Optimal power allocation scheme for plug-in hybrid electric vehicles using swarm intelligence techniques

Pandian M. Vasant1*, Imran Rahman1, Balbir Singh Mahinder Singh1 and M. Abdullah-Al-Wadud2

Abstract: Green technologies gain popularity to reduce the pollution and give higher penetration of renewable energy source in the transportation. This research induce that the extensive involvement of plug-in hybrid electric vehicles (PHEVs) requires adequate charging allocation strategy using a combination of smart grid systems and smart charging infrastructures. It is also noticed that daytime charging station are necessary for daily usage of PHEVs due to the limited all-electric-range. Most of the researches in the past have been stated that only proper charging control and infrastructure management can assure the larger participation of PHEVs. Therefore, researchers are trying to develop efficient control mechanism for charging infrastructure in order to facilitate upcoming PHEVs penetration in highway. Nevertheless, most of the past researcher already aware with the issue related to intelligent energy management. Yet, these studies could not fill the gap of the problem associated with intelligent energy management and require formulation of mathematical models with extensive use of computational intelligence-based optimization techniques to solve many technical problems. The outcome of this research study provides four optimization techniques that include Hybrid method.
within swarm intelligence group for the State-of-Charge (SoC) optimization of PHEVs. The finding of this research simulation results obtained for maximizing the highly nonlinear objective function evaluate the comparative performance of all four techniques in terms of best fitness, convergence speed, and computation time. Finally, the hybridization method (PSOGSA) presented in this dissertation uses the advantages of both PSO and GSA optimization and thus produce higher best fitness values. This study evaluates the performance of standard PSO, then Accelerated version of PSO (APSO), GSA algorithm and then Hybrid of PSO and GSA. The hybridization method (PSOGSA) uses the advantages of both PSO and GSA optimization and thus produce higher best fitness values. However, PSOGSA method takes much longer computational time than single methods because of incorporating two single methods in one algorithm. This research study suggests that PSOGSA method is a great promise for SoC optimization but it takes much longer computational time.

Subjects: Artificial Intelligence; Engineering Management; Power & Energy; Transport & Vehicle Engineering

Keywords: APSO; electric vehicle; gravitational search algorithm; optimization; PSO; PSOGSA; swarm intelligence

1. Introduction
Recent researches on sustainable technologies for transportation sector are gaining popularity among the research communities from different areas. In this wake, Plug-in hybrid electric vehicles (PHEVs) have great future because of their charge storage system and charging facilities from traditional grid system. Several researchers have proved that a great amount of reductions in greenhouse gas emissions and the increasing dependence on oil could be accomplished by electrification of transport sector (Rahman et al., 2016a). Future transportation sector will depend much on the advancement of this emerging field of vehicle optimization (Tie & Tan, 2013). PHEVs which is very recently introduced promise to boost up the overall fuel efficiency by holding a higher capacity battery system, which can be directly charged from conventional power grid system, that helps the vehicles to operate continuously in “all-electric-range” (AER). All-electric vehicles or AEVs is a kind of transport that uses electric power as only sources to run the system. PHEVs with a connection to the smart grid can own all of these strategies. Hence, the widely extended adoption of PHEVs might play a significant role in the alternative energy integration into traditional grid systems (Lund & Kempton, 2008). There is a need of efficient mechanisms and algorithms for smart grid technologies in order to solve highly diverse problems like energy management, cost reduction, efficient charging station, etc. with different objectives and system constraints (Hota, Juvvanapudi, & Bajpai, 2014).

According to a statistics of Electric Power Research Institute (EPRI), about 62% of the entire United States (US) vehicle will comprise PHEVs within the year 2050 (Soares et al., 2013). Moreover, there is an increasing demand to implement this technology on the electric grid system. Large numbers of PHEVs have the capability to make threats to the stability of the power system. For example, in order to avoid disturbance when several thousand PHEVs are introduced into the system over a small period, the load on the power grid will need to be managed very carefully. One of the main targets is to facilitate the proper communication between the power grid and the PHEV. For the maximization of customer contentment and minimization of burdens on the grid, a complicated control appliance will need to be addressed in order to govern multiple battery loads from a numbers of PHEVs properly (Su & Chow, 2012). The total demand pattern will also have an important impact on the electricity production due to differences in the needs of the PHEVs parked in the deck at certain time (Su & Chow, 2011a). Proper management can ensure strain minimization of the grid and enhance the transmission and generation of electric power supply. The control of PHEV charging depending on the locations can be classified into two groups; household charging and public charging. The proposed
optimization focuses on the public charging station for plug-in vehicles because most of PHEV charging is expected to take place in public charging location (Su, Eichi, Zeng, & Chow, 2012). Wide penetration of PHEVs in the market depends on a well-organized charging infrastructure. The power demand from this new load will put extra stress on the traditional power grid (Morrow, Karner, & Francfort, 2008). In (Amini & Parsa Moghaddam, 2013), a probabilistic approach is utilized to estimate the expected EV charging demand based on the historical data. To this end, Amini et al. used the expected arrival and departure time, and daily driven distance to obtain a realistic model for EV parking lots. Charging stations are needed to be built at workplaces, markets/shopping malls, and home. Boyle (2007) proposed the necessity of building new smart charging station with effective communication among utilities along with sub-station control infrastructure in view of grid stability and proper energy utilization. Furthermore, sizeable energy storage, cost minimization, Quality of Services (QoS), and intelligent charging station for optimal power are underway (Hess et al., 2012). In this wake, numerous techniques and methods were proposed for deployment of PHEV charging stations (Li et al., 2010).

