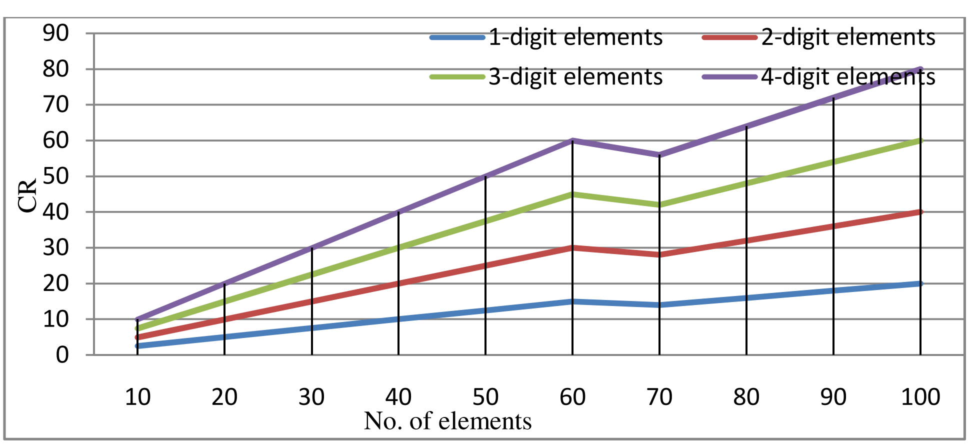
For data array containing repetitive elements, CR will be at its maximum and thereby giving the upper limit of the proposed algorithms. The variation of CR for compressing such data array will vary with the number of digits of the elements. The results being obtained with DBEA, E-DBEA, C-DBEAE and S-DBEAE with repetitive data array of varying number of elements is given in Figure 7.4.



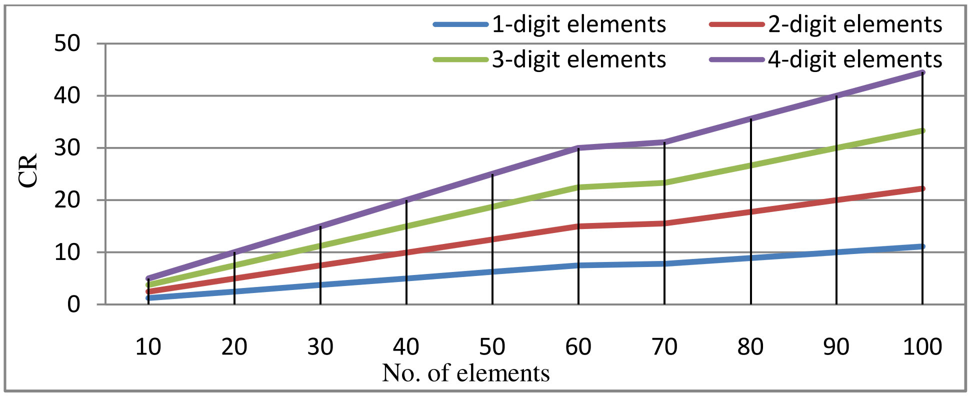
(a)



(b)



(c)



(d)

Fig. 7.4. Variation of CR with the number of identical elements in the data array obtained with (a) DBEA; (b) E-DBEA; (c) C-DBEAE and (d) S-DBEAE

It is clear from the comparative analysis of Lambda iteration, IW-PSO and ANN that the neural net containing 10 hidden layers give the best results where the absolute percentage error for two GS is less than 0.106% and 0.15% respectively. On the other hand, the absolute percentage error for two GS varies between 0.024% - 2.897% and 0.024% - 3.5% when the number of hidden layers is 3. From the average absolute % errors it is clear that the accuracy of 10-layer net is way ahead of 8-layer net and this can be considered as the best choice for solving ELD problem. But obvious ANN will be useless without proper training and thus require an enormous volume of data for training purpose. IW-PSO, on the other hand do not require any training data and can solve ELD problem directly. With increased particle size, the accuracy of IW-PSO increases, but it will result in increased computational time. For a 50-particle IW-PSO, the maximum absolute error being encountered is 2.787% and the average execution time is 0.314s. On the other hand, the maximum absolute error of 3.317% is obtained by 30-particle IW-PSO with average execution time of 0.24s. The difference of average absolute error obtained with IW-PSO having 50 and 30 particles is merely 0.1%. Due to lower execution time and comparable absolute error, 30-particle IW-PSO is apparently more suitable for solving ELD problem.

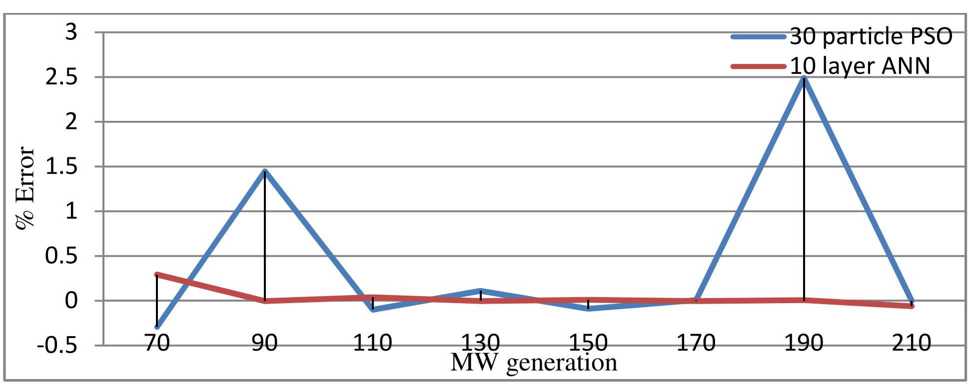
The comparison of the performance of the Lambda iteration method, 30-particle IW-PSO and 10-layer net is given in Table 7.3 to conclude the discussions. The variations of percentage error with load demand for 30-particle PSO and 10-layer neural net is given in Figure 7.5. It is clear the figure and table that 30-particle PSO gives pretty good results in the linear section of Figure 4 and the performance degrades at both ends.

10-layer net, in contrast, deviates a little from the training data pattern only in the beginning. This implies that 10-layer ANN is obviously a better choice than IW-PSO for solving ELD problem although its execution time is higher. For IEEE 5-bus system considered here, it is clear that Lambda iteration method is the fastest method of solving ELD problem due to much reduced computation involved. But with increased number of buses i.e. increased number of GS, solution of ELD problems by the Lambda iteration method becomes no longer feasible. In such cases, neural net will be a better alternative provided that the training data is available.

