Chapter 2

Modeling and Parameter Estimation of Battery

2.1 Introduction

The battery management system requires an accurate model of the battery for estimation of its internal states such as State-of-Charge (SOC) and State-of-Health (SOH) in EVs. Developing a battery model requires the accurate values of the parameters of the battery. Since the parameters change with the variation in temperature, SOC and C-rate, it is necessary to analyze the effect of parameter change on battery model [212]. Especially in the applications of EVs, the change in parameters will be more significant than other applications because the C-rate and temperature changes are relatively higher [37]. The primary purpose of this chapter is to introduce a simple and efficient methodology for accurate battery modeling and the identification of the parameters of the battery. The combined equivalent circuit model incorporates the benefits of Thevenin-based, impedance-based and runtime-based model. Hence, it has been utilized to mimic the dynamic behavior of battery during various operating conditions. Voltage response circuit part of combined equivalent circuit is composed similarly to the Thevenin battery model which contain various active and passive circuit elements to represent the electrochemical process taking place inside the battery. Detailed modeling of voltage response circuit considering the dependency of the battery on SOC, C-rate and temperature have been discussed in the Section 2.2. Dependency of model circuit elements has been described with a combination of exponential functions and 2^{nd} to 6^{th} order polynomial functions [213, 214]. In the literature [89–91], parameters of the equations are identified using curve fitting on the data collected using

experimentation on hardware. These experiments are time-consuming and require substantial financial investment. The proposed method is an alternative way to identify the parameters of the battery using voltage characteristic data provided by the manufactures. The proposed parameter identification approach has been formulated as an optimization problem which has been discussed in detail in the Section 2.3. This optimization problem is a high-dimensional and non-linear problem which requires a intensive computation overhead for the solution using conventional methods. According to No Free Lunch (NFL) theorem [211], a single optimization approach could not be considered suitable for solving all optimization problems. Hence, the performance of six heuristic techniques for estimation of battery parameters has been compared under similar execution conditions. Out of these six heuristic techniques, Genetic Algorithm (GA) and Particle Swarm Optimization (PSO) are time-tested techniques for optimization problems while Different Evolution (DE), Teaching Learning Based Optimization (TLBO), Grey Wolf Optimization (GWO) and Ageist Spider Monkey Optimization (ASMO) are state-of-the-art algorithms. An overview of heuristic optimization algorithms utilized for the purpose of identification of parameters has been discussed in Section 2.4. Following this, simulation results using the proposed method for parameters estimation have been discussed in detail in the Section 2.5. The efficacy of the proposed method has been validated by comparing the obtained results with battery parameters derived from standard Pulse-Current Charge-Discharge Test (PCDT). Further, the performance of all heuristic optimization approaches have been examined by the quality of the solution, the rate of convergence, computational error and statistical significance using both parametric (*t-test*) and non-parametric tests (Wilcoxon test) in the Section 2.5.3. Finally, the concluding remarks on the work carried out in this chapter have been summarized in the Section 2.6.

2.2 Equivalent circuit model of Li-ion batteries

Equivalent circuit based battery model conceptualizes the electrochemical mechanism of the battery using the SOC dependent voltage source, resistors, capacitor and non-linear components like Warburg impedance and diodes to approximate the dynamics of the battery. A combined equivalent circuit model incorporates the energy balance circuit and voltage response circuit as shown in Figure 1.7 for representing the dynamic behaviour of the battery. The voltage response circuit represents the transient dynamics of the battery. It is similar to Thevenin-based battery model as shown in Figure 2.1. It consists of a voltage source V_{oc} which denotes the Open Circuit Voltage (OCV), ohmic resistance R_0, electrochemical diffusion process resistance and capacitance represented by R_n and C_n

respectively. Energy balance circuit models battery runtime and DC response by utilizing capacitor C_{use}, resistance R_{dis} and current-controlled current source I_{BL}. The values of circuit components vary over a large range during the runtime time of the battery. These values depend upon many operating conditions such as SOC, C-rate (capacity normalized current), temperature, number of cycles and hysteresis effect.

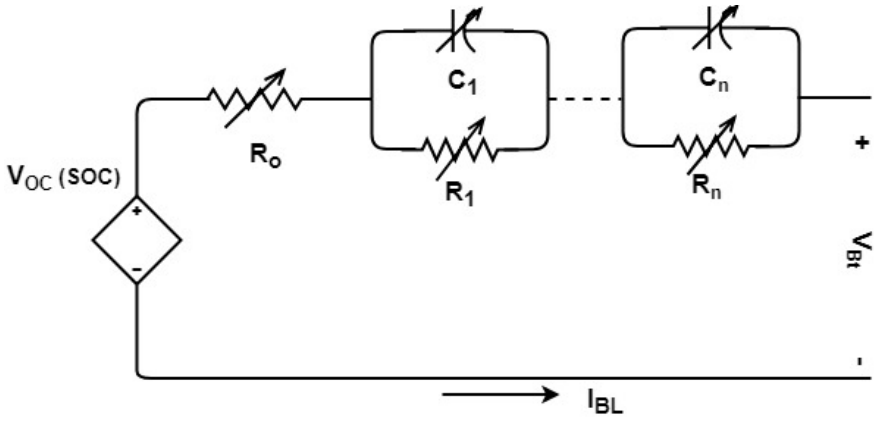


FIGURE 2.1: Voltage response part of the combined equivalent circuit based battery model with 1 RC network

Practical equivalent circuit modeling demands that the components of the voltage response circuit part should be related with the electrochemical process in the battery. In [66], the authors suggested that the relationship between the battery model parameters and the electrochemical process can be obtained by comparing given battery model with the impedance-based model. Internal impedance calculated using Electrochemical Impedance Spectroscopy (EIS) measurements were used to form 2^{nd} order Randles circuit as shown in Figure 1.4. Here, bulk resistance represented by R_{bk} denotes the electric conductivity of the separator, electrodes, and the electrolyte. Surface film layer electrode resistance and capacitance are represented by R_{sei} and C_{sei} respectively. Charge transfer resistance is represented by R_{ct}, double layer capacitance is represented by C_{dl} and diffusion process of the battery is represented by Warburg impedance Z_W. In real-time EV application, the sampling rate of data acquisition should be at least 1 Hz, since most of the existing driving cycle also have 1 Hz speed-time signals [215]. However, the energy consumption of a EV is evaluated at high sampling rates (1 Hz-10 Hz) to achieve more accurate and reliable characterization of EV efficiency [216] as dynamics of the battery of the vehicles can be evaluated more accurately at the high sampling rate [66]. Hence, this sampling rate of the data acquisition system causes high frequency R_{sei} and C_{sei} indistinguishable from R_{bk}. In the voltage response circuit R_{bk} is represented by R_0 [217]. Additionally,

the Z_W is approximated by the n number of parallel resistor-capacitor network where n defines the order of voltage response circuit.

Equivalent circuit based battery model has gained a lot of interest amongst battery management system designers for parameter estimation purpose because of it's simplified mathematical and numerical approach that minimizes the necessity for the computationally intensive procedure. It gives a good trade-off between exactness, complexity, and usability depending upon different RC networks used for voltage response circuit part of the battery model. In the research literature, different forms of Thevenin's equivalent circuit based battery model depending on the number of RC pairs, the effect of hysteresis and impact of aging, have been developed by researchers [56, 68]. In [218], the author compared complexity versus modeling error for the Thevenin RC network circuit models, by varying the number of RC pairs and concluded that increasing the number of RC pairs decreases the error and increases the computation time. In [39, 68], the authors utilized multi-objective particle swarm optimization to compare the practicality of different equivalent circuit based model incorporating model accuracy and complexity. The comparison between different equivalent circuit based models concluded that the first-order RC model is most suitable for a Li-ion battery. The addition of more RC networks generally improves the model accuracy but increases the computational burden. Hence, the first-order RC model has been utilized for identification of battery model parameters.

The empirical non-linear equations utilized to describe the transient behavior of the battery were not alike. The researchers have modeled components with a combination of exponential functions and 2^{nd} to 6^{th} order polynomial functions [213, 214]. Some researchers also considered the influence of stress factors, such as hysteresis effect and temperature. Consequently, depending on effect of the stress factor, the accuracy, and complexity of the model change. Equivalent circuit model of a Li-ion battery is presented in [72, 219], consisting of empirical non-linear equations extracted from measurement data obtained from experimental analysis. The first-order RC model parameters which are dependent on state-of-charge, C-rate, and temperature are formulated using a polynomial equation as given in equations (2.1)-(2.4), to exhibit the dynamic electrical performance of the battery [66, 72].

$$R_0 =((a_1 + a_2x + a_3x^2)e^{a_4y} + (a_5 + a_6x + a_7x^2))e^{a_{32}(\frac{1}{T_o}-\frac{1}{T_r})} \tag{2.1}$$

$$R_1 =((a_8 + a_9x + a_{10}x^2)e^{a_{11}y} + (a_{12} + a_{13}x + a_{14}x^2))e^{a_{33}(\frac{1}{T_o}-\frac{1}{T_r})} \tag{2.2}$$

$$C_1 = -((a_{15} + a_{16}x + a_{17}x^2)e^{a_{18}y} + (a_{19} + a_{20}x + a_{21}x^2))e^{a_{34}(\frac{1}{T_o}-\frac{1}{T_r})} \tag{2.3}$$

$$V_{oc} =(a_{22} + a_{23}x + a_{24}x^2)e^{a_{25}y} + (a_{26} + a_{27} + a_{28}y^2 + a_{29}y^3) - a_{30}x + a_{31}x^2 \tag{2.4}$$

Here, $a_1, a_2...a_{34}$ represent the polynomial coefficients to be determined; T_0 and T_r denote the instantaneous and reference temperature of the battery respectively. The transient behavior of the battery is different for charging and discharging process. Hence, for the charging process, x (C-rate) and y (current charge state in battery) are substituted by charging current rate C_r and SOC and for discharging process by discharge current rate D_r and $1 - DOD$ respectively. DOD represents depth of discharge which decreases with increase in model terminal voltage V_{Bt}^M. The C-rate for charging process (C_r) and discharging process (D_r) can be expressed as [104]:

$$C_r = \frac{I_{BL,c}}{Q_{nom}}; \qquad D_r = \frac{I_{BL,d}}{Q_{nom}}; \tag{2.5}$$

Here, Q_{nom} is the nominal battery capacity provided by the manufacturer; $I_{BL,c}$ is current flowing through battery terminals during charging process and $I_{BL,d}$ is current flowing through battery terminals during discharging process.