One of the main targets is to facilitate the proper interaction between the power grid and the PHEV. For the maximization of customer satisfaction and minimization of burdens on the grid, a complicated control mechanism will need to be addressed in order to govern multiple battery loads from a numbers of PHEVs appropriately (Su et al., 2012). Charging infrastructures are essential in order to facilitate the large-scale penetration of PHEVs. Some researchers for charging station optimization of PHEV have used different computational intelligence-based methods. Most of them applied traditional methods that needed to be improved furthermore.

Swarm intelligence came from the mimic of the living colony such as ant, bird, and fish in nature, which shows unparalleled excellence in swarm than in single in food seeking or nest building. Drawing inspiration from this, researchers design many algorithms simulating colony living, such as Ant Colony Optimization (ACO) algorithm (Dorigo, 2006), Particle Swarm Optimization (PSO) algorithm (Eberhart & Yuhui, 2001), Artificial Bee Colony (ABC) algorithm (Karaboga & Basturk, 2007), and Gravitational Search Algorithm (GSA) (Rashedi, Nezamabadi-pour, & Saryazdi, 2009), which shows excellent performance in dealing with complex optimization problems (Martens, Baesens, & Fawcett, 2011). The intrinsic characteristics of all the population-based metaheuristic algorithms like PSO and GSA are to maintain a good compromise between exploration and exploitation in order to solve the complex optimization problems (Rashedi et al., 2009).

The performance of PHEV depends upon proper utilization of electric power which is solely affected by the battery state-of-charge (SoC). In PHEVs, a key parameter is the SoC of the battery as it is a measure of the amount of electrical energy stored in it. It is analogous to fuel gauge on a conventional internal combustion (IC) car (Chiasson & Vairamohan, 2005). State-of-charge determination becomes an increasingly vital issue in all the areas that include a battery. Previous operation policies made use of voltage limits only to guard the battery against deep discharge and overcharge. Currently, battery operation is changing to what could rather be called battery management than simply protection. For this improved battery control, the battery SoC is a key factor indeed (Piller, Perrin, & Jossen, 2001).

A charging station is one way that the operator of an electrical power grid can adapt energy production to energy consumption, both of which can vary randomly over time. Generally, PHEVs in a charging station are charged during times when production exceeds consumption and are discharged at times when consumption exceeds production (Yang, He, & Fu, 2014). There is a need of in-depth study on maximization of average SoC in order to facilitate intelligent energy allocation for PHEVs in a charging station.

The purpose of this research is to optimize state-of-charge, with respect to charging time, present SoC. For this, swarm intelligence-based methods, PSO and GSA, Accelerated Particle Swarm...
Optimization (APSO), and Hybrid version of PSO and GSA (PSOGSA) were applied for solving the particular optimization problem hence presents comparative study on these four techniques.

2. Problem statement

The anticipation of a large penetration of PHEVs and Plug-in Electric Vehicles (PEVs) into the market brings up many technical problems that need to be addressed within the next 10 years (Hannan, Azdin, & Mohamed, 2014). In the future, electric-powered vehicles would be plugged into the grid, and their onboard energy storage systems would be recharged using clean, renewable electricity. One of the key missions is to facilitate the smooth interaction between the plug-in hybrid electric vehicle and the power system (Mwasilu, Justo, Kim, Do, & Jung, 2014). Therefore, there is a need of optimum power allocation scheme in order to facilitate the SoC of PHEVs as well as the overall performance of charging station for large-scale PHEVs in smart grid environment. Lack of proper and adequate charging infrastructure is another issue regarding PHEV compared to typical fuel station. As a result, uncertainty in energy management for the driver named as ‘Range anxiety’ (Contestabile, Offer, Slade, Jaeger, & Thoennes, 2011) develops which restricts the conventional vehicle owner to switch their vehicle to PHEV. Selecting appropriate place for charging station can decrease this range anxiety. There remain much scientific studies to be performed on how to manage PHEV storage energy, SoC effectively for real-time traffic situations. One of the important constraints for accurate charging is SoC. Charging algorithm can precisely be managed by the precise state of charge evaluation (Shafiei & Williamson, 2010). An approximate graph of a typical Lithium-ion cell voltage vs. SoC shown in Figure 1 indicates that the slope of the curve below 20% and above 90% is high enough to result in a significant voltage difference to be depended on by measurement circuits and charge balancing control. There is a need of in-depth study on maximization of average SoC in order to facilitate intelligent energy allocation for PHEVs in a charging station.

The idea behind smart charging is to charge the vehicle when it is most favorable, which could be when electricity price, demand is lowest, when there is excess capacity (Chang, 2013). When a vehicle is plugged in into a smart charging station a request for energy demand is sent to Substation Control Center (SCC), which decides based on the available energy from utility and either accepts the request or rejects it. Performance of this kind of load management is measured in terms of delay, delivery ration, and jitter. As a matter of fact EVs may be charged at any time of a day depending on requirement to top their batteries even during peak demand hours. In (Amini & Arif, 2014), an optimal reliability-constraint allocation of EVs’ parking lots is proposed based on the probabilistic charging demand model of EVs.