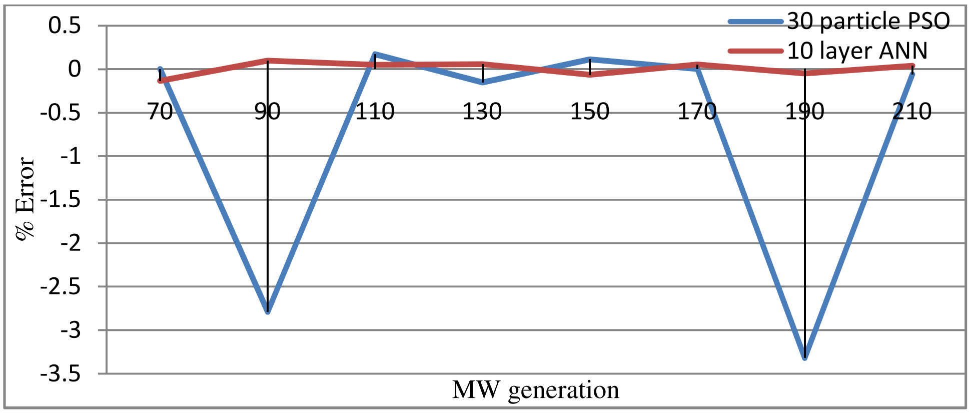
Table 7.3

Comparison of Lambda iteration method, 30-particle PSO and 10-layer neural net to solve ELD problem for IEEE 5-bus system

Load demand (in MW)	With lambda iteration method			With 30-particle PSO			With 10-layer ANN		
	P_1 (in MW)	P_2 (in MW)	Exec. time (ms)	P_1 (in MW)	P_2 (in MW)	Exec. time (ms)	P_1 (in MW)	P_2 (in MW)	Exec. time (ms)
70	41.08	30	64	40.96	30	207	41.2	29.96	886ms for entire data set
90	60.84	30.86	37	61.72	30	212	60.84	30.89	
110	71.21	41.28	45	71.14	41.35	122	71.24	41.3	
130	81.67	51.78	66	81.76	51.7	647	81.67	51.81	
150	92.22	62.37	48	92.14	62.44	44	92.23	62.33	
170	102.86	73.04	47	102.87	73.04	50	102.86	73.08	
190	113.61	83.8	78	116.44	81.02	315	113.62	83.76	
210	120	99.09	79	120	99.03	323	119.93	99.13	



(a)



(b)

Fig. 7.5. Variation of % error with MW generation obtained with 30-particle PSO and 10 layer ANN (a) For GS 1 and (b) For GS 2

The proposed compression system is realized both for data transfer and data storage. It is obvious that if one or multiple data is corrupted, actual information can't be extracted at decoding end. As there is a handshaking based data transfer, loss of information is difficult. If somehow there might have some loss of bits, this problem can be identified by parity bits. In order to achieve reliable data transfer through a lossy channel, convolution codes are used extensively. The real time testing done so far, do not suffer from this problem and thus data corruption wasn't considered. In modern day, internet based data transfer is getting extreme importance. The output of the proposed algorithms is a text file (in .txt format) which can be transferred (or shared) easily through internet without any information loss. While concluding the section, the major contributions of the research work can be summarized as:

i. In this thesis, works were carried out for the development of compression algorithms for compressing power system operational data. This will be helpful for reducing the memory requirement for storing such enormous volume of data for future references. Data compression is also helpful for transferring the information through suitable communication with less energy consumption. During the initial phase of the work, Huffman Coding based algorithm was designed for compressing a data array where the CR is less than 2. In the next stage, Basic Arithmetic Coding is used for compressing the data array. Though this algorithm gives higher CR than the previous one, but not high enough to satisfy the requirements. While checking the performance of RLDA for compressing power system operational data, the idea of DBEA had evolved. In order to overcome the limitation of DBEA, E-DBEA was threaded. Both DBEA and E-DBEA is capable to give high CR for the majority of the available CR without introducing much complicated computation.

ii. Data security is a matter of concern particularly during data transmission. To ensure that, the asymmetric cryptography algorithm is included where separate keys are required at encrypting and decrypting end. As RSA algorithm can encrypt an integer to another integer between 0 to (n-1), proper selection of n will result in the formation of integers having some character equivalent.

iii. DBEA fails when the difference between two consecutive elements exceeds a certain value. The range of differential value can be extended by using E-DBEA but the CR reduces drastically. In order to extract the highest possible CR, C-DBEAE is developed where DBEA or E-DBEA is selected according to the data array and an identity character is used to distinguish between them.

iv. Lambda iteration, IW-PSO and ANN were developed for solving ELD problem of the IEEE 5-bus system and tested to obtain the necessary results. Due to the much reduced computational time of Lambda iteration method, it was the best choice for IEEE 5-bus system. It is obvious that the performance of the Lambda iteration method will degrade with the increase in number of GS in the system. For such large system, IW-PSO or ANN will be a better choice over Lambda iteration method. Though high accuracy can be achieved by ANN in comparison to IW-PSO, requirement of training data sets is the major drawback of ANN.

v. For managing multiple data, none of the algorithms are suitable. S-DBEAE is developed for managing such information where few identity characters are added before the compressed, encrypted string according to different fields. A power system generation scheduling data management system is realized in the laboratory where various data associated with generation scheduling are compressed by S-DBEAE and updates in a single text file.

The following advantages were obtained while realising cloud based power system operational data management in the laboratory.

i. Reduced memory requirement and thereby lower data storage cost
ii. Provision of data security and system scalability
iii. Easier extraction of any particular dataset from a large chunk of information present in the cloud
iv. Autonomous system which involves very little human effort to serve the desired purpose
v. Due to the simplicity of the algorithms, any additional system is not required. So, the proposed system can be used with the existing infrastructure with only some slight modifications.

7.2. Future Scope of Research

- As DBEA and E-DBEA gives lower CR with less-repetitive data array, there is a scope of suitable modification so as to improve CR for any less-repetitive data array.
- Load requirement of distribution companies can be computed directly from the current reading of distribution feeders. If a cloud based system can be realized, it is possible to store the sensor reading directly to the cloud, according to which distribution data centre can take necessary action for load shedding or demanding additional power from the LDC monitoring system condition directly from cloud data. The block diagram of this system can be as given in Figure 7.6.

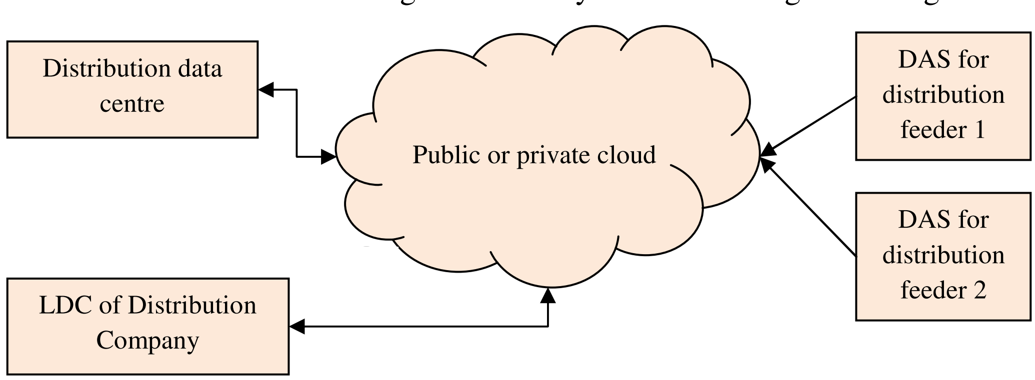


Fig. 7.6. Cloud based distribution system monitoring

- In the developed system, data transfer for generation scheduling between LDC, GS and DC employs internet connectivity. As the volume of data handled during generation scheduling data transfer is lower in comparison to that handled during parameter monitoring data transfer, it can be transferred by using GSM technology instead. But, the GSM data transfer supports transfer of printable characters only. This implies that compression algorithms like M-DBEA [109] can be useful for realizing such system. The block diagram of this system can be as given in Figure 7.7. As the time required for data transfer will be high when the GSM network is congested, some alternative arrangement must be provided. Beside this constraint, GSM based data transfer also results in increased cost of data transfer due to considerable SMS charges.