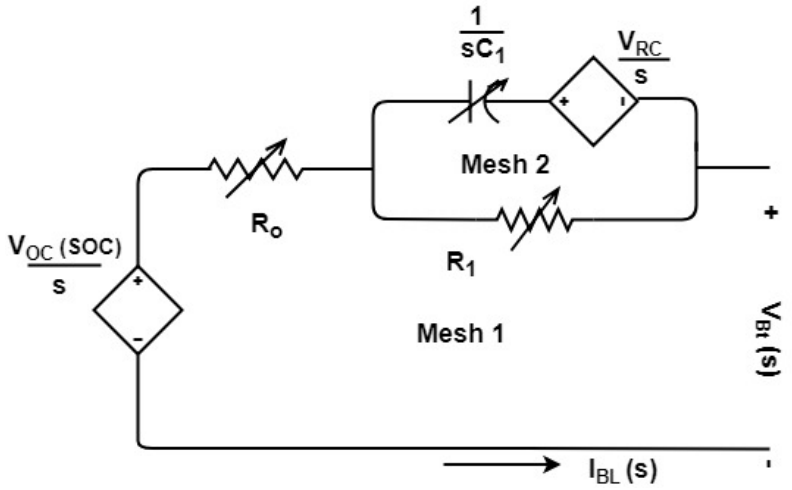


FIGURE 2.2: Battery Model with 1 RC network represented in s-domain

The equation for terminal voltage is derived using Laplace transform of the voltage response circuit. Figure 1.7 illustrates the s-domain representation of the voltage response circuit part of the battery model. Using Krichhoff's law, the dynamics of the circuit can be expressed as follows:
In the mesh 1:

$$-V_{Bt}^M(s) + \frac{V_{oc}}{s} - I_{BL}(s)R_0 - \Big[I_{BL}(s) - I_{RC}(s)\Big]R_1 = 0 \tag{2.6}$$

$$-V_{Bt}^M(s) + \frac{V_{oc}}{s} - I_{BL}(s)[R_0 + R_1] + I_{RC}(s)R_1 = 0$$

$$I_{RC}(s) = \frac{V_{Bt}^{M}(s)}{R_1} - \frac{V_{oc}}{R_1 s} + I_{BL}(s)\left[\frac{R_0}{R_1} + 1\right] \tag{2.7}$$

In the mesh 2:

$$-V_{Bt}^{M}(s) + \frac{V_{oc}}{s} - I_{BL}(s)R_0 - \frac{I_{RC}(s)}{C_1 s} - \frac{V_{RC}}{s} = 0 \tag{2.8}$$

substituting value of I_{RC}

$$-V_{Bt}^{M}(s) + \frac{V_{oc}}{s} - I_{BL}(s)R_0 - \frac{V^{M}(s)}{R_1 C_1 s} + \frac{V_{oc}}{R_1 C_1 s^2} - \frac{I_{BL}(s)}{C_1 s}\left[\frac{R_0}{R_1} + 1\right] - \frac{V_{RC}}{s} = 0 \tag{2.9}$$

$$V_{Bt}^{M}(s) = \frac{V_{oc}}{s} - V_{RC}\left[\frac{R_1 C_1}{1 + R_1 C_1 s}\right] - I_{BL}(s)\left[R_0 + \frac{R_1}{1 + R_1 C_1 s}\right]$$

$$V_{Bt}^{M}(s) = \frac{V_{oc}}{s} - V_{RC}\left[\frac{1}{s + \frac{1}{R_1 C_1}}\right] - I_{BL}(s)\left[R_0 + \frac{1}{C_1}\Big(\frac{1}{s + \frac{1}{R_1 C_1}}\Big)\right]$$

using inverse Laplace transform:

$$V_{Bt}^{M}(t) = V_{oc} - V_{RC}e^{\frac{-t}{R_1 C_1}} - I_{BL}(t)R_0 - \frac{1}{C_1}\int_{\zeta=0}^{\zeta=t} I_\zeta e^{\frac{-(t-\zeta)}{R_1 C_1}} d\zeta \tag{2.10}$$

$$V_{Bt}^{M}(t) = V_{oc} - V_{RC}e^{\frac{-t}{R_1 C_1}} - I_{BL}R_0 - \frac{I_{BL}}{C_1}\left[R_1 C_1 e^{\frac{-(t-\zeta)}{R_1 C_1}}\right]_{\zeta=0}^{\zeta=t} \tag{2.11}$$

$$V_{Bt}^{M}(t) = V_{oc} - V_{RC}e^{\frac{-t}{R_1 C_1}} - I_{BL}R_0 - I_{BL}R_1\Big[1 - e^{\frac{-t}{R_1 C_1}}\Big] \tag{2.12}$$

For the constant current in the circuit, initial value of voltage across capacitor is represented by $V_{RC} = \frac{Q(0)}{C_1}$ where $Q(0)$ is the initial charge across the capacitor.

$$V_{Bt}^{M}(t) = V_{oc} - \frac{Q(0)}{C_1}e^{\frac{-t}{R_1 C_1}} - I_{BL}R_0 - I_{BL}R_1\Big[1 - e^{\frac{-t}{R_1 C_1}}\Big] \tag{2.13}$$

Time-dependent nonlinear relation of the output terminal voltage V_{BL}^{M} for the constant current condition can be determined using equations (2.1)-(2.4). Output terminal voltage is obtained by rearranging equation (2.13) and it follows a polynomial equation as: [48] :

$$V_{Bt}^{M} = \left(\frac{Q(0)}{C_1} + I_{BL}R_1\right)e^{-t/R_1 C_1} + V_{oc} - I_{BL}(R_0 + R_1) \tag{2.14}$$

Here t is sample time interval; I_{BL} is charging ($-$ve)/discharging(+ve) current.

The modeling of a battery is an offline problem. An accurate model mimics the behavior of the battery under different operating conditions. Once a model has been estimated, it is deployed in battery management system for online estimation of the states of the battery. With the help of empirical equation discussed in this section, battery model with describes the transient behavior of the battery at different SOC, C-rate and temperature values. However, to utilize this model for further estimation of battery internal states it is required to identify the parameter values of the empirical equation. The objective of parameter identification is achieved by minimizing the error between expected catalog voltage and the evaluated voltage from battery model at the different operating condition.

2.3 Objective function for optimization process

For battery modeling, one of the key challenges is to identify model parameters accurately. Battery parameters can be identified by experimentation on hardware-setup and analysis either in the frequency domain or in time domain [220]. The EIS testing method is performed by applying a small amplitude ac current of varying frequency to the battery and calculating impedance using voltage response [83]. The parameters of the battery are identified by impedance analysis in the frequency domain. Similarly, battery model parameters are identified using curve fitting on the data collected using experimentation on hardware using standard PCDT. These approaches are expensive and requires specialized electronic equipment to perform the experiment. The requirement of expertise and high computing power hinders the applicability of the approach [83]. Model-based estimation methods such as variants of Kalman filters [210], and neural network-based methods analyze voltage responses in the time domain for varying operating conditions. Variants of Kalman filters require less computation power, but these methods assume model equation to be linear (or slightly non-linear); whereas battery parameter estimation problem is a high dimensional non-linear problem. Neural network based methods require a large amount of training data and high computational power to properly estimate model parameters. Various tests have to be performed under various operating conditions in a laboratory to obtain voltage responses for modeling battery using these methods. To avoid time-consuming experimentation which requires substantial financial investment, parameter estimation is formulated as an optimization problem to identify parameters of battery using voltage characteristic data provided by manufactures. The battery parameter modeling is a high dimensional non-linear problem which require intensive computation overhead for solution using conventional methods. According to NFL theorem [211], a single optimization approach could not be considered suitable for solving all optimization problems, hence, the performance

comparison for six heuristic techniques has been examined. The motivation behind adopting mathematical optimization is to determine the coefficients of the polynomial equation that are used to describe the parameters of the first order RC model given in equations ((2.1)-(2.4)). The objective function evaluation was done in terms of minimizing the difference between the expected catalog voltage and the evaluated voltage curve at various temperatures. Thus this objective function was expressed as the Manhattan Euclidean distance between value of the output voltage obtained from described battery model and the voltage values given by the manufacturer at each sampling interval for both charging and discharging process. The objective function can be mathematically expressed as:

$$min\ F(x) = \sum_{i=1}^{n} \left| V_{Bt,i}^{M} - V_{Bt,i}^{C} \right| \tag{2.15}$$

Here $V_{Bt,i}^{C}$ denotes the catalog value given by the manufacturer and $V_{Bt,i}^{M}$ denotes the output voltage value obtained from battery model at different SOC (or DOD) values.

The detailed description of various optimization algorithms utilized for identification of the optimal values of coefficients of the polynomial equation has been included in the following section. The concise information has been incorporated to familiarize with basics of heuristic algorithms, while the pseudo-codes assists in understanding similarities and differences in their implementation.

2.4 Meta-heuristic optimization algorithms for battery modeling

The battery parameter optimization is a high dimensional multimodal optimization problem, and different stochastic optimization algorithms yield different approximations of the optimal solution. Hence, six different meta-heuristic optimization techniques are utilized to determine the optimal values of polynomial coefficients $a_1, a_2 ... a_{34}$ for identification of parameters of the battery model. The objective function, described in the previous section, was modeled to incorporate the effect of various temperature levels, SOC and C-rate on the battery modeling. The generalized problem formulation of parameter estimation of the battery and its pseudo code has been added. A suitable meta-heuristic algorithm (2-7) can be incorporated in step 12 of algorithm 1 to update the population.