3. Scope of study

This research mainly focuses on swarm intelligence-based optimization methods for solving PHEVs charging problem. Here, a single parking lot with the aggregation of distribution network-connected PHEVs is considered. We make use of historical data for office parking from the city of Livermore, CA (Rahman et al., 2016b). There are many swarm intelligence-based method applied for solving real-world problems. Among them four methods have been applied to maximize the state-of-charge. Authors in (Su & Chow, 2012) first applied Genetic Algorithm (GA) and PSO methods to solve this particular problem. According to the results, the authors claim that PSO algorithm converges to a solution in a reasonable amount of time, faster than GA. They also suggest applying other optimization techniques for comparison purposes and selecting the best-suited technique to maximize the SoC of PHEV. From this future research direction, we have come up with swarm intelligence-based methods with comparative study in order to find out the suitable techniques among swarm intelligence-based optimization domain. According to Rahman et al. (2016c), many papers do not document experimental settings in sufficient detail, and hence replication of experiments is almost impossible. We have found the same problem in the research of PSO-based optimization for charging PHEV. Moreover, they used swarm size (population size) 40 which is not sufficient to describe the performance of PSO algorithm properly. Therefore, we have replicated the PSO optimization in our own computer configuration and parameter settings for the fair comparison with other three techniques.
4. Methodology
Suppose there is a charging station with the capacity of total power $P$, total $N$ numbers of PHEVs need to be served in a day (24 h). The proposed system should allow PHEVs to leave the charging station before their expected leaving time for making the system more effective. It is worth to mention that each PHEV is regarded to be plugged-in to the charging station once. The main aim is to allocate power intelligently for each PHEV coming to the charging station. The State-of-Charge is the main parameter which needs to be maximized in order to allocate power efficiently. For this, the fitness function considered in this research is the maximization of average SoC and thus allocate energy for PHEVs at the next time step. The constraints considered are: charging time, present SoC, and price of the energy.

The fitness function is defined as:

$$\text{Max } J(k) = \sum_i w_i(k) \cdot \text{SoC}_i(k + 1)$$  \hspace{2cm} (1)

$$w_i(k) = f(C_{r,i}(k), T_{r,i}(k), D_i(k))$$  \hspace{2cm} (2)

$$C_{r,i}(k) = (1 - \text{SoC}_i(k)) \cdot C_i$$  \hspace{2cm} (3)

where $C_{r,i}(k)$ is the battery capacity (remaining) needed to be filled for $i$ No. of PHEV at time step $k$; $C_i$ is the battery capacity (rated) of the $i$ no. of PHEV; remaining time for charging a particular PHEV at time step $k$ is expressed as $T_{r,i}(k)$; the price difference between the real-time energy price and the price that a specific customer at the $i$ no. of PHEV charger is willing to pay at time step $k$ is presented by $D_i(k)$; $w_i(k)$ is the charging weighting term of the $i$ no. of PHEV at time step $k$ (a function of charging time, present SoC, and price of the energy); $\text{SoC}_i(k + 1)$ is the state of charge of the $i$ no. of PHEV at time step $k + 1$.

The weighting term gives a reward proportional to the attributes of a specific vehicle. For example, if a vehicle has a lower initial SoC and less remaining charging time, but the driver is willing to pay a higher price, the controller allocates more power to this vehicle battery charger:

$$w_i(k) \cdot [C_{r,i}(k) + D_i(k) + 1/T_{r,i}(k)]$$  \hspace{2cm} (4)

The charging current is also assumed to be constant over $\Delta t$.

$$[\text{SoC}_i(k + 1) - \text{SoC}_i(k)] \cdot \text{Cap}_i = Q_i = I_i(k) \Delta t$$  \hspace{2cm} (5)
SoC\(_i\)(k + 1) = SoC\(_i\)(k) + \(I_i(k)\Delta t/Cap_i\),  \(\text{(6)}\)

where the sample time \(\Delta t\) is defined by the charging station operators, and \(I_i(k)\) is the charging current over \(\Delta t\). The battery model is regarded as a capacitor circuit, where \(C_i\) is the capacitance of battery (Farad). The model is defined as:

\[C_i \frac{dV_i}{dt} = I_i\]  \(\text{(7)}\)

Therefore, over a small time interval, one can assume the change of voltage to be linear,

\[C_i[V_i(k + 1) - V_i(k)]/\Delta t = I_i\]  \(\text{(8)}\)

\[V_i(k + 1) - V_i(k) = I_i\Delta t/C_i\]  \(\text{(9)}\)

Since the decision variable is the power allocated to the vehicles, replacing \(I_i(k)\) with \(P_i(k)\):

\[I_i(k) = P_i(k)/0.5 \times [V_i(k + 1) - V_i(k)]\]  \(\text{(10)}\)

\[V_i(k + 1) = \sqrt{2P_i(k)\Delta t/C_i} + V_i^2(k)\]  \(\text{(11)}\)

Substituting (10) into (6) yields:

\[\text{SoC}_i(k + 1) = \text{SoC}_i(k) + \frac{P_i(k)\Delta t}{0.5C_i} \left[\sqrt{\frac{2P_i(k)\Delta t}{C_i} + V_i^2(k) + V_i(k)}\right]\]  \(\text{(12)}\)

Finally, the objective function becomes:

\[J(k) = \sum w_j \left| \frac{P_i(k)\Delta t}{0.5C_i} \left[\sqrt{\frac{2P_i(k)\Delta t}{C_i} + V_i^2(k) + V_i(k)}\right] \right|\]  \(\text{(13)}\)