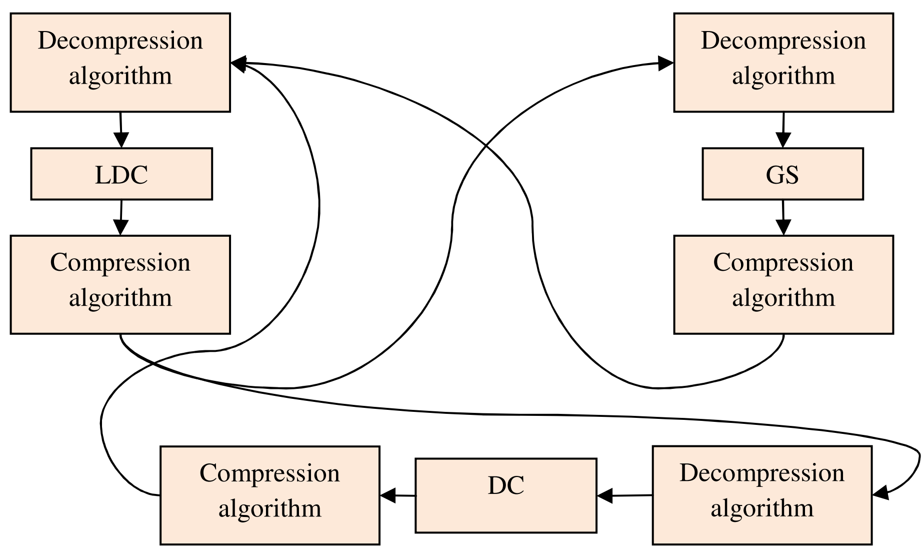


Fig. 7.7. GSM based scheduling data transfer

- Installation of PMU is extremely common now-a-days due to its advantages. But due to increased sampling frequency, the volume of data increases considerably. The performance of the proposed algorithms can be investigated with those data and if the performance is found to be satisfactory, they can be good choice to reduce the volume of information as well. For any domestic household load, energy consumption often does not vary over a wide range. This implies that the difference array obtained from the smart meter reading will contain considerable repetition of data. If the proposed algorithms are tested with those data and a high CR is obtained, they again can be a good choice for load profile data compression.

- It is possible to include various predictive algorithms with the proposed parameter monitoring data management system which can predict any system instability in the upcoming future. This feature can be useful to any DAS monitoring some critical parameters.
- In [132], a microcontroller based system is realized where a domestic consumer consumes more energy than a threshold value during peak hours, the consumer is warned by SMS. If the energy consumption is not reduced, a relay will operate to disconnect the supply. A GSM based system to actuate relays for disconnecting various service mains of a power transformer is realized in [133]. Similar works are available in some other literatures where GSM communication is employed to perform load shedding in specific areas. With the advent of smart metering, it is possible to monitor electricity consumed by any household remotely by GSM, PLCC or any other communication technique. A system can be realized to monitor power consumed by different household and store the compressed information periodically in a database by using the suitable communication channel. The system will be smart enough to take care of demand side management, if required by sending SMS to the consumer before taking any action of its own. The information of all household readings will be decompressed at the data centre for regular monitoring. The block diagram of such system can be as given in Figure 7.8.

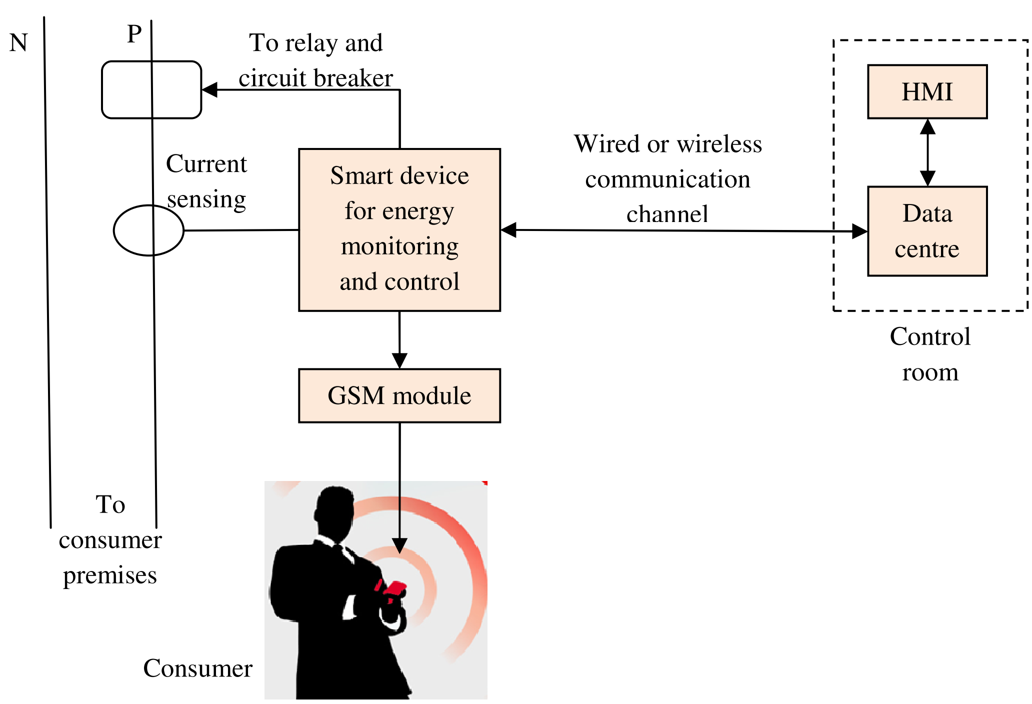


Fig. 7.8. A demand side management system for smart metering

REFERENCES

[1] Manar Jaradat, Moath Jarrah, Abdelkader Bousselham, Yaser Jararweh, Mahmoud Al-Ayyoub, "The Internet of Energy: Smart Sensor Networks and Big Data Management for Smart Grid", Proc. of The International Workshop on Networking Algorithms and Technologies for IoT (NAT-IoT 2015), 17th – 20th August, 2015, Belfort, France, pp. 592- 597.

[2] Xi Fang, Satyajayant Misra, Guoliang Xue, Dejun Yang, "Smart grid; the new and improved power grid: A survey", IEEE Communications Surveys & Tutorials, Vol. 14, Iss. 4, Fourth quarter, 2012, pp. 944-980.

[3] Atimad El Khaouat, Laila Benhlima, "Big Data Management for Smart Grids", Proc. of International Renewable and Sustainable Energy Conference (IRSEC), 2015, 10th – 13th December, 2015, Marrakech, Morocco, pp. 1044- 1047.

[4] Umar Ahsan, Abdul Bais, "Distributed Big Data Management in Smart Grid", Proc. of 26th Wireless and Optical Communication Conference (WOCC), 2017, 7th - 8th April, 2017, Newark, USA, pp. 1-6.