Algorithm 1 Pseudocode to estimate battery parameters

1: Read the battery data from manufacturer catalog for different temperature.
2: Define the upper bound (u_{bd}) and lower bounds (l_{bd}) for each parameter cofficient (a_d) for $d = 1, 2, ...34$.
3: Define Maximum iteration ($maxiter$).
4: Randomly initialize the population, represented by P solution sets, within the search space defined by upper and lower bounds i.e. $x_p = [a_{p,1}, a_{p,2}...a_{p,34}]$, $p = 1, 2....P$. Here $l_{bd} \leq a_{p,d} \leq u_{bd}$, $d = 1, 2, 3....34$.
5: **for** $it = 0 : maxiter$ **do**
6: **for** $p = 1 : P$ **do**
7: Calculate R_0, R_1, C_1, V_{oc} and V_{Bt}^M using eq. (2.1)-(2.4), for p^{th} solution (x_p).
8: Calculate fitness $f_p = F(x = x_p)$ for p^{th} solution (x_p).
9: **end for**
10: $f^* = F(x = x^*)$
11: Update the population x_p ($p = 1, 2....P$) using one of the optimizer algorithm: GA(Algorithm 2), PSO(Algorithm 3), DE(Algorithm 4), TLBO(Algorithm 5), GWO(Algorithm 6) or ASMO(Algorithm 7).
12: **end for**

2.4.1 Genetic Algorithm (GA)

GA was introduced by Holland et al. in 1988 [221] based on the principle of natural selection and genetics. The process of natural selection starts with the selection of fittest individuals from a population. In GA, a candidate solution to the optimization problem is represented in the form of strings referred as chromosomes. The process is launched by randomly initiating a set of chromosomes across the search space accompanied by creating offspring population using a set of genetic operators such as selection, crossover, and mutation to evaluate the fitness value over successive generations. The method of implementing the algorithm is described in Algorithm 2.

In the algorithm, P_s is the number of selected optimal population that are retained in the next iteration. r_1, r_2, r_c and r_m are random variables between 0 and 1. r_c and r_m used for selection of crossover and mutation process. The various parameters affecting the performance of GA were taken as follows: population size (P) was taken as 1000, P_s was taken as 100, crossover rate (C_{prob}) as 0.8 and mutation rate (M_{prob}) was selected to be 0.2. GA was initially developed to solve combinatorial optimization problems. Although this makes it efficient in solving discrete optimization problems, but not that good for continuous problems. By encoding the chromosomes in binary form and using binary mutation and binary crossover, the GA can be used for discrete optimization problems. For example, in a generation two parent chromosome encoded as: 000000 and 111111 can result in offspring 000111 and 111000 in next generation through crossover after third

Algorithm 2 Solution update using GA

Selection: Sort the population on the basis of fitness $F(x_p)$.
Crossover:
for $p = P_s + 1 : P$ **do**
    Select 2 solutions randomly from optimal solutions (P_s): $a, b \epsilon [1, P_s]$.
    for $d = 1 : D$ **do**
        if $r_1 < C_{prob}$ **then**
            $x_p[d] = r_c.x_a[d] + (1 - r_c).x_b[d]$
        else
            $x_p[d] = x_a[d]$
        end if
    end for
end for
Mutation:
for $p = 1 : P$ **do**
    for $d = 1 : D$ **do**
        if $r_2 < M_{prob}$ **then**
            $x_p[d] = l_{bd} + r_m * (u_{bd} - l_{bd})$
        end if
    end for
end for

binary digit. Mutation can be implemented by swapping some of the digits from 0 to 1 and vice-versa.

2.4.2 Particle Swarm Optimization (PSO) Algorithm

PSO, introduced by Kennedy et al. in 1995 [222], is a swarm intelligence based method simulating the intelligent movement and information-sharing behavior among the population of different species throughout their food search. The optimization process comprises of a population of particles that operate collectively through velocity and position update influenced by personal best location and group past location to converge towards the optimal solution. The method of implementing the algorithm is described in Algorithm 3.

Here v_p is the velocity of the p^{th} solution; w is an inertia weight factor that linearly decreases from 0.9 to 0.1; c_1 and c_2 are acceleration constants with a value of 2 in this case; r_1 and r_2 are stochastic component of the optimization and are chosen randomly between 0 and 1 in each iteration; pb_p is the personal best position found by the p^{th} solution and gb is the global best position found so far. The population size (P) was taken as 40. PSO algorithm is more suitable for continuous optimization problems and is easy to implement, but compared to GA, PSO lacks in solving discrete problems. Use of continuous values of

Algorithm 3 Solution update using PSO

for $p = 1 : P$ **do**
  $v_p = w.v_p + c_1.r_1.(pb_p - x_p) + c_2.r_2.(gb - x_p)$
  $x_p = x_p + v_p$
  if $F(x_p) < F(pb_p)$ **then**
    $pb_p = x_p$
  end if
  if $F(x_p) < F(gb)$ **then**
    $gb = x_p$
  end if
end for

inertia weight factor (w) and acceleration constants (c_1 and c_2) makes it difficult to use velocity update equation for solution update in discrete optimization problems.

2.4.3 Differential Evolution (DE) Algorithm

DE was introduced by Storn et al. in 1997 [223] and is an evolutionary optimization algorithm. The candidate population moves in the search space iteratively by three evolutionary steps (Mutation, Crossover and Selection) given in equations (2.16), (2.17) and (2.18).

Algorithm 4 Solution update using DE

for $p = 1 : P$ **do**
  Randomly select $a, b, c \, \epsilon [1, P]$.
  $v_{d,p} = x_a + M.(x_b - x_c)$.
  if $r_j \leq CR$ or $j = I_r$ **then**
    $u_p = v_{d,p}$
  end if
  if $r_j > CR$ or $j \neq I_r$ **then**
    $u_p = x_p$
  end if
  if $F(u_p) \leq F(x_p)$ **then**
    $x_p = u_p$
  end if
end for

$$v_{d,p} = x_a + M.(x_b - x_c) \tag{2.16}$$

$$u_p = \begin{cases} v_{d,p} & \text{if } r_j \leq CR \text{ or } j = I_r \\ x_p & \text{if } r_j > CR \text{ or } j \neq I_r \end{cases} \tag{2.17}$$

$$x_p = \begin{cases} u_p & \text{if } F(u_p) \leq F(x_p) \\ x_p & \text{if } F(u_p) > F(x_p) \end{cases} \tag{2.18}$$

Here x_a, x_b and x_c are randomly chosen candidate solutions, M is the mutation factor (between 0 and 2) and v_d is a donor vector. u is the trail vector, r is a random variable between 0 and 1, I_r is random integer between 0 and T, and CR is crossover constant. The population size (P) was taken as 40. The algorithm can be implemented as given in Algorithm 4. Compared to GA and PSO, DE is much more robust as it is good against both discrete and continuous optimization problems. Further, similar to PSO an iteration of DE is much faster than GA and other more complex algorithms described later on. But compared to more advanced algorithms it lacks any capability to avoid stagnation problems.

2.4.4 Teaching Learning Based Optimization (TLBO) Algorithm

TLBO was introduced by Rao et al. in 2012 [224] as a population-based optimization algorithm inspired by the teaching-learning process in a class room. In this algorithm, the influence of a teacher on the output of the learners in a class is evaluated. The algorithm uses two modes of learning: (1) Teacher phase and (2) Learner phase. In this algorithm, a population is a group of learners, design variables of the optimization are the subjects offered and fitness value of optimization are the performance results of learners. The method of implementing the algorithm is described in Algorithm 5.

Here T_F is the randomly chosen teaching factor (its value is either 1 or 2); r and r_p are random variables within range of 0 and 1 and M_s is the mean solution value. The population size (P) was taken as 40. Although TLBO is much harder to implement compared to PSO and DE, but due to implementation of hierarchical group based solution update strategies it is more efficient against complex optimization problems.

Algorithm 5 Solution update using TLBO

$M_s = \frac{1}{P}.\sum_{p=1}^{P} x_p$
for $p = 1 : P$ **do**
    $dM_s = r.(x^* - T_F.M_s)$
    $x_p^{'} = x_p + dM$
    if $F(x_p^{'}) < F(x_p)$ **then**
        $x_p = x_p^{'}$
    end if
end for
for $p = 1 : P$ **do**
    Select q randomly between 1 and P.
    $X_p = \begin{cases} X_p + r_p(X_p - X_q) & if\ F(X_p) < F(X_q) \\ X_p + r_p(X_q - X_p) & if\ F(X_q) < F(X_p) \end{cases}$
end for

2.4.5 Grey Wolf Optimization (GWO) Algorithm

GWO was introduced by Mirjalili et al. in 2014 [225]. It is meta-heuristic algorithm based on social hierarchy, and hunting mechanism of grey wolves. For implementing social hierarchy, four kinds of grey wolves (alpha, beta, delta, and omega) are employed. Following steps are considered to simulate hunting behavior: searching, encircling and attacking prey. The method of implementing the algorithm is described in Algorithm 6.

$$A_p = 2.c_{v,p}.r_1 - c_{v,p} \tag{2.19}$$

$$C_p = 2.r_2 \tag{2.20}$$

Algorithm 6 Solution update using GWO

Update x_α, x_β and x_δ to assign them the best three solution parameters from the population.
for $p = 1 : P$ **do**
    Calculate A_α, A_β and A_δ using (2.19).
    Calculate C_α, C_β and C_δ using (2.20).
    $x_1 = x_\alpha - A_\alpha.\,|C_\alpha.x_\alpha - x_p|$
    $x_2 = x_\beta - A_\beta.\,|_\beta.x_\beta - x_p|$
    $x_2 = x_\delta - A_\delta.\,|C_\delta.x_\delta - x_p|$
    $x = \frac{x_1+x_2+x_3}{3}$
end for

Here, c_v is the control variable that linearly decreased from 2 to 0; A and C are coefficient vectors; r_1 and r_2 are random variables within the range of 0 and 1. The population size

(P) was taken as 40. Compared to TLBO and ASMO, GWO is easier to implement. It is much faster to iterate over the solutions in case of GWO, but it is inefficient against complex optimization problems due to stagnation problems.