There are two kinds of inequality constraints used here to optimize the fitness function—(i) Power from the charging station operator and (ii) individual PHEV’s SoC. Power obtained from the utility (\(P_{\text{utility}}\)) and the maximum power (\(P_{\text{max}}\)) absorbed by a specific PHEV are the primary energy constraints being considered in this chapter. The overall charging efficiency of a particular charging station is described by \(\eta\). From the system point of view, charging efficiency is supposed to be constant at any given time step. Maximum battery SoC limit for the \(i\) No. of PHEV is \(\text{SoC}_{i_{\text{max}}}\). When \(\text{SoC}_i\) reaches the values close to \(\text{SoC}_{i_{\text{max}}}\), the \(i\) No. of battery charger shifts to a standby mode. The state of charge ramp rate is confined within limits by the constraint \(\Delta\text{SoC}_{\text{max}}\). The overall control system changes the state when (i) system utility data updates; (ii) a new PHEV is plugged-in; (iii) time period \(\Delta t\) has periodically passed. Obviously, SoC maximization method being considered in this chapter can provide a uniformly higher SoC for all PHEVs/PEVs at plug-out as compared with the alternative schemes. It also proves that the proposed function aims at ensuring some fairness in the SoC distribution at each time step. This will help to ensure that a reasonable level of battery power is attained, even in the event of an early departure. Table 1 shows all the fitness function parameters that were tuned for performing the optimization. There are total three (03) kinds of parameters: fixed, variables, and constraints. Total charging time is fixed to 20 min and charging station efficiency is assumed to be 0.9. The values are retrieved from various literatures (Hota et al., 2014; Kulshrestha, Wang, Chow, & Lukic, 2009). Moreover, SoC is in the range of 0.2–0.8 (Su & Chow, 2011b).
The objective function and parameter settings have already been discussed earlier. After that, we set the parameters of each swarm intelligence-based optimization technique. If the experiment is not replicated with sufficient care, any performance measures and statistical approaches cannot remedy the problems introduced by inexact experiment replication. In other words, if collected data are gathered from experiments that exhibit large deviations the comparison is meaningless despite statistical test being applied. Hence, it is crucial that experiment replications are properly conducted. For proper comparison, number of runs as well as iterations should be strictly maintained for all four techniques. After MATLAB simulation with exactly same computer configuration and simulation environment, we analyzed the convergence behavior of each swarm intelligence technique for optimizing the objective function. Finally, the performance evaluation and comparison of each technique was carried out in terms of “Fitness value,” “Computational time,” and “Robustness”. Figure 2 shows the flow chart of our methodology.

4.1. Particle swarm optimization (PSO)
PSO is an evolutionary computation technique that is proposed by Eberhart and Yuhui (2001). The PSO was inspired from social behavior of bird flocking. It uses a number of particles (candidate solutions) which fly around in the search space to find best solution. Meanwhile, they all look at the best particle (best solution) as well as the best solution has found so far. Each particle in PSO should consider the current position, the current velocity, the distance to pbest, and the distance to gbest in order to modify its position.

PSO was mathematically modeled as follows:

\[ V_i^{t+1} = w_i V_i^t + c_1 \text{rand} (p_{\text{best}_i} - x_i^t) + c_2 \text{rand} (g_{\text{best}} - x_i^t) \]  

\[ x_i^{t+1} = x_i^t + V_i^{t+1} \]

where \( V_i^t \) is the velocity of particle \( i \) at iteration \( t \), \( w \) is a weighting function usually used as follows:

\[ w = \alpha_{\text{max}} - \frac{w_{\text{max}} - \alpha_{\text{min}}}{t_{\text{max}}} t_{\text{re}} \]  

### Table 1. Parameter settings of the objective function

| Parameter | Values |
|-----------|--------|
| **Fixed parameters** | Maximum power, \( P_{i,\text{max}} = 6.7 \text{ kWh} \)  
Charging station efficiency, \( \eta = 0.9 \)  
Total charging time, \( \Delta t = 20 \text{ Min} \)  
Power to each PHEV: 30 W |
| **Variables** | 0.2 \( \leq \) State-of-Charge (SoC) \( \leq 0.8 \)  
Waiting time \( \leq 30 \text{ Min} \) (1,800 S)  
16 kWh Battery capacity (\( j \)) \( \leq 40 \text{ kWh} \) |
| **Constraints** | \( \sum P_i(k) \leq P_{\text{utility}}(k) \times \eta \)  
0 \( \leq P_i(k) \leq P_{i,\text{max}}(k) \)  
0 \( \leq SoC_i(k) \leq SoC_{i,\text{max}} \)  
0 \( \leq SoC_i(k + 1) - SoC_i(k) \leq \Delta SoC_{\text{max}} \) |
There are updates of velocities as well as positions of the particles. The algorithm repeats until the maximum number of iterations or the minimum error criteria is met. Figure 3 shows the flowchart for PSO.

Appropriate values for $\omega_{\text{min}}$ and $\omega_{\text{max}}$ are 0.4 and 0.9 (Wu, Cao, Wen, & Bian, 2008). Suitable value ranges for $c_1$ and $c_2$ 1 to 2 (Su & Chow, 2011a), but 2 is most appropriate in many cases (Soares, Morais, & Vale, 2012). rand is a random number between 0 and 1 (Su & Chow, 2011a), $x_i^t$ is the current position of particle $i$ at iteration $t$, $p\text{best}_i$ is the $p\text{best}$ of agent $i$ at iteration $t$ and $g\text{best}$ is the best solution so far. PSO algorithm works by simultaneously maintaining several particles or potential solutions in the search space. For each iteration of the algorithm, each particle is evaluated by the fitness function being optimized, based on the fitness of that solution. Table 2 shows the parameter settings of PSO method.

### 4.2. Accelerated particle swarm optimization (APSO)

Accelerated PSO was developed by Xin-She Yang (Martens et al., 2011) at Cambridge University in 2007 in order to accelerate the convergence of the algorithm is to use the global best only. PSO- and APSO-based optimizations have already been studied by the researchers for optimal design of substation grounding grid (El_Fergany, 2013), performance analysis of MIMO radar waveform (Reddy, 2012), design of frame structures (Talatahari et al., 2013), dual channel speech enhancement (Prajna et al., 2014), a faster path planner (Mohamed et al., 2012), etc.