[5] Joonsang Baek, Quang Hieu Vu, Joseph K. Liu, Xinyi Huang, Yang Xiang, "A Secured Cloud Computing based Framework for Big Data Information Management of Smart Grid", IEEE Transactions on Cloud Computing, Vol. 3, Iss. 2, April-June 1 2015, pp. 233- 244.

[6] Xu Daqing, Han Yinghua, "An Adaptive Data Management Model for Smart Grid", Proc. of 8th International Conference on Intelligent Computation Technology and Automation (ICICTA), 2015, 14th - 15th June, 2015, Nanchang, China, pp. 126- 129.

[7] Yingyao Zhou, Ping Li, Yuning Xiao, Anum Masood, Qichen Yu, Bin Sheng, "Smart Grid Data Mining and Visualization", Prof. of International Conference on Progress in Informatics and Computing (PIC), 2016, 23rd – 25th December, 2016, Shanghai, China, pp. 536- 540.

[8] A. R. Al-Ali, Raafat Aburukba, "Role of Internet of Things in the Smart Grid Technology", Journal of Computer and Communications, Vol. 3, Iss. 3, June- July 2015, (Online) Available: http://file.scirp.org/Html/57559_57559.htm.

[9] White paper: A Smarter Grid with the Internet of Things, Olivier Monnier, pp. 1-10, Available (Online): http://www.ti.com/lit/ml/slyb214/slyb214.pdf.

[10] Tamzida Akter, “Smart Grid with Internet of Things”, (Online) Available: https://www.sanog.org/resources/sanog28/SANOG28-Conference_Smart-Grid-with-Internet-of-Things.pdf.

[11] Lee-Cheun Hau, Jer-Vui Lee, Yea-Dat Chuah and An-Chow Lai, “Smart Grid- The Present and Future of Smart Physical Protection: A Review”, International Journal of Energy, Information and Communications, Vol. 4, Iss. 4, August, 2013, pp. 43- 53.

[12] P Naveen , Wong Kiing Ing , Michael Kobina Danquah , Amandeep S Sidhu, Ahmed Abu-Siada, “Cloud computing for energy management in smart grid – an application survey”, Proc. of Curtin University of Technology Science and Engineering International Conference (CUTSE), 2015, 6th -8th November, 2015, Miri, Malaysia, pp. 1-11. (Online) Available: http://iopscience.iop.org/article/10.1088/1757-899X/121/1/012010/pdf.

[13] Nilotpal Chakraborty, Arijit Mondal, Samrat Mondal, “Intelligent Scheduling of Thermostatic Devices for Efficient Energy Management in Smart Grid”, IEEE Transactions on Industrial Informatics, Vol. 13, Iss. 6, December 2017, pp. 2899-2910.

[14] Wenchao Meng, Xiaoyu Wang, “Distributed Energy Management in Smart Grid with Wind Power and Temporally Coupled Constraints”, IEEE Transactions on Industrial Electronics, Vol. 64, Iss. 8, August, 2017, pp. 6052- 6062.

[15] Chengcheng Zhao, Jianping He, Peng Cheng, Jiming Chen, “Consensus Based Energy Management in Smart Grid with Transmission Losses and Directed Communication”, IEEE Transactions on Smart Grid, Vol. 8, Iss. 5, September 2017, pp. 2049- 2061.

[16] Umang M Patel, Mitul M Modi, “A Review on Smart Meter System”, International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering, Vol. 3, Iss. 12, December, 2015, pp. 70- 73. (Online) Available: http://www.ijireeice.com/upload/2015/december-15/IJIREEICE%2015.pdf.

[17] Damminda Alahakoon, Xinghuo Yu, “Smart Electricity Meter Data Intelligence for Future Energy Systems: A Survey”, IEEE Transactions on Industrial Informatics, Vol. 12, Iss. 1, Feb. 2016, pp. 425- 436.

[18] Janardhana S., M. S. Shashikala, “Smart Meter Parameters for Smart Grid”, International Journal of Research in Engineering and Technology, Vol. 5, Iss. 3, March, 2016, pp. 134- 137. (Online) Available: http://esatjournals.net/ijret/2016v05/i03/IJRET20160503028.pdf.

[19] Reynaldo Francisco Nuqui, "State Estimation and Voltage Security Monitoring Using Synchronized Phasor Measurements", PhD Thesis, Virginia Polytechnic Institute and State University, July, 2001. (Online) Available: http://citeseerx.ist.psu. edu/viewdoc/download? doi=10.1.1.2.7959&rep=rep1&type=pdf.

[20] Ruchi Verma, Balram Timande, "A Survey on Phasor Measurement Unit for Power System", International Journal of Digital Application & Contemporary Research, Vol. 2, Iss. 1, August, 2013. (Online) Available: https://pdfs.semanticscholar.org/2dc7/ 9df370dff29f9e 0204d961b4ac0825622423.pdf.

[21] Rohini P. Haridas, "GPS based Phasor Technology in Electrical Power System", International Journal of Electronics and Electrical Engineering Vol. 3, No. 6, December 2015, pp. 493- 496. (Online) Available: http://www.ijeee.net/uploadfile/ 2015/0710/20150710054436686.pdf.

[22] R. E. Wilson, "PMUs [Phasor Measurement Units]", IEEE Potentials, Vol. 13, Iss. 2, April 1994, pp. 26 -28.

[23] Muhammad Aamir , Javier Poncela , Muhammad Aslam Uqaili , B.S.Chowdhry , Nishat Ahmad Khan, "Optimal Design of Remote Terminal Unit (RTU) for Wireless SCADA System for Energy Management", Wireless Personal Communications, Vol. 69, Iss. 3, pp. 999-1012. (Online) Available: https://pdfs.semantic scholar.org/db76/ dd8bbdc7891e4c86d4f249e3d872a443e8df.pdf.

[24] White Paper: SCADA Systems. (Online) Available: https://www.motorola solutions.com/content/dam/msi/Products/scadasystems/SCADA_Sys_Wht_Ppr2a_New. pdf.

[25] Francis Enejo Idachaba, Ayobami Ogunrinde, "Review of Remote Terminal Unit (RTU) and Gateways for Digital Oilfield deployments", International Journal of Advanced Computer Science and Applications, Vol. 3, No. 8, August 2012, pp. 157-160. (Online) Available: http://citeseerx.ist.psu.edu/viewdoc/download? doi=10.1.1.259. 1524&rep=rep1&type=pdf.

[26] Shabnam Rukhsar, "SCADA in Transmission Line", Proc. of International Conference on Advances in Engineering & Technology – 2014 (ICAET-2014), pp. 67-71. (Online) Available: http://www.iosrjournals.org/iosr-jeee/Papers/ICAEndT2014/ electrical/volume-1/13.pdf?id=7590.

[27] Yang Wang, Wilsun Xu, James Shen, “Online Tracking of Transmission Line Parameters using SCADA Data”, IEEE Transactions on Power Delivery, Vol. 31, Iss. 2, April 2016, pp. 674- 682.

[28] Usama Fayyad, Gregory Piatetsky-Shapiro, Padhraic Smyth, “From Data Mining to Knowledge Discovery in Databases”, AI Magazine, 1996, pp. 37- 54. (Online) Available: http://www.kdnuggets.com/gpspubs/aimag-kdd-overview-1996-Fayyad.pdf.