2.4.6 Ageist Spider Monkey Optimization (ASMO) Algorithm

Spider monkey optimization was introduced by Bansal et al. in 2014 [226]. It is swarm intelligence-based method simulating foraging behavior of spider monkeys. In 2016, the lead author of the article Sharma et al. [227] proposed a modified version of spider monkey optimization which groups the population of spider monkey based on their ages. The algorithm works by dividing the population into groups. Each group is assigned (based on fitness value) a leader called Local Leader (LL). The overall leader of the population (Global Leader or GL) is the Local Leader with best fitness. All the solutions are updated

Algorithm 7 Solution update using ASMO

$GL = x^*$
Update LL for each solution.
if LL is not changed for LL_{limit} **then**
    Reinitialize the solutions with that LL
end if
if GL is not changed for GL_{limit} **then**
    Split one of the groups and reassign Local Leader(LL) for new groups
end if
for $p = 1 : P$ **do**
    for $d = 1 : D$ **do**

$$x_p^{'}[d] = \begin{cases} x_p[d] + r_1.(LL_p[d] - x_p[d]) & if\ r_2 \geq pr_1 \\ \quad +r_2.(x_r[d] - x_p[d]) & \\ \quad\quad x_p[d] & if\ r_2 < pr_1 \end{cases}$$

    end for
    if $F(x_p^{'}) < F(x_p)$ **then**
        $x_p = x_p^{'}$
    end if
end for
for $p = 1 : P$ **do**
    for $d = 1 : D$ **do**

$$x_p^{'}[d] = \begin{cases} x_p[d] + r_3.(GL_p[d] - x_p[d]) & if\ r_4 \geq pr_2 \\ \quad +r_4.(x_r[d] - x_p[d]) & \\ \quad\quad x_p[d] & if\ r_4 < pr_2 \end{cases}$$

    end for
    if $F(x_p^{'}) < F(x_p)$ **then**
        $x_p = x_p^{'}$
    end if
end for

in two steps based on the local leader and global leader positions. Further, to counter the problem of stagnation, there are two more steps that can be taken. If a particular LL does not move from its position for a particular number of iterations (LL_{limit}), the solutions within it are reinitialized along with reassigning of the LL in that group. Further, if the GL does not move from its position for a particular number of iterations (GL_{limit}), one of the group in the population is split into two parts and each is assigned a LL. In case the number of groups exceeds the limit, the algorithm combines the groups and reassign the local leaders. The method of implementing the algorithm is described in Algorithm 7.

$$pr_2 = 0.1 + 0.9.\frac{F(x^*)}{F(x_p)} \tag{2.21}$$

Here, r_1 and r_3 are random variables within the range of 0 and 1; r_2 and r_4 are random variables within the range of -1 and 1. $pr_1(= 0.5)$ and pr_2 are perturbation rates. pr_2 is calculated as given by equation (2.21).

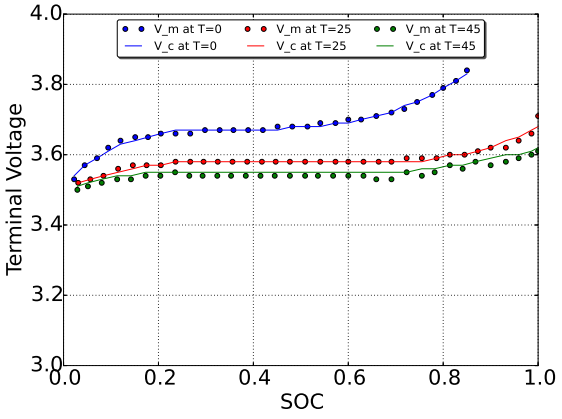
The ASMO algorithm has an inherent group breaking and merging behavior within it which prevents it from stagnating at a local optimum for a long time. In other words, ASMO performs well against complex optimization problems and in most cases it easily avoids stagnation problems due to its efficient multiple group fission-fusion solution update strategies. These group splitting and merger strategies helps the algorithm in finding the better solutions compared to other such algorithms and makes it more robust against complex multi-modal optimization problems. Further, the group based design enables easy parallel implementation of the algorithm to utilize multi-core systems effectively.

2.5 Results and Discussion

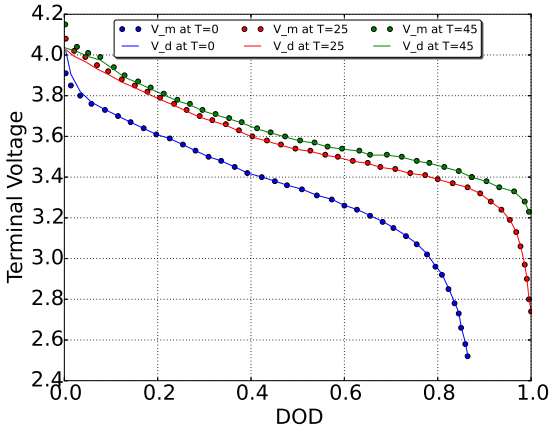
To evaluate the parameters of the battery model with the variation of temperature, a Panasonic 3.1Ah 18650 cylindrical battery was used and operated over a wide range of temperature ($0°C$ to $45°C$) in the simulation. Detail about the battery characteristics values is given in Appendix A.For identification of battery model parameters dependent on C-rate, SOC and temperature, six different meta-heuristic optimization techniques (GA, PSO, DE, TLBO, GWO, and ASMO), as described in last Section were utilized. The efficacy of the proposed method has been validated by comparing the obtained results with battery parameters derived from standard PCDT. Further, the performance of all heuristic optimization approaches have been examined and the conclusion obtained for the specified battery has been tabulated and compared in Section 2.5.3.

2.5.1 Estimated Parameters of the Battery Using Proposed Approach

For the purpose of evaluation, a particular algorithm is executed for ten identical trails on a specific model (charge or discharge) with an individual run consisting of 20,000 function evaluations. To reduce the effect of this randomness in comparing the results of different algorithms, we had to take multiple results with each algorithm. Further, the statistical tests like t-test work well on a sample size less than 30. Hence, considering the limited applicability of statistical tests and keeping reasonable computational time, we set the number of trials for each algorithm to 10. The battery model parameters corresponding to the most optimal solution, evaluated using the objective function value given in equation (2.15), are used for further model investigation. Figure 2.3 (a) and



(a)



(b)

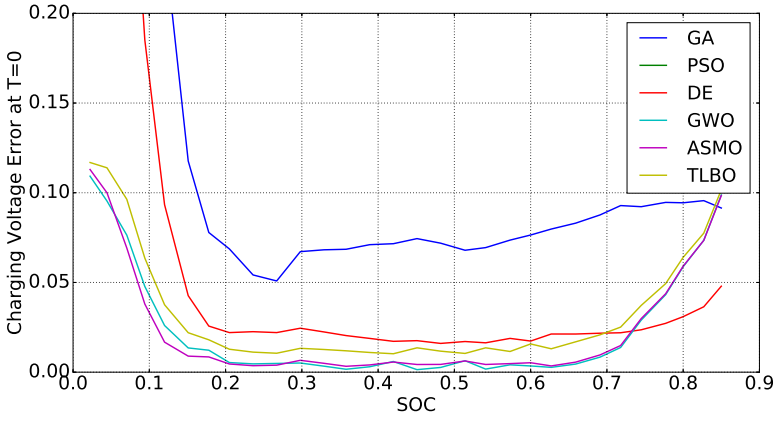
FIGURE 2.3: Predicted Characteristic Curve (a) For Charging (b) For Discharging (c) Error for Charging at $25°C$ (b) Error for Discharging at $25°C$

TABLE 2.1: PC's a_1 to a_{34} for charging and discharging

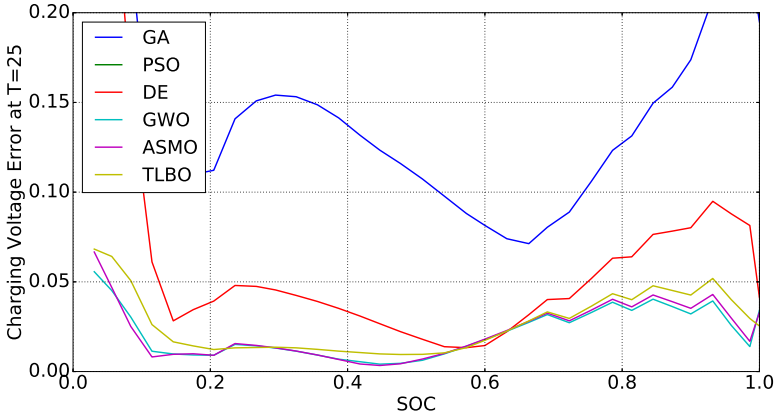
	Charging						**Discharging**					
	GA	**PSO**	**DE**	**TLBO**	**GWO**	**ASMO**	**GA**	**PSO**	**DE**	**TLBO**	**GWO**	**ASMO**
a_1	0.513	0.012	0.237	0.006	0.009	0.007	0.189	0.000	0.038	0.002	0.003	0.019
a_2	0.687	0.012	0.125	0.001	0.003	0.010	0.325	0.025	0.024	0.026	0.009	0.006
a_3	0.175	0.054	0.144	0.005	0.047	0.016	0.336	0.099	0.025	0.013	0.013	0.009
a_4	28.7	16.8	36.1	3.44	0.689	39.3	34.9	37.2	7.21	9.54	4.56	25.10
a_5	0.008	-0.115	0.069	0.233	0.018	0.024	0.066	0.032	0.400	0.018	0.012	0.059
a_6	0.110	-0.021	0.268	0.040	0.354	0.048	0.640	-0.074	0.094	0.027	0.005	0.027
a_7	0.216	0.125	0.017	0.050	0.008	0.008	0.084	0.056	0.167	0.010	0.069	0.015
a_8	0.336	0.468	0.276	0.003	0.033	0.028	0.122	0.096	0.064	0.316	0.000	0.060
a_9	0.493	0.043	0.268	0.009	0.024	0.000	0.553	-0.012	0.005	0.181	0.532	0.236
a_{10}	0.578	0.087	0.725	0.006	0.034	0.002	0.513	0.128	0.050	0.398	0.112	0.282
a_{11}	15.3	37.6	35.9	5.7	23.4	39.6	15	12	2.95	0.357	1.09	1.13
a_{12}	0.130	0.216	0.154	0.001	0.890	0.179	0.458	0.414	0.353	0.262	0.042	0.364
a_{13}	0.077	0.179	0.116	0.294	0.010	0.234	0.559	0.400	0.496	0.372	0.190	0.728
a_{14}	0.032	0.404	0.205	0.127	0.010	0.528	0.198	0.140	0.376	0.080	0.963	0.257
a_{15}	293	389	72	2.63	12	238	13.6	194	11.9	5.19	7.71	18.3
a_{16}	0.771	0.107	0.368	0.980	0.124	0.617	0.140	0.331	0.780	0.435	0.058	0.321
a_{17}	0.261	0.038	0.879	0.013	0.027	0.159	0.191	0.797	0.546	0.231	0.002	0.211
a_{18}	18.4	20.0	10.2	16.5	1.3	29.4	21.7	9.01	24.9	22.9	1.32	17.3
a_{19}	416	570	646	338	360	689	629	691	697	547	688	698
a_{20}	0.646	0.713	0.435	0.840	0.003	0.644	0.557	0.479	0.271	0.112	0.035	0.858
a_{21}	0.703	0.635	0.531	0.384	0.574	0.803	0.653	0.853	0.895	0.132	0.005	0.796
a_{22}	0.834	0.830	0.558	0.000	0.014	0.065	0.001	0.030	0.232	0.223	0.017	0.111
a_{23}	0.492	0.050	0.567	0.009	0.011	0.091	0.905	0.822	0.055	0.004	0.034	0.010
a_{24}	0.380	0.437	0.521	0.022	0.004	0.064	0.011	0.139	0.106	0.004	0.010	0.052
a_{25}	13.1	15.9	25.6	2.81	19.5	39.8	29.8	13.2	28	2.6	13.9	14.6
a_{26}	1.050	-0.166	0.408	0.125	0.018	0.143	0.783	1.200	0.086	0.000	0.006	0.182
a_{27}	0.379	-0.086	0.083	0.189	0.169	0.051	0.215	0.432	0.096	0.001	0.046	0.228
a_{28}	0.647	0.287	0.045	0.002	0.009	0.051	0.508	0.413	0.100	0.062	0.001	0.069
a_{29}	0.827	0.062	0.083	0.000	0.001	0.012	0.692	0.240	0.129	0.058	0.018	0.032
a_{30}	0.546	0.472	0.267	0.002	0.956	0.763	0.415	0.643	0.995	0.002	0.505	0.891
a_{31}	0.340	0.920	0.038	0.217	0.013	0.422	2.520	0.187	0.034	0.005	0.164	0.036
a_{32}	3.408	20.784	0.811	0.338	0.127	6.509	7.490	5.079	0.898	0.190	0.032	0.283
a_{33}	16.461	2.028	2.161	2.378	1.181	0.625	8.563	3.267	0.115	0.031	0.006	0.004
a_{34}	21.282	2.619	8.563	21.615	2.278	2.610	1.280	6.360	3.109	0.202	1.089	1.139