In APSO, each member of the population is called a particle and the population is called a swarm. Starting with a randomly initialized population and moving in randomly chosen directions, each particle moves through the searching space and remembers the best earlier positions, velocity and accelerations of itself and its neighbors. Particles of a swarm communicate good position, velocity and acceleration to each other as well as dynamically adjust their own position, velocity and acceleration derived from the best position of all particles. The next step starts when all particles have been shifted. Finally, all particles inclined to fly towards better positions over the searching process until the swarm move close to an optimum of the fitness function.
A simplified version that could accelerate the convergence of the algorithm is to use the global best only. Thus, in the APSO (Talatahari et al., 2013), the velocity vector is generated by a simpler formula as where \( \text{randn} \) is drawn from \((0, 1)\) to replace the second term. The update of the position is simply like-

\[
V_{t+1}^i = V_t^i + \alpha \text{randn}(t) + \beta (g^* - x_t^i)
\]

(17)

where \( \text{randn} \) is drawn from \(N(0, 1)\) and the update of the position is like the standard PSO method. In order to increase the convergence even further, the update of the position can be written in a single step as

\[
x_{t+1}^i = (1 - \beta)x_t^i + \beta g^* + \alpha r
\]

(18)

In our simulation, we use (Gandomi et al., 2013):

\[
\alpha = 0.7^t
\]

(19)

Figure 4 shows the flowchart of APSO method.

The typical values for this APSO are \( \alpha \approx 0.1-0.4 \) and \( \beta \approx 0.1-0.7 \); however, \( \alpha \approx 0.2 \) and \( \beta \approx 0.5 \) are recommended (El_Fergany, 2013). In general, any evolutionary search algorithm shows improved performance with a relatively larger population. However, a very large population will cost more in terms of fitness function evaluations without producing significant improvements. In this simulation, the population size is set to 100. The parameter settings for APSO are demonstrated in Table 3.

### 4.3. Gravitational search algorithm (GSA)

GSA is an optimization method which has been introduced by Rashedi et al. in the year of 2009 (Rashedi et al., 2009). In GSA, the specifications of each mass (or agent) are total four, which is mass (inertial), position, mass (active gravitational), and mass (passive gravitational). The position of the mass presents a solution of a particular problem, and masses (gravitational and inertial) are obtained by using a fitness function. GSA can be considered as a collection of agents (candidate solutions), whose masses are proportional to their value of fitness function.
GSA-based optimization has already been used by the researchers for post-outage bus voltage magnitude calculations, economic dispatch with valve-point effects, optimal sizing and suitable placement for distributed generation (DG) in distribution system, optimization of synthesis gas production (Ganesan et al., 2013), solving thermal unit commitment (UC) problem (Roy, 2013) and finding out optimal solution for optimal power flow (OPF) problem in a power system (Duman, Güvenç, Sönmez, & Yörükeren, 2012), etc. Specifically, we are investigating the use of the GSA method for developing real-time and large-scale optimizations for allocating power.

The gravitational force is expressed as follows:

\[ F_{d_{ij}}^{d}(t) = G(t) M_{pi}(t) \times M_{aj}(t) \left( x_{d}^{i}(t) - x_{d}^{j}(t) \right) \]  

(20)

where \( M_{aj} \) is the active gravitational mass related to agent \( j \), \( M_{pi} \) is the passive gravitational mass related to agent \( i \), \( G(t) \) is gravitational constant at time \( t \), \( \varepsilon \) is a small constant, and \( R_{ij}(t) \) is the Euclidian distance between two agents \( i \) and \( j \). The \( G(t) \) is calculated as:

\[ G(t) = G_{0} \times \exp(-a \times \text{iter} / \text{maxiter}) \]  

(21)

where \( a \) and \( G_{0} \) are descending coefficient and primary value, respectively, current iteration and maximum number of iterations are expressed as \( \text{iter} \) and \( \text{maxiter} \). In a problem space with the dimension \( d \), the overall force acting on agent \( i \) is estimated as following equation:

\[ F_{i}^{d}(t) = \sum_{j=1,j \neq i}^{N} \text{rand}_{i} F_{d_{ij}}^{d}(t) \]  

(22)

where \( \text{rand}_{i} \) is a random number with interval \([0, 1]\). From law of motion we know that an agent’s acceleration is directly proportional to the resultant force and inverse of its mass, so the acceleration of all agents should be calculated as follows:

\[ ac_{d_{i}}^{d}(t) = \frac{F_{i}^{d}(t)}{M_{i}(t)} \]  

(23)

where \( t \) is a specific time and \( M_{i} \) is the mass of the object \( i \). The velocity and position of agents are calculated as follows:

\[ \text{vel}_{i}^{d}(t + 1) = \text{rand}_{i} \times \text{vel}_{i}^{d}(t) + ac_{d_{i}}^{d}(t) \]  

(24)

\[ x_{d_{i}}^{d}(t + 1) = x_{d_{i}}^{d}(t) + \text{vel}_{i}^{d}(t + 1) \]  

(25)
where \( rand \) is a random number with interval \([0, 1]\). Moreover, the step involved in optimization using GSA is shown Figure 5.

In GSA, all agents are initialized first with random values. Each of the agents is a candidate solution. After initialization, velocities for all agents are defined using (24). Moreover, the gravitational constant, overall forces, and accelerations are determined by equations (21), (22), and (23), respectively. The positions of agents are calculated using (25). At the end, GSA will be terminated by meeting the stopping criterion of maximum 100 iterations. The parameter settings for GSA are demonstrated in Table 4.