[29] Katherine Margaret Rogers, “Data-Enhanced Applications for Power Systems Analysis”, PhD Thesis, Graduate College of the University of Illinois, 2011. (Online) Available: https://tcipg.org/sites/default/files/papers/2011_Q4_RPA3.pdf.

[30] I. S. Saeh , M. W. Mustafa, “Data Mining Classifiers for Static Security Evaluation in Power System”, International Journal of Computer Science Issues, Vol. 9, Iss. 2, No 1, March 2012, pp. 146- 151. (Online) Available: https://www.ijcsi.org/ papers/IJCSI-9-2-1-146-151.pdf.

[31] Yagang Zhang, Jing Ma, Jinfang Zhang, Zengping Wang, “Applications of Data Mining Theory in Electrical Engineering”, Vol. 1, No. 3, November 2009, pp. 211- 215. (Online) Available: https://file.scirp.org/pdf/ Engineering20090300010_13 502123.pdf.

[32] Bogdan Constantin Neagu, Gheorghe Grigoras, “Data Mining Tools in Electricity Distribution Systems”, Proc. of 6th International Conference on Modern Power Systems (MPS- 2015), 18th – 21st May, 2015, Cluj- Napoca, Romania, pp. 209- 212.

[33] Nurulhuda Binti Hj Hanzah, “Intelligent Power System Protection Data Management”, M.E.E. Thesis, Universiti Tun Hussien Onn Malaysia, December, 2013. (Online) Available: http://eprints.uthm.edu.my/5536/1/NURULHUDA_BINTI_HJ_ HANZAH.pdf.

[34] Yongli Wang, Dongxiao Niu, Yakun Wang, “Power Load Forecasting using Data Mining and Knowledge Discovery Technology”, Proc. of 2nd Asian Conference on Intelligent Information and Database Systems (ACIIDS 2010), 24th – 26th March, 2010, Hue City, Vietnam, pp. 319- 328.

[35] John J. Grainger, William D. Stevenson Jr., “Power System Analysis”, International Edition, McGraw Hill, 1994.

[36] D. P. Kothari, I. J. Nagrath, “Modern Power System Analysis”, McGraw Hill, 2008. (Online) Available: http://www.gcebargur.ac.in/sites/gcebargur.ac.in/files/ lectures_desk/MPSA-Sol.pdf.

[37] Xi-Fan Wang, Yonghua Song, Malcolm Irving, "Modern Power Systems Analysis", Springer, 2008.

[38] B.M. Weedy, B.J. Cory, N. Jenkins, J.B. Ekanayake, G. Strbac, "Electric Power System", 5th Edition, John Wiley & Sons Ltd., 2012.

[39] Draft of Indian Electricity Grid Code by Central Electricity Regulatory Commission. (Online) Available: http://www.cercind.gov.in/2010/ORDER/February 2010/IEGC_Review _Proposal.pdf.

[40] Subhra J. Sarkar, Palash K. Kundu, "A Proposed Method of Load Scheduling and Generation Control using GSM and PLCC Technology", Proc. of Michael Faraday IET International Summit (MFIIS 2015), 12th – 13th September, 2015, Kolkata, India, pp. 273- 277.

[41] P. S. Kulkarni, R. M. Holmukhe, "Infrastructural Analysis of Load Despatch Centre", International Journal of Computer Applications, Vol. 1, No. 7, 2010, pp. 100- 105. (Online) Available: https://www.ijcaonline.org/journal/number7/pxc387276.pdf.

[42] Vivek Pandey, "Electricity Grid Management in India: An Overview", 2007 Annual issue of Electrical India, Vol. 47, No. 11, November 2007. (Online) Available: http://nrldc.in:83/docs/documents/Articles/GridMgmtOverview_Nov2007_ElectricalIndia.pdf.

[43] Discussion Paper: "Revised scheme of incentive and disincentive for Thermal Generating Stations". (Online) Available: http://www.cercind.gov.in/08022007/ Discussion% 20Paper_1.pdf.

[44] Subhra J. Sarkar, Nabendu Kr. Sarkar, Aritra Banerjee, "A Novel Huffman Coding based Approach to Reduce the Size of Large Data Array", Proc. of IEEE International Conference on Circuit, Power and Computing Technologies (ICCPCT- 2016), 18th – 19th March 2016, Nagercoil, Tamilnadu, India, pp. 1-5.

[45] Rose Tsang, "Cyberthreats, Vulnerabilities and Attacks on SCADA Networks". (Online) Available: https://web.archive.org/web/20120813015252/http://gspp.berkeley.edu/iths/Tsang_SCADA%20Attacks.pdf.

[46] Nicoleta Arghira , Daniela Hossu , Ioana Făgărăşan , Sergiu Stelian Iliescu , Daniel Răzvan Costianu, "Modern SCADA Philosophy in Power System Operation- A Survey", U.P.B. Science Bulletin, Series C, Vol. 73, Iss. 2, 2011, pp. 153- 166. (Online) Available: https://www.scientificbulletin.upb.ro/rev_docs_arhiva/full87719.pdf.

[47] Subhra J. Sarkar , Nabendu Kr. Sarkar , Trishayan Dutta , Panchalika Dey, Aindrila Mukherjee, "Arithmetic Coding Based Approach for Power System Parameter Data Compression", Indonesian Journal of Electrical Engineering and Computer Science, Vol. 2, No. 2, May 2016, pp. 268- 274.

[48] Jeff Dagle, "Supervisory Control and Data Acquisition (SCADA) Introduction". (Online) Available: http://www.science.smith.edu/~jcardell/Readings/TRUST%20US/2005_09_15_Jeff_Dagle.pdf.

[49] Kalapatapu Rao, "SCADA Protocols and Communication Trends", ISA2004 Paper. (Online) Available: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.401.6666&rep=rep1&type=pdf.

[50] West Bengal State Load Despatch (WBSLDC) Website. (Online) Available: http://wbsldc.in/schd.aspx?typ=5.

[51] Eastern Regional State Load Despatch (ERLDC) Website. (Online) Available: http://www.erldc.org/final-schedule.aspx.

[52] Tae-Il Choi, Kwang Y. Lee, Dong Ryul Lee, Jeong Kil Ahn, "Communication System for Distribution Automation Using CDMA", IEEE Transactions on Power Delivery, Vol. 23, No. 2, April 2008, pp. 650- 656.

[53] Muhammad Irfan Ullah Khan, Muhammad Riaz, "Various Types of Smart Grid Techniques: A Review", International Journal of Multidisciplinary Sciences and Engineering, Vol. 7, No. 8, December 2016, pp. 8- 14. (Online) Available: http://www.ijmse.org/ Volume7/Issue8/paper2.pdf.

[54] Mu Wei, Zhe Chen, "Communication Systems and Study Method for Active Distribution Power Systems", Proc. of 9th Nordic Electricity Distribution and Asset Management Conference, 6th – 9th September, 2010, Aalborg, Denmark, pp. 1- 11. (Online) Available: http://vbn.aau.dk/files/45991065/Communication_systems_and_study_method_for_active_dictribution.pdf.