(b) correspond to closeness of the estimated voltage curve with expected catalog voltage at various temperatures ($0°C, 25°C$ and $45°C$) for both charge and discharge processes respectively. Voltage-SOC characteristics have linear behavior in the middle SOC (0.2 to 0.8) values and hence require fewer sample points in that region. On the other hand, at higher and lower SOC values, they have exponential behavior that requires more sample points for accurate estimation of parameters. It can be observed from these curves that the voltage characteristic curves obtained from the model are following the catalog curves for all the measured temperatures. Due to the presence of non-linearity in the voltage characteristics for the low and high SOC values, most of the estimation errors occur in these region as represented by the small amount of deviations in the curves (Figure 2.3 (a) and (b)).

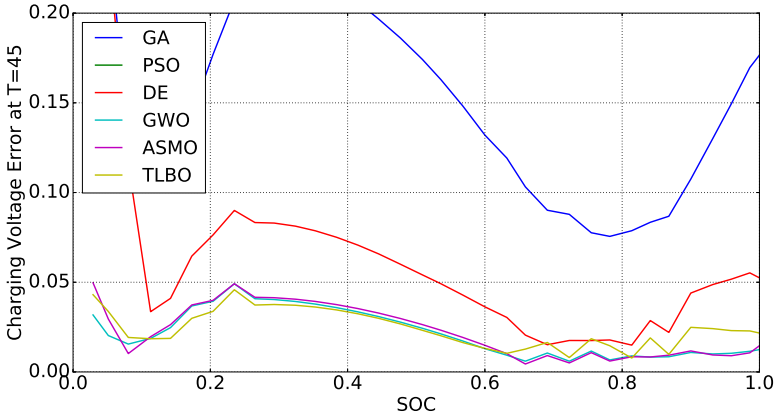
Error in the prediction of characteristic curves for battery terminal voltage at different



(a)

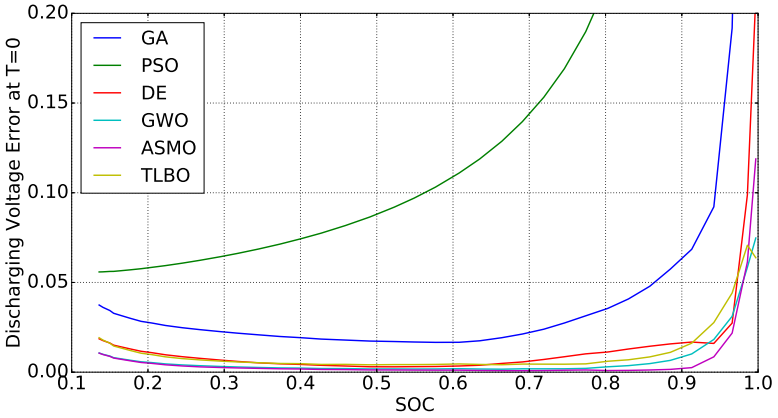


(b)

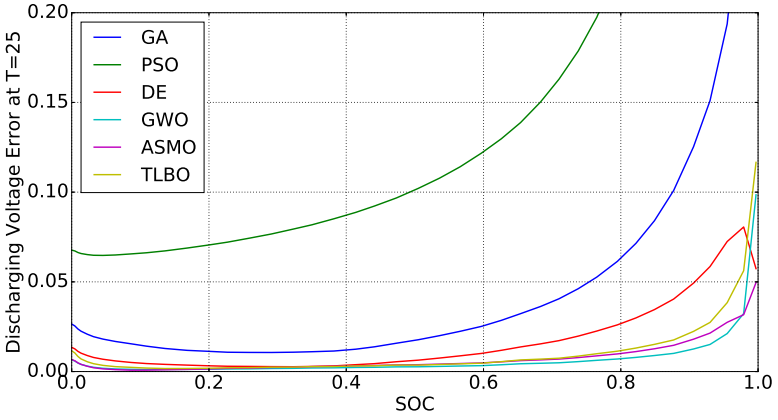


(c)

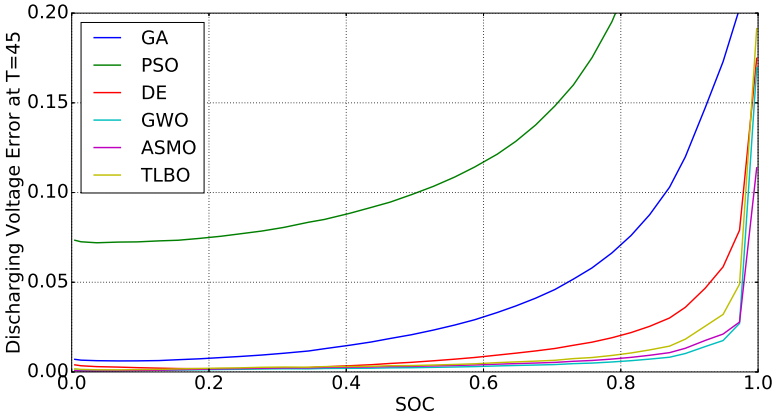
FIGURE 2.4: Predicted Characteristic Curve Error For charging at (a) $0°C$ (b) $25°C$ (c) $45°C$



(a)



(b)



(c)

FIGURE 2.5: Predicted Characteristic Curve Error For discharging at (a) $0°C$ (b) $25°C$ (c) $45°C$

temperature values ($0°C$, $25°C$, $45°C$) for both charge and discharge process utilizing different optimization techniques has been illustrated in Figure 2.4 and Figure 2.5 respectively. The predicted voltage error is computed corresponding to different SOC values within range 0 to 1 at different temperatures. The effect of non-linearity of voltage characteristics is apparent in these curves, as for all the optimization techniques, most of the error is present in the lower and higher SOC regions. The comparison between different optimization techniques has been made using different coloured lines. The high performance of ASMO and GWO algorithms are quite evident from these figures at all the measured temperatures for both charge and discharge processes. ASMO algorithm shows least error for most of the SOC values for all three temperature values compared to other optimization algorithms for both tasks.

The values of battery polynomial coefficients corresponding to best accuracy of the voltage characteristics obtained after optimization for both charging and discharging state have been tabulated in Table 2.1. The given battery parameter optimization problem belongs to a class of multimodal optimization problems. These problems, by nature, have a large number of local optima. Further, the high dimensionality (34 in this case) of the objective function, makes it almost impossible to find the global optima of the problem. In general, the solutions for these problems are approximate, and the given evolutionary optimization techniques are known to solve such problems by finding multiple approximate solutions of such problems and finally selecting the best approximation. Not all methods are capable of finding equally good approximations of the solution, thus resulting in different combinations of the parameter values for each solution. In general, each set of optimized parameter represents different local optima in the objective function solution surface.

2.5.2 Validation of estimated battery parameters

The estimated parameter values obtained with the proposed model are cross-validated by Constant-current Pulse Charge-Discharge Test (PCDT) in the present section. PCDT involves alternate cycles of charging/discharging and relaxing the battery for calculation of model parameters values. This test allows to observe the dynamic behavior of the battery at different temperatures. The procedure of PCDT is as follows [228]: firstly, the battery was fully charged using Constant-Current Constant-Voltage (CC-CV) charging technique. During CC-CV charging, the battery was charged at constant current with 1 C-rate until the terminal voltage of the battery reaches upper-threshold voltage 4.2 V. Thereafter, the voltage was held constant, and the current decays exponentially to 0.01 C as the battery begins to saturate [99]. Secondly, the battery was discharged by applying a positive pulse

current (C/2-rate) for 712 seconds followed by a relaxation period with zero current for 7126 seconds with 10% change of SOC. The foregoing process was repeatedly performed until the voltage reaches lower-threshold voltage 2.4 V. Finally, the battery was again charged by following the same routine with negative pulse current (C/2-rate) until the battery gets fully charged. The terminal voltage and current response of the battery for PCDT at room temperature ($25°C$) are shown in Figure 2.6 (a). In the same way, PCDT was carried out for temperature $0°C$ and $45°C$.