### 4.4. Hybrid particle swarm optimization and gravitational search algorithm (PSOGSA)

Hybrid PSOGSA was introduced by Seyedali Mirjalili (Mirjalili & Hashim, 2010) at soft computing research lab of Universiti Teknologi Malaysia (UTM) in 2010 in order to integrate the ability of exploitation in PSO with the ability of exploration in GSA. PSOGSA-based optimization has already been used
by the researchers for economic load dispatch (Dubey, Pandit, Panigrahi, & Udgir, 2013), optimal static state estimation (Mallick, Ghoshal, Acharjee, & Thakur, 2013), dual channel speech enhancement (Kunch, Rao, Reddy, & Maheswari, 2015), training feed-forward neural networks (Mirjalili, Mohd Hashim, & Moradian Sardroudi, 2012), multi distributed generation planning (Tan, Hassan, Rahman, Abdullah, & Hussin, 2013), etc.

The basic idea is to fit in the exploitation capability in PSO with the exploration capability in GSA to combine both algorithms’ strength. In order to combine these two algorithms, velocity update is proposed as

\[
v_i(t + 1) = w \times v_i(t) + a' \times \text{rand} \times ac_i(t) + \beta' \times \text{rand} \times (\text{gbest} - x_i(t))
\]  

(26)

where \(v_i(t)\) is the velocity of agent \(i\) at iteration \(t\), \(w\) is a weighting factor, \(\text{rand}\) is a random number between 0 and 1, \(ac_i(t)\) is the acceleration of agent at iteration \(t\), and \(\text{gbest}\) is the best solution so far. Here, \(a'\) and \(\beta'\) are the weighting factors. With adjusting \(a'\) and \(\beta'\), the abilities of global search and local search can be balanced. The position of the particle \(x_i(t + 1)\) in each iteration is updated using the equation

\[
x_i^d(t + 1) = x_i^d(t) + \text{vel}_i^d(t + 1)
\]

(27)

The flowchart of hybrid PSOGSA method is shown in Figure 6.

| Table 4. GSA parameter settings |
|--------------------------------|
| Parameter                  | Values |
| Primary parameter          | 100    |
| No. of mass agents         | 100    |
| Acceleration coefficient   | 20     |
| Constant parameter, \(\beta\) | 0.01   |
| Power of \(“R”\)           | 1      |
| Maximum iteration          | 100    |
| Number of runs             | 50     |
PSOGSA with the parameter settings stated in Table 5 was also performed for the same fitness function and compared with the performance of GSA in terms of average best fitness. The swarm size and maximum iterations were set exactly the same to that of GSA and PSO techniques for the comparison purpose. The values of parameters $c_1$, $c_2$, and alpha were set as standard values, 0.5, 1.5, and 23, respectively.

5. Simulation results of applied swarm intelligence techniques

In order to optimize state-of-charge, with respect to charging time, present SoC. For this, swarm intelligence-based methods, PSO and GSA, APSO and Hybrid version of PSO and GSA (PSOGSA) were applied. All the optimization techniques were simulated to achieve the best fitness values of objective function stated at equation number 13. All the simulations were run on the following computer configuration stated below:

Table 5. Parameter settings of PSOGSA

| Parameter               | Values |
|-------------------------|--------|
| Size of the swarm       | 100    |
| GSA Constant parameter  | 23     |
| PSO parameter, $c_1$    | 0.5    |
| PSO parameter, $c_2$    | 1.5    |
| Gravitational constant  | 1      |
| Maximum Iteration       | 100    |
| Number of runs          | 50     |
5.1. Particle swarm optimization (PSO)
Figure 7 shows the convergence behavior (iteration vs. fitness value) of PSO technique for 100 PHEVs. It can be apparently seen from the simulation study that although the algorithm has been set to run maximum 100 iterations, the fitness value converges before five (05) iterations for all five scenarios and become stable. Therefore, an early convergence may cause the fitness function to trap into local minima. This can be avoided by increasing the size of swarm hence the computational time will also be increased as well. As a result, a trade-off should be taken into consideration between the proper convergence and computational time.

Figure 8 depicts the best fitness value for 100 PHEVs. In this case, the maximum best fitness and minimum best fitness were 767.8722 and 9.5076, respectively. The average best fitness increases up to 182.9313.

As PSO is a population-based optimization technique and the fitness function is nonlinear, so the fitness values fluctuate for each iteration (El-Fergany, 2013; Rahman et al., 2016c; Soares et al., 2012;
However, the maximum best fitness remains in the range of 650 to 950 and the minimum best fitness remains in the range of 0.70–8. Table 6 summarizes the result. From that it can be concluded that average best fitness remain almost in similar pattern for four (05) different scenarios.

Table 7 shows the average computational time requirement for PSO method. The average computational time for 50 PHEVs is 1.620 s while for 1,000 PHEVs it increases up to 2.328 s.

### 5.2. Accelerated particle swarm optimization (APSO)

Figure 9 shows the convergence behavior (iteration vs. fitness value) of APSO technique for 100 PHEVs. Although the algorithm has been set to run for maximum 100 iterations, the fitness value converges after 10 iterations and becomes stable. Consequently, there is an early convergence that may cause the fitness function to trap into local minima. This can be avoided by increasing the size of swarm hence the computational time will also be increased as well. For instance, a trade-off should be taken into consideration between the proper convergence and computational time.

Figure 10 depicts the best fitness value for 100 PHEVs. In this case, the maximum best fitness and minimum best fitness were 679.7151 and 9.5076, respectively. The average best fitness increases up to 182.9313. In order to evaluate the performance and show the efficiency and superiority of the proposed algorithm, we ran each scenario a total of 50 times.

As APSO is a population-based optimization technique and the fitness function is nonlinear, so the fitness values fluctuate for each iteration (El-Fergany, 2013; Rahman et al., 2016c; Soares et al., 2012; Wu et al., 2008). However, the maximum best fitness remains in the range of 450–700 and the minimum best fitness remains in the range of 0.5–10. Table 8 summarizes the result. From that it can be concluded that average best fitness remains almost in similar pattern for five different scenarios.