[55] Edmund O. Schweitzer, III, David Whitehead, Ken Fodero, and Paul Robertson, "Merging SONET and Ethernet Communications for Power System Applications", pp. 1-7. (Online) Available: https://cdn.selinc.com/assets/Literature/Publications/Technical%20Papers/6513_MergingSONET_KF_20120424_Web.pdf?v=20151125-084434.

[56] Richard Joseph, Nerey Mvungi, "Effective Communication System(s) in Automation of Power Systems", International Journal of Research in Engineering and Technology (IJRET) Vol. 2, No. 5, 2013, pp. 228- 231. (Online) Available: http:// journalsweb.org/site admin/upload/73976%20IJRET025047.pdf.

[57] Abdul Mannan, D.K.Saxena, Mahroosh Banday, "A Study on Power Line Communication", International Journal of Scientific and Research Publications, Vol. 4, Iss. 7, July 2014, pp. 1-4. (Online) Available: http://www.ijsrp.org/research-paper0714/ ijsrp-p3152.pdf.

[58] S. M. Raza, F. Azam, "An intelligent monitoring and control system using power line data communication", Proc. of IEEE Students Conference (ISCON'02), 16th – 17th August, 2002, Lahore, Pakistan, pp. 1-5.

[59] Stephen F. Adam, Hewlett Packard, "Microwave Theory and Applications". (Online) Available: http://uspas.fnal.gov/materials/MicrowaveTheory.pdf.

[60] "Fundamentals of Optical Fiber Transmission". (Online) Available: http://www .mitr.p.lodz.pl/evu/lectures/Abramczyk1.pdf.

[61] Barnaby D Pitt, "Applications of Data Mining Techniques to Electric Load Profiling", Ph.D. Thesis, Dept. of Electrical and Electronic Engineering, University of Manchester Institute of Science and Technology, 2000.

[62] Muhammad Shaheen, Muhammad Shahbaz, Khalid Afsar Khan Jadoon, "Data Mining for Wind Energy Site Selection", Proc. of the World Congress on Engineering and Computer Science 2012 (WCECS 2012), 24th – 26th October 2012, San Francisco, USA, Vol. I. (Online) Available: http://www.iaeng.org/publication/WCECS2012/ WCECS2012_pp471-476.pdf.

[63] Sarasij Das, P S Nagendra Rao, "Understanding Power System Behaviour Through Mining Archived Operational Data", Proc. of the Fifteenth National Power Systems Conference (NPSC), 16th – 18th December 2008, IIT Bombay, India. (Online) Available: http://eprints.iisc.ac.in/17606/2/npsc_08_sarasij_nagendra_iisc.pdf.

[64] Michel P. Tcheou, Lisandro Lovisolo, Moises V. Ribeiro, Eduardo A. B. da Silva, Marco A. M. Rodrigues, João M. T. Romano, Paulo S. R. Diniz, "The Compression of Electric Signal Waveforms for Smart Grids: State of the Art and Future Trends", IEEE Transactions on Smart Grid, Vol. 5, No. 1, January 2014, pp 291-301.

[65] Jesmin Khan, Sharif M. A. Bhuiyan, Gregory Murphy, Morgan Arline, "Embedded-Zerotree-Wavelet-Based Data Denoising and Compression for Smart Grid", IEEE Transactions on Industry Applications, Vol. 51, No. 5, September/October 2015, pp. 4190- 4200.

[66] Jesmin Khan, Sharif Bhuiyan, Gregory Murphy, Johnathan Williams, "Data Denoising and Compression for Smart Grid Communication", IEEE Transactions on Signal and Information Processing over Networks, Vol. 2, No. 2, June 2016, pp. 200-214.

[67] Joseph Euzebe Tate, "Preprocessing and Golomb–Rice Encoding for Lossless Compression of Phasor Angle Data", IEEE Transactions on Smart Grid, Vol. 7, No. 2, March 2016, pp. 718- 728.

[68] Norman C. F. Tse, John Y. C. Chan,Wing-Hong Lau, Jone T. Y. Poon, L. L. Lai, "Real-Time Power-Quality Monitoring With Hybrid Sinusoidal and Lifting Wavelet Compression Algorithm", IEEE Transactions on Power Delivery, Vol. 27, No. 4, October 2012, pp. 1718- 1726.

[69] S. M. Shafiul Alam, Balasubramaniam Natarajan, Anil Pahwa, "Distribution Grid State Estimation from Compressed Measurements", IEEE Transactions on Smart Grid, Vol. 5, No. 4, July 2014, pp. 1631- 1642.

[70] Andreas Unterweger, Dominik Engel, "Resumable Load Data Compression in Smart Grids", IEEE Transactions on Smart Grid, Vol. 6, Issue 2, pp 919-929, March 2015.

[71] Xing Tong, Chongqing Kang, Qing Xia, "Smart Metering Load Data Compression Based on Load Feature Identification", IEEE Transactions on Smart Grid, Vol. 7, No. 5, September 2016, pp. 2414- 2422.

[72] Sarasij Das, P S Nagendra Rao, "Arithmetic coding based lossless compression schemes for power system steady state operational data", Electrical Power and Energy Systems, Vol.43, No. 1, December 2012, pp. 47- 53.

[73] Wei Jhih Wang, Chang Hong Lin, "Code Compression for Embedded Systems Using Separated Dictionaries", IEEE Transactions On Very Large Scale Integration (VLSI) Systems, Vol. 24, No. 1, January 2016, pp. 266- 275.

[74] Chacko John Deepu, Xiaoyang Zhang, Chun Huat Heng, Yong Lian, "A 3-Lead ECG-on-Chip with QRS Detection and Lossless Compression for Wireless Sensors", IEEE Transactions on Circuits and Systems—II: Express Briefs, Vol. 63, No. 12, December 2016, pp. 1151-1155.

[75] Ching-Hsing Luo, Wei-Jhe Ma, Wen-Ho Juang, Shin-Hung Kuo, Chih-Yuan Chen, Pei-Chen Tai, Shin-Chi Lai, "An ECG Acquisition System Prototype Design With Flexible PDMS Dry Electrodes and Variable Transform Length DCT-IV Based Compression Algorithm", IEEE Sensors Journal, Vol. 16, No. 23, December, 2016, pp. 8244- 8254.

[76] Ericson Chua, Wai-Chi Fang, "Mixed Bio-Signal Lossless Data Compressor for Portable Brain-Heart Monitoring Systems", IEEE Transactions on Consumer Electronics, Vol. 57, No. 1, February 2011, pp. 267-273.

[77] S. K. Mukhopadhyay, M. Mitra, S. Mitra, "A lossless ECG data compression technique using ASCII character encoding", Computers and Electrical Engineering, Vol. 37, Iss. 4, July 2011, pp. 486-497.

[78] S. K. Mukhopadhyay, M. Mitra, S. Mitra, "An ECG signal compression technique using ASCII character encoding", Measurement, Vol. 45, Iss. 6, July 2012, pp.1651-1660.

[79] S. K. Mukhopadhyay, M. Mitra, S. Mitra, "ECG signal compression using ASCII character encoding and transmission via SMS", Biomedical Signal Processing and Control, Vol. 8, No. 4, July 2013, pp 354- 363.