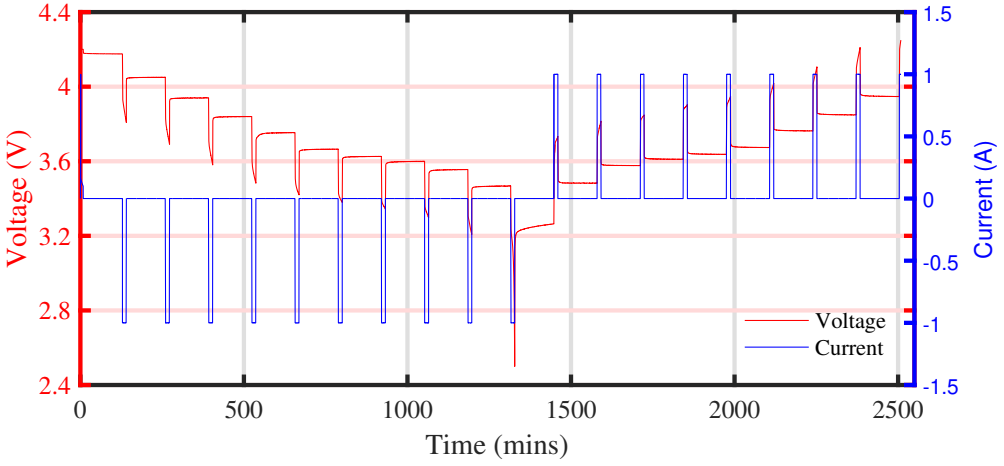


FIGURE 2.6: Voltage and current profile for Pulse Charge-Discharge Test at $25°C$

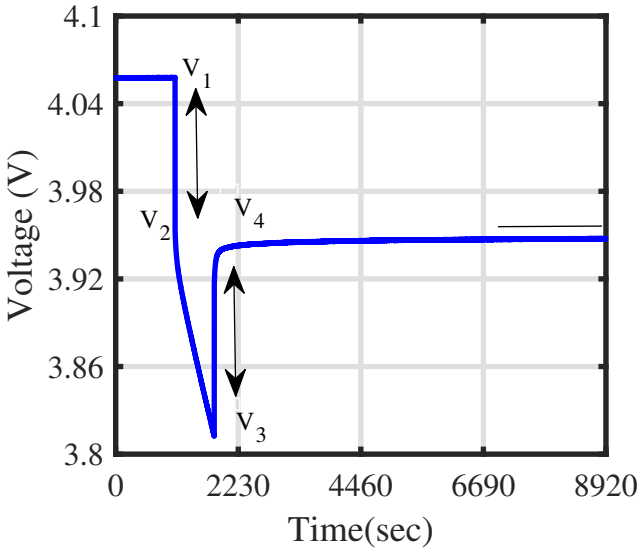


FIGURE 2.7: Voltage response to the pulse current

The terminal voltage of battery can be expressed as:

$$V_{Bt}^{M}(t) = V_{oc}(t) - I_{BL}(t)R_0 - V_{RC}(t) \tag{2.22}$$

$$V_{RC}(t) = \begin{cases} I_{BL}(t)R_1(1 - e^{(t-t_0)/R_1C_1}) & for \quad t_0 < t < t_d \\ \frac{Q(0)}{C_1}e^{-(t-t_d)/R_1C_1} & for \quad t_d < t < t_r \end{cases} \tag{2.23}$$

Here, t_0 and t_d denote time instant at the starting and end of the pulse respectively during charging and discharging scenario; whereas t_r denotes time instant at the end of relaxation; V_{RC} denotes voltage drop across R_1C_1.

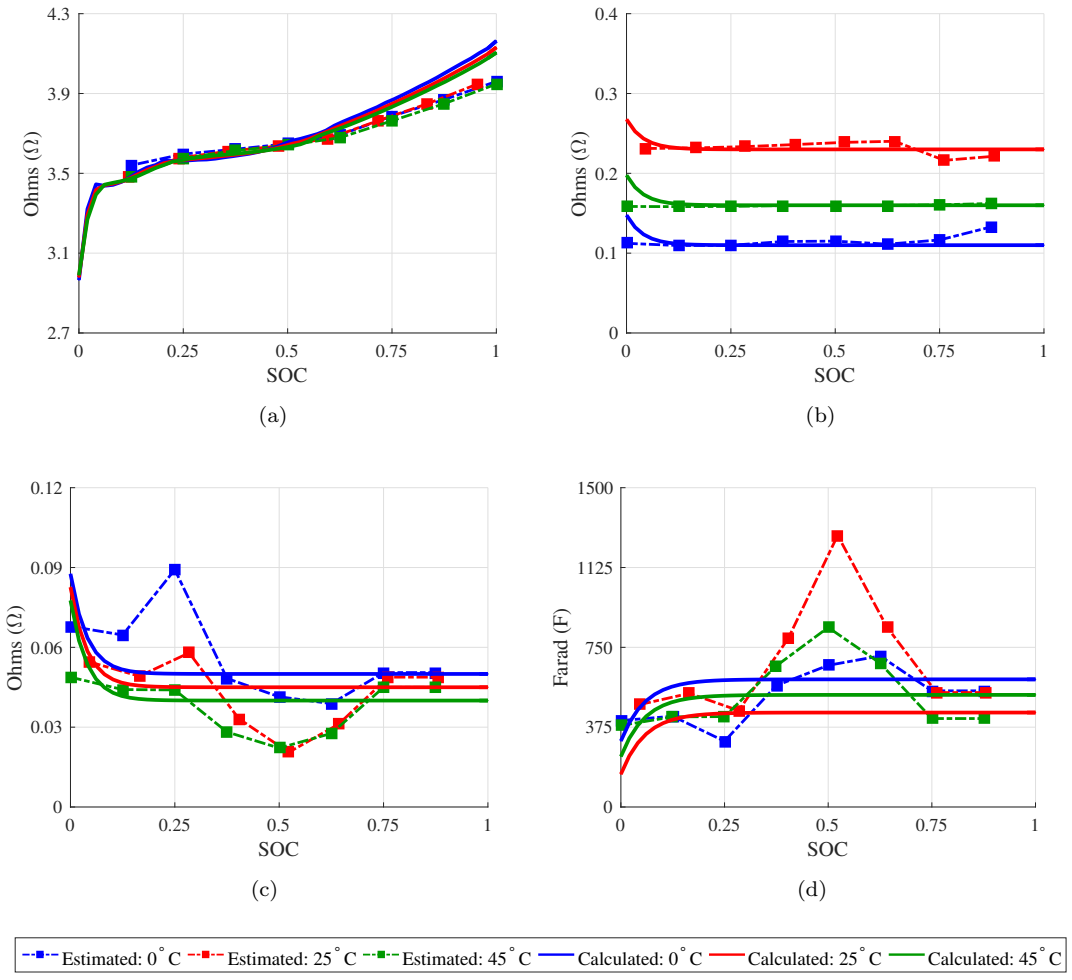


FIGURE 2.8: Plots for parameters estimated using proposed model and PCDT For Charging (a) V_{oc} (b) R_0 (c) R_1 (d) C_1

The values of parameter identified for battery model at different temperatures and SOC/-DOD during charging/discharging scenario are shown in Figure 2.8 and 2.9 respectively. V_{oc} is defined as the measured terminal voltage when the battery reaches steady-state.

An assumption is made that all transients have settled by the time instant t_r, and the terminal voltage of the battery at time instant t_r (i.e. V_1 in Figure 2.7) is considered as OCV (i.e. V_{oc}) value during charging/discharging scenario.

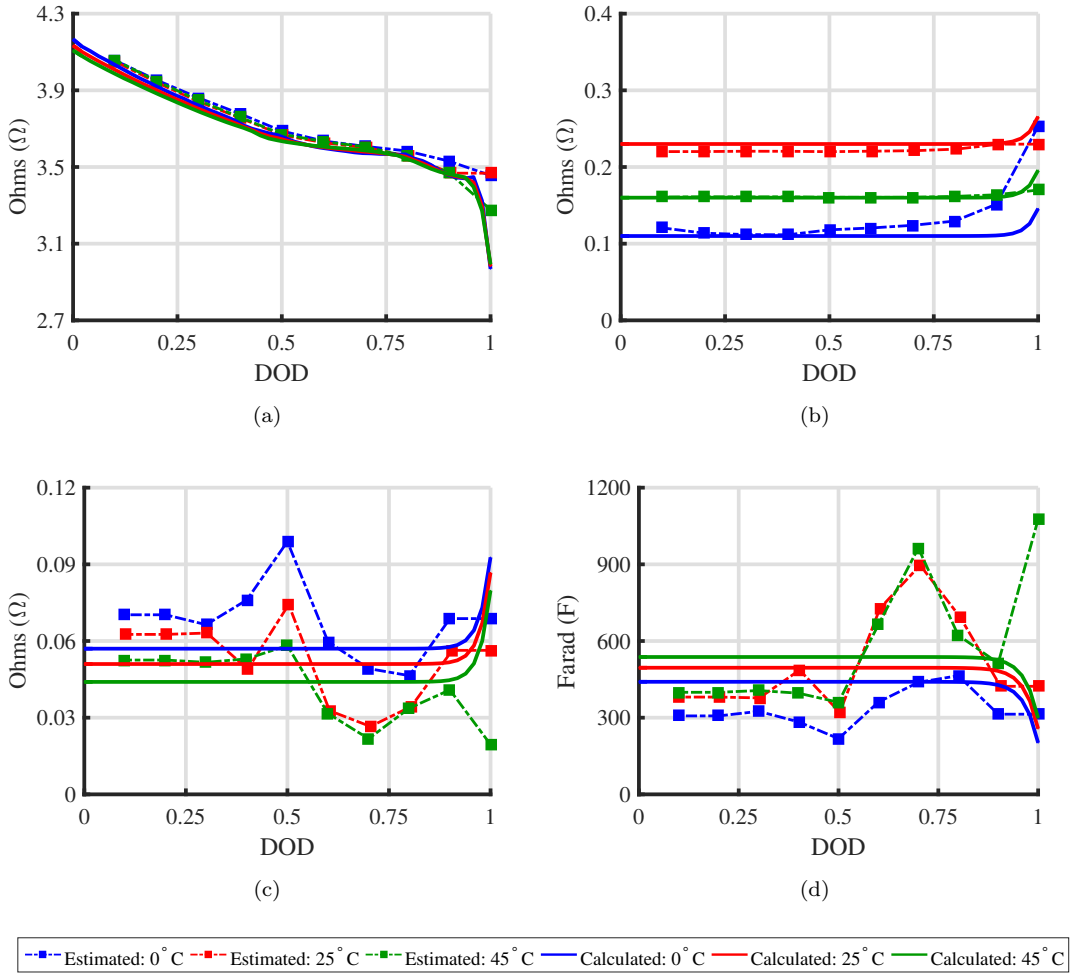


FIGURE 2.9: Plots for parameters estimated using proposed model and PCDT For Discharging (a) V_{oc} (b) R_0 (c) R_1 (d) C_1

Figure 2.8 (a) and 2.9 (a) show the relationships between V_{oc} and SOC/DOD at different temperatures and illustrate an agreement between values obtained from proposed method and PCDT. The consistency of V_{oc} at various temperatures indicates the low sensitivity of V_{oc} to temperature. V_{oc} change rapidly with low values of SOC and high values of DOD. Otherwise, there is approximately linear variation between V_{oc} and SOC/DOD.