Table 9 shows the average computational time requirement for APSO method. The average computational time for 50 PHEVs is 1.696 s while for 1,000 PHEVs it increases up to 2.092 s.

### 5.3. Gravitational search algorithm (GSA)

Many optimization algorithms involve local search techniques which can get stuck on local maxima. Most search techniques strive to find a global maximum in the presence of local maxima (Amini & Arif, 2014). One of the most important characteristic of GSA is its significant performance during
exploration process. The capability of an algorithm to extend the problem in search gap is known as exploration while the ability of an algorithm to recognize optimal solution near a favorable one is exploitation (Kulshrestha et al., 2009; Su & Chow, 2011b).

Figure 11 shows the convergence behavior of GSA for 100 PHEVs. From the simulation study we know that the best fitness function convergences after same iterations (35 iterations) for both 50 and 100 numbers of PHEVs while for 500 and 1,000 numbers of PHEVs, it shows early convergence (converges before 20 iterations).

Figure 12 illustrates the best fitness value for 100 PHEVs by using GSA method. In this case, the maximum best fitness and minimum best fitness were 872.648 and 1.005. The average best fitness decreases up to 182.309.

Finally, Table 10 sums up the results for GSA.

Table 11 shows the average computational time requirement for GSA method.

5.4. Hybrid particle swarm optimization and gravitational search algorithm (PSOGSA)

Figure 13 shows the convergence behavior (iteration vs. fitness value) of Hybrid PSOGSA technique for 100 PHEVs. Here also the algorithm has been set to run for maximum 100 iterations, the fitness
value converges after five (05) iterations and becomes stable. So, there is an early convergence which may cause the fitness function to trap into local minima. This can be avoided by increasing the size of swarm hence the computational time will also be increased as well. In order to evaluate the performance and show the efficiency and superiority of the proposed algorithm, we ran each scenario a total of 50 times.

Figure 14 depicts the best fitness value for 100 PHEVs. In this case, the maximum best fitness and minimum best fitness were 625.82 and 3.39, respectively. The average best fitness decreases up to 184.36.

As PSOGSA is a population-based optimization technique and the fitness function is nonlinear, so the fitness values fluctuate for each iteration (Abro & Mohamad-Saleh, 2012; Ganesan, Vasant, & Elamvazuthi, 2014; Jiménez, Sánchez, & Vasant, 2013). However, the maximum best fitness remains in the range of 400–950 and the minimum best fitness remains in the range of 0.1–8.

Table 12 summarizes the result. From that it can be concluded that average best fitness remains almost in similar pattern for four (05) different scenarios.

Table 13 shows the average computational time requirement for APSO method. The average computational time for 50 PHEVs is 4.228 s while for 1,000 PHEVs it increases up to 72.408 s.

| Table 8. Fitness evaluation of APSO |
|-------------------------------------|
| J(0) | 50 PHEVs | 100 PHEVs | 30 PHEVs | 500 PHEVs | 1,000 PHEVs |
| Max. best fitness | 469.75 | 679.71 | 679.55 | 615.83 | 678.92 |
| Average best fitness | 162.70 | 168.23 | 147.42 | 184.15 | 171.16 |
| Min. best fitness | 7.65 | 3.46 | 3.54 | 5.96 | 0.99 |

| Table 9. Average computational time for APSO |
|-----------------------------------------------|
| Number of PHEVs | Computational time (s) |
|-----------------|------------------------|
| 50 PHEVs | 1.69 |
| 100 PHEVs | 1.71 |
| 300 PHEVs | 1.76 |
| 500 PHEVs | 1.83 |
| 1,000 PHEVs | 2.09 |

Figure 11. Iteration vs. fitness value, J(k) for GSA (100 PHEVs).
6. Comparisons among swarm intelligence techniques

This section deals with the comparisons among the applied swarm intelligence-based optimization techniques. All four techniques were run on the same computer along with the same iterations (100) and total 50 independent runs in order to ensure a fair comparison (Rahman et al., 2016c); (Derrac, García, Molina, & Herrera, 2011). The comparisons among applied swarm intelligence-based techniques are given below.

### 6.1. Stopping criteria

In any swarm intelligence algorithm there are some initial solutions from which candidate solutions are created. After that, each solution is evaluated and the algorithm chooses the best solution. If the stopping criteria is met, then the algorithm will produce a final solution otherwise it will again search for best solutions from the initial step. Proper balance between exploration and exploitation is the basic criteria to analyze the performance of an algorithm. Proper exploration will diversify the search space of an optimization technique whereas the exploitation ensures the high-quality solutions.

Since an iterative method computes successive approximations to the solution of a system, stopping criteria is needed to determine when to stop the iteration. The maximum number of iteration

---

**Table 10. Fitness evaluation of GSA**

| J(k)       | 50 PHEVs | 100 PHEVs | 30 PHEVs | 500 PHEVs | 1,000 PHEVs |
|------------|----------|-----------|----------|-----------|-------------|
| Max. best fitness | 781.13   | 872.65    | 743.13   | 836.27    | 968.77      |
| Average best fitness | 158.83   | 182.31    | 172.43   | 152.36    | 161.52      |
| Min. best fitness   | 0.22     | 1.00      | 2.33     | 0.98      | 7.27        |

**Table 11. Average computational time for GSA**

| Number of PHEVs | Computational time (s) |
|-----------------|------------------------|
| 50 PHEVs        | 2.72                   |
| 100 PHEVs       | 4.44                   |
| 300 PHEVs       | 11.28                  |
| 500 PHEVs       | 18.17                  |
| 1,000 PHEVs     | 36.28                  |
was set to 100 for all four optimization techniques. Previous researchers use 100 iterations for their simulation study (Hendtlass, 2007; Martens et al., 2011).