[80] Butta Singh, Dixit Sharma ,Manjit Singh ,Dilbag Singh, "An Improved ASCII Character Encoding Method for Lossless ECG Compression", Advances in Biomedical Science and Engineering, Vol. 1, No. 2, December 2014, pp. 1- 11. (Online) Available: www.scipublish.com/journals/ABSE/papers/download/3402-958.pdf.

[81] A.Y. Saber, T. Senjyu, A. Yona, T. Funabashi, "Unit commitment computation by fuzzy adaptive particle swarm optimization", IET Generation, Transmission and Distribution, Vol. 1, No. 3, May 2007, pp. 456-465.

[82] Wu Jiekang, Zhu Jianquan, Chen Guotong, Zhang Hongliang, "A Hybrid Method of Optimal Scheduling of Short-Term Electric Power Generation of Cascaded Hydroelectric Plants Based on Particle Swarm Optimization and Chance Constrained Programming", IEEE Transactions on Power Systems, Vol. 23, No. 4, November 2008, pp. 1570- 1579.

[83] Yun-Won Jeong, Jong-Bae Park, Se-Hwan Jang, Kwang Y. Lee, "A New Quantum-Inspired Binary PSO: Application to Unit Commitment Problems for Power Systems", IEEE Transactions on Power Systems, Vol. 25, No. 3, August 2010, pp. 1486- 1495.

[84] V. Karthikeyan, S. Senthilkumar, V. J. Vijayalakshmi, "A New Approach to the Solution of Economic Dispatch using Particle Swarm Optimization with Simulated Annealing", International Journal on Computational Sciences & Applications (IJCSA) Vol. 3, No. 3, June 2013, pp. 37-49.

[85] Nimish Kumar, Uma Nangia, Kishan Bhushan Sahay, "Economic Load Dispatch Using Improved Particle Swarm Optimization Algorithms", Proc. of 6th IEEE Power India International Conference 2014 (PIICON 2014), 5th- 7th December, 2014, Delhi, India, pp. 1-6.

[86] M. Suman, M. Venu Gopala Rao, A. Hanumaiah, K. Rajesh, “Solution of Economic Load Dispatch problem in Power System using Lambda Iteration and Back Propagation Neural Network Methods”, International Journal on Electrical Engineering and Informatics, Vol. 8, No. 2, June 2016, pp. 347-355. (Online) Available: http://www.ijeei.org/docs-17898078505775e98de0ee0.pdf.

[87] Ze-Nian Li, Mark S. Drew, Jiangchuan Liu, “Fundamentals of Multimedia”, 2nd Edition, Springer, 2014.

[88] Subhra J. Sarkar, Kalyan Kar, Indranil Das, “Basic Arithmetic Coding based Approach for Compressing Generation Scheduling Data Array”, Proc. of 2017 IEEE Calcutta Conference (CALCON 2017), 2nd - 3rd December, 2017, Kolkata, India, pp. 21-25.

[89] Mamta Sharma, “Compression using Huffman Coding”, International Journal of Computer Science and Network Security (IJCSNS), Vol.10, No.5, May 2010, pp 133-141.

[90] Nilkesh Patra, Sila Siba Sankar, “Data Reduction by Huffman Coding and Encryption by Insertion of Shuffled Cyclic Redundancy Code”, B. Tech. Thesis, N.I.T Rourkela, 2007.

[91] S. J. Sarkar, B. Das, T. Dutta, P.Dey, A. Mukherjee, “An Alternative Voltage and Frequency Monitoring Scheme for SCADA based Communication in Power System using Data Compression”, Proc. of International Conference and Workshop on Computing and Communication (IEMCON)- 2015, 15th – 17th October, 2015, Vancouver, Canada, pp. 1- 7.

[92] Subhra J. Sarkar, Palash K. Kundu, Gautam Sarkar, “Performance Analysis of Resumable Load Data Compression Algorithm (RLDA) for Power System Operational Data”, Proc. of 2017 IEEE Calcutta Conference (CALCON 2017), 2nd - 3rd December, 2017, Kolkata, India, pp. 16-20.

[93] Subhra J. Sarkar, Palash K. Kundu, Gautam Sarkar, “DBEA: A Novel Approach of Repetitive Data Array Compression for Power System Applications”, Proc. of 2nd International Conference for Convergence in Technology (I2CT, 2017), 7th – 9th April, Pune, India, pp. 1-4.

[94] Subhra J. Sarkar, Palash K. Kundu, Gautam Sarkar, “Development of Lossless Compression Algorithms for Power System Operational Data”, IET Generation, Transmission and Distribution, Vol. 12, Iss. 17, September 2018, pp. 4045-4052.

[95] Ye Yan, Yi Qian, Hamid Sharif, David Tipper, “A Survey on Smart Grid Communication Infrastructures: Motivations, Requirements and Challenges”, IEEE Communications Surveys & Tutorials, Vol. 15, Iss. 1, First Quarter 2013, pp. 5-20.

[96] Murat Kuzlu, Manisa Pipattanasomporn, Saifur Rahman, “Communication network requirements for major smart grid applications in HAN, NAN and WAN”, Computer Networks, Vol. 67, July 2014, pp. 74-88.

[97] V. C. Gungor, F. C. Lambert, “A Survey on Communication Networks for Electric System Automation”, Computer Networks, Vol. 50, Iss. 7, May, 2006, pp. 877-897.

[98] Subhra Jyoti Sarkar, “Development of Power Line Carrier Communication based Data Acquisition System”, M.E.E. Thesis, Jadavpur University, June, 2013.

[99] Arati Kurde, V. S. Kulkarni, “IOT Based Smart Power Metering”, International Journal of Scientific and Research Publications, Vol. 6, Iss. 9, September 2016, pp. 411-415. (Online) Available: http://www.ijsrp.org/research-paper-0916/ijsrp-p5757.pdf.

[100] Chen Hongqian, Hongwei Xin, Teng Guanghui, Meng Chaoying, Du Xiaodong, Mao Taotao, Wang Cheng, “Cloud-based data management system for automatic real-time data acquisition from large scale laying-hen farms”, International Journal of Agricultural and Biological Engineering, Vol. 9, No. 4, pp. 106-115. (Online) Available: https://ijabe.org/index.php/ijabe/article/view/2488/pdf.

[101] Dan Raphaeli, “Spread Spectrum Communication System utilizing Differential Code Shift Keying”, US Patent 6064695, 16th May, 2000.

[102] DCSK Modulation. Yitran Technologies website. (Online) Available: http://www.yitran.com/index.aspx?id=3387.

[103] J. Jayasamraj, “SCADA Communication & Protocols”. (Online) Available: http://srldc.in/var/NRC/SRLDC%20Fees%20Feb%202013/Trg%20Psti%20System%20Operator%20Trg/PSO%20PSTI%20%20517%20Sep2011%20Adisesha/ppts/Protocols/SCADA% 20Communications%20and%20Protocols.pdf.

[104] Peter Ceferin, Matija Naglic, Andrej Souvent, “Wide Area Monitoring Systems and Information and Communication Technology Networks (WAMS systems and ICT networks)”. (Online) Available: http://www.smart-com.si/wp-content/uploads/2017/01/Wide-Area-Monitoring-Systems-and-Information-and-Communication-Technology NetworksWAMS-systems-and-ict-networks.pdf.