The values of R_0, R_1 and C_1 were extracted from voltage drop obtained after subtracting terminal voltage from OCV. The voltage drop can be divided in two portions as shown in Figure 2.7 (b): firstly, the ohmic resistance voltage drop is the average of difference

$(V_1 - V_2)$ and $(V_3 - V_4)$. It is considered as the voltage drop across resistance R_0 of the battery. The value of R_0 can be calculated as the average of resistance obtained by dividing difference of voltages by difference of current at corresponding instant of time [74]. From Figure 2.8 (b) and 2.9 (b), it can be observed that R_0 is sensitive to temperature but there is a small difference in the curves for different SOC and DOD indicating that R_0 is approximately independent of SOC and DOD. Secondly, the transient voltage drop between $(V_2 - V_3)$ followed by voltage rise after V_4 depends on the R_1 of the battery. It is difficult to experimentally measure the time constant for polarization.

From equation (2.23), it can be observed that during charging and discharging process, V_{RC} is represented by the exponential function. Non-linear Least Square Fitting (NLSF) method is used to fit PCDT data with simulated terminal voltage obtained using equation (2.23). NLSF approach minimizes the error and optimize the value of the parameters.

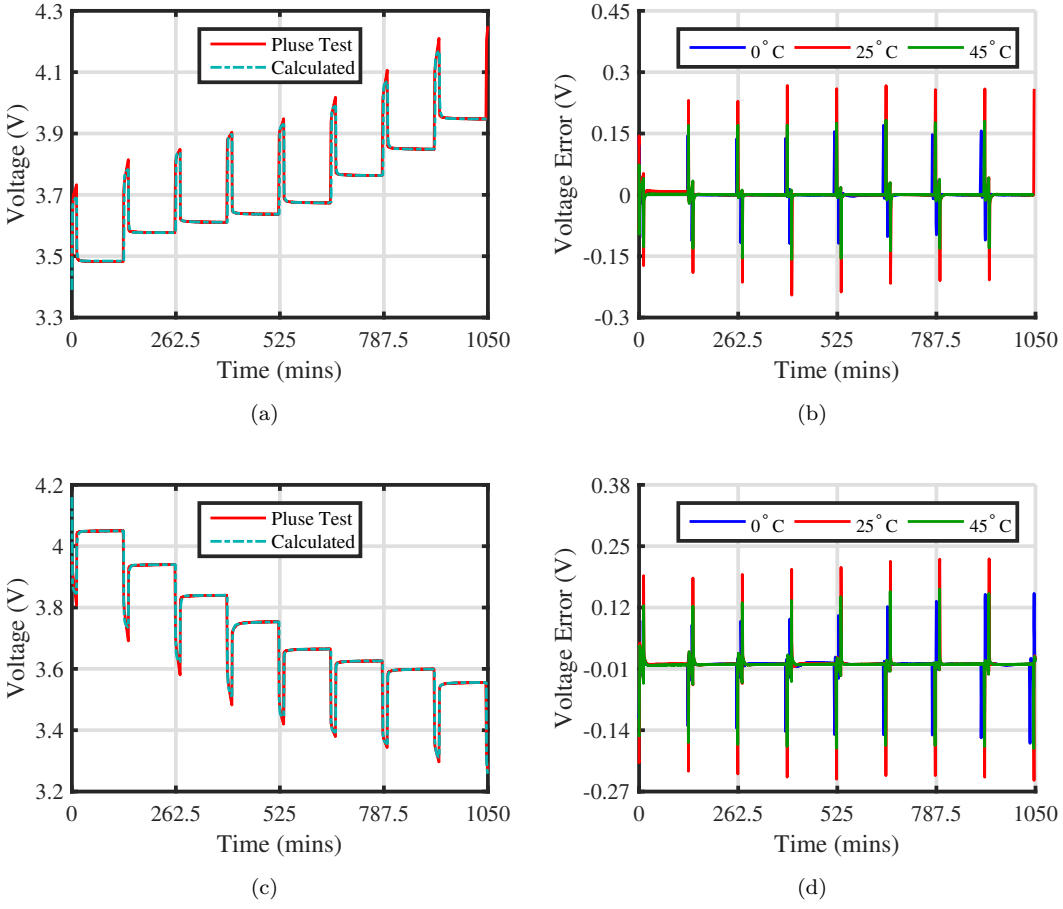


FIGURE 2.10: Plots for the PCDT voltages and calculated voltage from the extracted parameters For charging (a) Terminal voltage (b) Prediction error; For discharging (a) Terminal voltage (b) Prediction error

Through this approach, the value of R_1 and C_1 can be determined. The plots in Figure 2.8 (c) and 2.9 (c) indicate the value of R_1 and Figure 2.8 (d) and 2.9 (d) indicate the value of C_1 for charging and discharging scenario respectively estimated by NLSF approach and proposed method. The similarity between the values of parameters R_1 and C_1 obtained from the proposed method and PCDT indicates the effectiveness of the proposed work. As shown in Figures R_1 tends to decrease slowly with increasing SOC. Similarly, R_1 increases significantly with increase in DOD. High value of R_0 and R_1 at low temperature $0°C$ oppose the current flow in/form the battery which will limit the amount of extractable energy in battery.

To validate the estimated parameters of the battery model, the battery terminal voltage at $25°C$ obtained with the PCDT is compared with the terminal voltage obtained from equation (2.23) considering previously calculated value of the parameters. The plots of corresponding voltage at room temperature $25°C$ and predication error at various temperatures $(0°C, 25°C\ , 45°C)$ for both charging and discharging scenarios are shown in Figure 2.10. The maximum error in predicted voltage is less than ± 0.3 at various temperatures.

2.5.3 Performance analysis of various optimization algorithms

For evaluation purpose, a particular algorithm was employed ten times on specific models (charge and discharge) with an individual run consisting of 20,000 function evaluations based on convergence analysis. The sample size considered for GA was 1000, 40 for ASMO and PSO, DE, TLBO, and GWO each were taken with a population of sample size 40. The objective function for optimization given in equation (2.15) was adopted for comparison of quality and accuracy of estimation of parameters through different optimization techniques. The proposed model of identification of battery parameters is evaluated with different optimization algorithms under similar execution conditions. Results are obtained at three different temperatures, and the solution quality, computation efficiency, and convergence characteristics of all algorithms are compared.

For comparing the computational burden of the algorithms, the average convergence plots of different algorithms in case of charging and discharging scenarios have been given in Figure 2.11. In both scenarios, TLBO, GWO and ASMO algorithms converged more quickly than PSO, GA and DE, thus proving themselves to be less computationally expensive.

The performance of the executed optimization approaches is analyzed in terms of best and worst fitness function value, standard deviation and mean of the fitness function. Each algorithm is executed for ten identical trails and the calculation of statistical performance

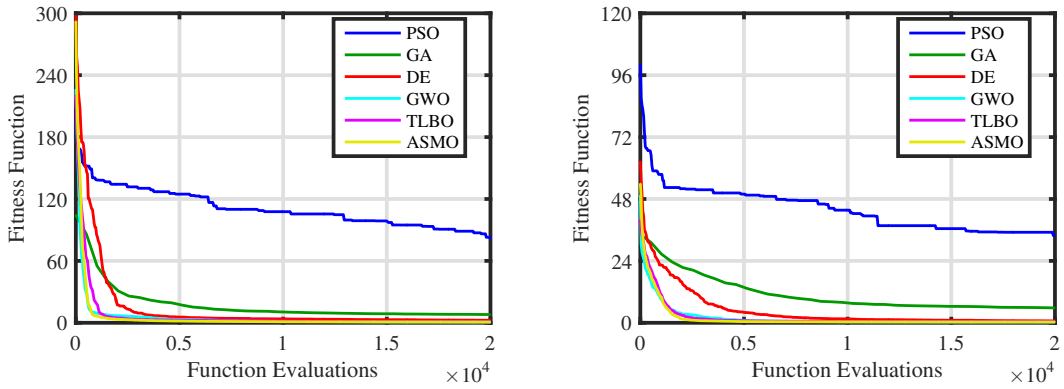


FIGURE 2.11: Convergence curves for (a) Charging (b) Discharging

indices in the case of charging and discharging is presented Table 2.2. Compared to GA, PSO, DE and TLBO algorithms, ASMO and GWO produced much lower mean error in both the charging and discharging scenarios. This shows the robustness of these algorithms in different scenarios (Charging and Discharging). Further, the much lower standard-deviation values show that algorithms performed consistently over all the runs justifying their reliability. ASMO marginally outperformed GWO algorithm in terms of mean error. This is due to fusion-fission based group formation in the ASMO algorithm which helps it to avoid the stagnation condition. The best fitness value obtained by ASMO is better than best fitness value obtained by other methods for both charging (0.572) and discharging (0.272) scenarios indicating the superiority of ASMO algorithm over other approaches. Further, in both scenarios, the best result was given by the ASMO algorithm which was used for plotting the predicted voltage characteristic curves (Figure 2.3 (a) and (b)).