[105] Consultation Paper on Cloud Computing, Telecom Regulatory Authority of India (TRAI). (Online) Available: https://www.trai.gov.in/sites/default/files/Cloud_Computing_Consultation_paper_10_june_2016.pdf.

[106] Larry Dignan, "Top cloud providers 2018". [Online] Available: https://www.zdnet.com/article/cloud-providers-ranking-2018-how-aws-microsoft-google-cloud-platform-ibm-cloud-oracle-alibaba-stack/.

[107] Geng Wu, Shilpa Talwar, Kerstin Johnsson, Nageen Himayat, Kevin D. Johnson, "M2M: From mobile to embedded internet", IEEE Communications Magazine, Vol. 49, Iss. 4, April 2011, pp. 36-43.

[108] Jayavardhana Gubbi, Rajkumar Buyya, Slaven Marusic, Marimuthu Palaniswami, "Internet of Things (IoT): A Vision, Architectural Elements, and Future Directions". (Online) Available: https://arxiv.org/ftp/arxiv/papers/1207/1207.0203.pdf.

[109] Subhra J. Sarkar, Palash K. Kundu, Gautam Sarkar, "Development of Embedded Power System Data Management System", Modelling, Measurement and Control A, Vol. 90, No. 3, September 2017, 279-295.

[110] J. Duncan Glover, Mulukutla S. Sarma, Thomas J. Overbye, "Power System-Analysis & Design", 5th Edition, Cengage Learning, 2012.

[111] P.S. R. Murty, "Operation and Control in Power Systems", B. S. Publication, 2008.

[112] Shubham Tiwari, Ankit Kumar, G.S Chaurasia, G.S Sirohi, "Economic Load Dispatch using Particle Swarm Optimization", International Journal of Application or Innovation in Engineering & Management (IJAIEM), Vol. 2, Iss. 2, April 2013, pp. 476-485. (Online) Available: https://pdfs.semanticscholar.org/c7bc/dde3c90d73cd52345449114e59bed58e315d.pdf.

[113] Susheel Kumar Dewangan, Achala Jain, Dr. A.P. Huddar, "A Traditional Approach to Solve Economic Load Dispatch Problem Considering the Generator Constraints", IOSR Journal of Electrical and Electronics Engineering (IOSR-JEEE), Vol. 10, Iss. 2, Ver. III, March-April, 2015, pp. 27-32. (Online) Available: https://pdfs.semanticscholar.org/8b73/17b3b3ab4260f668ab049e3a17502e6f162b.pdf.

[114] Maurice Clerc, "Particle Swarm Optimization", 1st edition, ISTE Ltd, 2006.

[115] "Artificial Neural Networks", (Online) Available: http://shodhganga.inflibnet.ac.in/bitstream/10603/108800/16/16_chapter%206.pdf.

[116] I. A. Basheer, M. Hajmeer, "Artificial Neural Networks: Fundamentals, Computing, Design and Application", Journal of Microbiological Methods, Vol. 42, Iss. 1, September, 2000, pp. 3-31.

[117] Laurene Fausett, "Fundamentals of Neural Networks: Architectures, Algorithms and Applications", (Online) Available: http://www.csbdu.in/csbduold/pdf/ Fundamentals%20Of% 20Neural%20Networks. pdf.

[118] S.Sapna, A.Tamilarasi, M. Pravin Kumar, "Backpropagation learning algorithm based on Levenberg Marquardt algorithm", (Online) Available: https://airccj.org/CSCP/ vol2/csit2438.pdf.

[119] S. Usha Rani, C. H. Padmanabha Raju, "A Solution to Unit Commitment Problem via Dynamic Programming and Particle Swarm Optimization", International Journal of Current Engineering and Technology, Vol. 3, No. 4, October 2013, pp. 1495-1503. (Online) Available: http://inpressco.com/wp-content/uploads/2013/10/Paper 621495-1503.pdf.

[120] Subhra J. Sarkar, Palash K. Kundu, "A Novel DCSK Communication Scheme for PLCC based DAS", Proc. of International Conference on Electrical and Computer Engineering (ICECE), 2014, 20th – 23rd December 2014, Dhaka, Bangladesh, pp 50 – 53.

[121] Subhra J. Sarkar, Palash K. Kundu, Ipsita Mondal, "A Novel Data Compression Approach to Reduce the String Size for PLCC based DAS", Proc. of 12th IEEE India International Conference on Electronics, Energy, Environment, Communication, Computer and Control (INDICON-2015), 17th – 19th December, 2015, New Delhi, India, pp 1-5.

[122] Subhra J. Sarkar, Palash K. Kundu, Ipsita Mondal, "Modified DCSK Communication Scheme for PLCC based DAS with Improved Performance", Proc. of International Conference on Control, Instrumentation, Energy and Communication (CIEC), 2016, 28th – 30th January, 2016, Kolkata, India, pp. 456- 460.

[123] Subhra J. Sarkar, Palash K. Kundu, Gautam Sarkar, "Comparison of the performance of DCSK for PLCC with and without data compression", Proc. of International Conference on Intelligent Control Power and Instrumentation (ICICPI), 2016, 21st – 23rd October, 2016, Kolkata, India, pp. 246-250.

[124] Subhra J. Sarkar, Palash K. Kundu, Gautam Sarkar, “Development of DBEA Compressed Data Transfer System over Power Line”, Modelling, Measurement and Control A, Vol. 90, No. 3, September 2017, 262-278.

[125] Dorothy Elizabeth Robling Denning, “Cryptography and Data Security”, 1st edition, Addison-Wesley Publishing Company, 1982. (Online) Available: http://faculty. nps.edu/dedennin/publications/Denning-CryptographyDataSecurity.pdf.

[126] Gary C. Kessler, “An overview of Cryptography”. (Online) Available: https:// www.garykessler.net/library/crypto.html.

[127] William Stallings, “Cryptography and Network Security: Principles and Practice”, Sixth Edition, Pearson, 2014.

[128] William Stallings, “Network Security Essentials: Applications and Standards”, Fourth Edition, Prentice Hall, 2011.

[129] Subhra J. Sarkar, Palash K. Kundu, Gautam Sarkar, “Development of Compressed Encrypted Data Management System for Power System Applications”, Proc. of 2017 IEEE Calcutta Conference (CALCON 2017), 2nd - 3rd December, 2017, Kolkata, India, pp. 11-15.

[130] Wikipedia: Dropbox (service). (Online) Available: https://en.wikipedia.org/ wiki/Dropbox_(service).

[131] Subhra J. Sarkar, Palash K. Kundu, Gautam Sarkar, “Development of Cloud based Power System Operational Data Management System”, IET Generation, Transmission and Distribution, Vol. 13, Iss. 5, March 2019, pp. 644-651.

[132] P. T. Patil, Dattatraya Raut, Sagar Sutar, Sandesh Narute, “GSM Based Wireless Load-Shedding Management System for Non Emergency Condition”, International Journal of Modern Trends in Engineering and Research, Vol. 2, Iss. 1, January, 2015, pp. 594-597.

[133] Tanvira Ismail, Dipesh Karki, Zohir Mollick, Sagar Das, Manoj Deka, “Remote Power Distribution System Using GSM”, International Journal of Advanced Research in Computer and Communication Engineering, Vol. 3, Iss. 4, April 2014, pp. 6289- 6292.