TABLE 2.2: Error comparison for charge and discharge Model

	GA	**PSO**	**DE**	**TLBO**	**GWO**	**ASMO**
Charging						
Mean	7.9904	81.5254	2.4461	1.0541	0.7750	**0.6991**
St.D	2.9043	26.0813	0.1674	0.2775	**0.0594**	0.0950
Worst	12.1405	121.8897	2.7140	1.5447	**0.8868**	0.8932
Best	2.9802	39.9198	2.2520	0.7680	0.6862	**0.5724**
Discharging						
Mean	5.7681	23.3100	0.7406	0.4778	0.3269	**0.2983**
St.D	5.0248	1.8921	0.0848	0.0913	0.0287	**0.0191**
Worst	19.543	25.9502	0.8635	0.6658	0.3729	**0.3373**
Best	2.4044	21.2600	0.6060	0.3755	0.2869	**0.2718**

Although classical performance metrics such as mean, standard deviation, best and worst fitness value, convergence rate are suitable methods of comparing the algorithms, but they are not sufficient to find the difference in performance of the algorithms. This can be seen clearly in previous results that there is not much distinction between GWO and ASMO algorithms. To compare the performance and statistical significance of computationally intelligent algorithms, the popularity of parametric and nonparametric tests has increased

TABLE 2.3: T-test for charging scenario

		GA	**PSO**	**DE**	**TLBO**	**GWO**	**ASMO**
GA	t	0	-8.836	6.033	7.525	7.863	7.943
	h	0	1	1	1	1	1
	p	1	5.79×10^{-8}	1.05×10^{-5}	5.79×10^{-7}	3.13×10^{-7}	2.71×10^{-7}
PSO	t	8.836	0	9.558	9.726	9.760	9.769
	h	1	0	1	1	1	1
	p	5.79×10^{-8}	1	1.78×10^{-8}	1.37×10^{-8}	1.3×10^{-8}	1.28×10^{-8}
DE	t	-6.033	-9.558	0	13.59	29.80	28.75
	h	1	1	0	1	1	1
	p	1.05×10^{-5}	1.78×10^{-8}	1	6.64×10^{-11}	9.02×10^{-17}	1.7×10^{-16}
TLBO	t	-7.525	-9.726	-13.59	0	3.118	3.837
	h	1	1	1	0	1	1
	p	5.79×10^{-7}	1.37×10^{-8}	6.64×10^{-11}	1	5.94×10^{-3}	1.21×10^{-3}
GWO	t	-7.863	-9.760	-29.80	-3.118	0	2.145
	h	1	1	1	1	0	1
	p	3.13×10^{-7}	1.3×10^{-8}	9.02×10^{-17}	5.94×10^{-3}	1	4.58×10^{-2}
ASMO	t	-7.943	-9.769	-28.75	-3.837	-2.145	0
	h	1	1	1	1	1	0
	p	2.71×10^{-7}	1.28×10^{-8}	1.7×10^{-16}	1.21×10^{-3}	4.58×10^{-2}	1

TABLE 2.4: T-test for discharging scenario

		GA	**PSO**	**DE**	**TLBO**	**GWO**	**ASMO**
GA	t	0	-10.35	3.169	3.334	3.430	3.448
	h	0	1	1	1	1	1
	p	1	5.21×10^{-9}	5.32×10^{-3}	3.69×10^{-3}	2.99×10^{-3}	2.87×10^{-3}
PSO	t	10.35	0	37.56	37.99	38.28	38.33
	h	1	0	1	1	1	1
	p	5.21×10^{-9}	1	1.49×10^{-18}	1.22×10^{-18}	1.06×10^{-18}	1.04×10^{-18}
DE	t	-3.169	-37.56	0	6.660	14.59	16.07
	h	1	1	0	1	1	1
	p	5.32×10^{-3}	1.49×10^{-18}	1	3×10^{-6}	2.05×10^{-11}	4.06×10^{-12}
TLBO	t	-3.334	-37.99	-6.660	0	4.987	6.087
	h	1	1	1	0	1	1
	p	3.69×10^{-3}	1.22×10^{-18}	3×10^{-6}	1	9.56×10^{-5}	9.43×10^{-6}
GWO	t	-3.430	-38.28	-14.59	-4.987	0	2.622
	h	1	1	1	1	0	1
	p	2.99×10^{-3}	1.06×10^{-18}	2.05×10^{-11}	9.56×10^{-5}	1	1.73×10^{-2}
ASMO	t	3.448	-38.33	-16.07	-6.087	-2.622	0
	h	1	1	1	1	1	0
	p	2.87×10^{-3}	1.04×10^{-18}	4.06×10^{-12}	9.43×10^{-6}	1.73×10^{-2}	1

in last few years. T-test (Parametric) and Wilcoxon test (Non-Parametric) can be carried out for comparing different algorithms. The key performance metrics include p-value, h-value along with the corresponding t-value during statistical analysis that are calculated by executing at least ten identical trails for all the algorithms. Table 2.3 and 2.4 show the p-value, h-value along with the corresponding t-value of all algorithms in comparison to each other for charging and discharging scenarios. The algorithms in the first column are used as base algorithms in reference to which the algorithms in the first row have been compared. p-Value represents the probability of rejection of the null hypothesis. Its value lies in between 0 and 1. Lesser the p-value, more is the difference between the compared algorithms. Hypothesis test or h-value also indicates the rejection of the null hypothesis. h-value of 1 represents confirmation of the rejection of null hypothesis and thus indicates that the compared algorithms are different. For hypothesis testing, a significance level of 5% was taken. The t-test assesses whether the mean of two groups of results is statistically different from each other or not. For the purpose of testing, two-tailed t-test was adopted

Table 2.5: Wilcoxon-test for charging scenario

		GA	**PSO**	**DE**	**TLBO**	**GWO**	**ASMO**
	h	0	1	1	1	1	1
GA	p	1	1.83×10^{-4}	1.83×10^{-4}	1.83×10^{-4}	1.83×10^{-4}	1.83×10^{-4}
	h	1	0	1	1	1	1
PSO	p	1.83×10^{-4}	1	1.83×10^{-4}	1.83×10^{-4}	1.83×10^{-4}	1.83×10^{-4}
	h	1	1	0	1	1	1
DE	p	1.83×10^{-4}	1.83×10^{-4}	1	1.83×10^{-4}	1.83×10^{-4}	1.83×10^{-4}
	h	1	1	1	0	1	1
TLBO	p	1.83×10^{-4}	1.83×10^{-4}	1.83×10^{-4}	1	3.61×10^{-4}	8.77×10^{-4}
	h	1	1	1	1	0	1
GWO	p	1.83×10^{-4}	1.83×10^{-4}	1.83×10^{-4}	3.61×10^{-4}	1	4.12×10^{-4}
	h	1	1	1	1	1	0
ASMO	p	1.83×10^{-4}	1.83×10^{-4}	1.83×10^{-4}	8.77×10^{-4}	4.12×10^{-4}	1

Table 2.6: Wilcoxon-test for discharging scenario

		GA	**PSO**	**DE**	**TLBO**	**GWO**	**ASMO**
	h	0	1	1	1	1	1
GA	p	1	1.81×10^{-4}	1.83×10^{-4}	1.83×10^{-4}	1.82×10^{-4}	1.82×10^{-4}
	h	1	0	1	1	1	1
PSO	p	1.81×10^{-4}	1	1.81×10^{-4}	1.81×10^{-4}	1.80×10^{-4}	1.80×10^{-4}
	h	1	1	0	1	1	1
DE	p	1.83×10^{-4}	1.81×10^{-4}	1	3.30×10^{-4}	1.82×10^{-4}	1.82×10^{-4}
	h	1	1	1	0	1	1
TLBO	p	1.83×10^{-4}	1.81×10^{-4}	3.3×10^{-4}	1	1.82×10^{-4}	1.82×10^{-4}
	h	1	1	1	1	0	1
GWO	p	1.82×10^{-4}	1.8×10^{-4}	1.82×10^{-4}	1.82×10^{-4}	1	2.1×10^{-2}
	h	1	1	1	1	1	0
ASMO	p	1.82×10^{-4}	1.80×10^{-4}	1.82×10^{-4}	1.82×10^{-4}	2.1×10^{-2}	1

with 5% significance level. The negative t-value with ASMO as base algorithm along with low p-value and h-value of 1 with respect to all the other algorithms proves ASMO to be significantly better than other algorithms including GWO. A further comparison has been made in Table 2.5 and 2.6 using the Wilcoxon ranked sum test. For this test, the comparison data was taken in normalized form with a significance level of 5%. This test finds if there is a significant difference between the two compared algorithms. As evident from the h-value of 1, there is a significant difference in the performance of the ASMO algorithm and other algorithms including GWO.

T-test and Wilcoxon test clearly show that the ASMO algorithm performed significantly better than all other tested algorithms. The lower mean fitness value and standard deviation of the ASMO prove its superioty in the battery parameter estimation scenario. Further, the low computational requirements signify the robustness of the algorithm, thus making it a prime choice for the task of battery parameter estimation.

2.6 Chapter Summary

An accurate determination of the parameters of battery model plays an indispensable part in replicating the behavior of battery. In this chapter,a first-order RC battery model incorporating the effect of the temperature on the battery parameters with SOC and C-rate is utilized. The parameter estimation is modeled in optimization framework as minimization of the Manhattan distance between the voltage computed from the battery model and catalog voltages supplied by the manufacturer. Six different meta-heuristic optimization techniques (*viz.* GA, PSO, DE, TLBO, GWO, and ASMO) have been utilized to obtain optimal parameters of the model. The performance of applied optimization techniques is compared in terms of their ability and accuracy in the extraction of the battery model parameters with a lower convergence rate. For the model, GWO and ASMO optimization techniques proved to be more robust, reliable and gave an optimal solution in both charging and discharging scenarios. The lower mean fitness value (charging:0.699/discharging:0.298) and standard deviation (less than 1) of the Ageist Spider Monkey Optimization (ASMO) proved its superiority in battery parameter estimation. For precise statistical investigation of computational intelligence algorithms, parametric (T-test) and non-parametric tests (Wilcoxon test) have been performed. From performance index and test-based study, it was concluded that the ASMO algorithm functions more reliably than all other tested algorithms. Further, the low computational requirements signify the simplicity of the algorithm, thus presenting it as an appropriate option for the evaluation of battery model

parameters. The values of parameters extracted using ASMO are validated with values evaluated from PCDT. It is observed from the validation process that there is a similarity between parameters values obtained from both the methods. Therefore, the proposed technique can be used as an alternative for estimation of battery parameters.