6.2. Convergence analysis

When an algorithm finds an optimal solution to a given problem, one of the important factors is speed and rate of convergence to the optimal solution (Martens et al., 2011). Among the four techniques, the convergence of PSO and APSO techniques is of same pattern while GSA takes higher number of iterations to be converged. Among all four techniques, the hybrid method: PSOGSA takes the least number of iterations for convergence. So, there is an early convergence which may cause the fitness function to trap into local minima. This can be avoided by increasing the size of swarm hence the computational time will also be increased as well. As a result, a trade-off should be taken into consideration between the proper convergence and computational time. Table 14 shows the number of iterations needed to be converged for each algorithm for five different cases.

Table 12. Fitness evaluation of PSOGSA

|       | 50 PHEVs | 100 PHEVs | 30 PHEVs | 500 PHEVs | 1,000 PHEVs |
|-------|----------|-----------|----------|-----------|-------------|
| Max. best fitness | 931.03 | 625.82 | 434.16 | 454.04 | 740.40 |
| Average best fitness | 184.36 | 188.67 | 181.03 | 186.70 | 185.16 |
| Min. best fitness | 3.39 | 3.71 | 7.43 | 7.23 | 0.17 |
6.3. Best fitness value

Best fitness value presents the solutions for optimization technique applying to a particular objective function upon given constraints and set of parameters. The best fitness value represents the strength of any optimization technique. For the maximization problems, higher best fitness values indicate the effectiveness of particular techniques whereas the lower fitness values show the effectiveness of any minimization problems. As our optimization goal is maximization, so the higher best fitness values are the best solution.

Maximum best fitness, average best fitness, and minimum best fitness are presented in order to evaluate the performance of applied optimization techniques. According to the simulation result, the hybrid method, PSOGSA shows best fitness values for all five cases (50, 100, 300, 500, and 1,000 PHEVs). The fitness value comparison among all techniques are shown in Figure 15. Moreover single techniques like GSA and APSO show overall better result compared to PSO technique.

Table 13. Average computational time for GSA

| Number of PHEVs | Computational time (s) |
|----------------|------------------------|
| 50 PHEVs       | 4.228                  |
| 100 PHEVs      | 7.902                  |
| 300 PHEVs      | 22.326                 |
| 500 PHEVs      | 36.824                 |
| 1,000 PHEVs    | 72.408                 |

Table 14. Convergence iteration

| Number of PHEVs | Number of iterations taken to be converged |
|-----------------|-------------------------------------------|
|                 | PSO | APSO | GSA | PSOGSA |
| 50              | 5   | 2    | 30  | 2      |
| 100             | 6   | 3    | 35  | 2      |
| 300             | 5   | 3    | 15  | 2      |
| 500             | 7   | 5    | 40  | 2      |
| 1,000           | 7   | 6    | 5   | 3      |

Figure 15. Average best fitness vs. number of PHEVs.
6.4. Computational time
In order to maintain fair comparison, all the simulation runs on same computer as well as same swarm size (population = 100) and iteration (100). Total running time of each optimization technique is also same (total 50 runs). If the number of runs is less for example, 10 or 20 runs only then the purpose of fair comparison will be hampered. Optimizing the design is done using an advanced optimization algorithm, which requires running a large number of simulations. This can be much more efficient than running a parameter sweep, particularly if there is more than one parameter to optimize (Martens et al., 2011). Figure 16 shows the average computational time comparison of all four optimization techniques considering five cases.

PSOGSA—the hybrid technique takes the highest time to complete 100 iterations whereas both PSO and APSO techniques show better result in terms of computation time. The GSA method takes higher time than other single methods, PSO and APSO in order to find best fitness for objective function J (k). Based on the previous literatures, the original GSA has some weaknesses such as using complex operators and long computational time. GSA also suffers from slow searching speed in the last iterations (Mirjalili & Hashim, 2010). Another problem is the difficulty for the appropriate selection of gravitational constant parameter, G. The parameter controls the search accuracy and does not guarantee a global solution at all time.

6.5. Robustness
Robustness is based on the ability of an optimization problem to perform well over a wide range of population (Shaikh, Nor, Nallagownden, & Elamvazuthi, 2013). Furthermore, optimization strategies and parameters must remain either constant over the set of problems or should be automatically set using individual test problems attributes.

| Table 15. Paired t-test |
|-------------------------|
| Paired t-test           | Paired differences | t Value | Sig. (2-tailed) |
|                         | Mean               | Standard deviation | Standard error mean | 95% Confidence interval of the difference | Lower | Upper |           |        |
| Pair 1 PSO-PSOGSA       | −28.21             | 13.69              | 6.12               | −45.21 −11.21 | −4.60 | 0.010 |
| Pair 2 APSO-PSOGSA      | −18.45             | 11.36              | 5.08               | −32.56 −4.33 | −3.63 | 0.022 |
| Pair 3 GSA-PSOGSA       | −19.69             | 11.88              | 5.31               | −34.45 −4.93 | −3.70 | 0.021 |
A paired t-test was performed on the simulation results in order to determine which method outperforms others. In statistical significance testing, the p-value is the probability of obtaining a test statistic at least as extreme as the one that was actually observed, assuming that the null hypothesis is true. A small p-value indicates that the null hypothesis is less likely to be true. Let \( J_{PSOGSA} \) is the average best fitness value achieved by the PSOGSA, while \( J_p \) is the average best fitness value achieved by the others methods (PSO, APSO, and GSA). First, the comparison was performed between PSOGSA’s performance based on average best fitness values with PSO for a total of five different scenarios (starting from 50 PHEVs up to 1,000 PHEVs). Then, the null hypotheses is defined as

\[
H_0: J_{PSOGSA} \leq J_{PSO}
\]  

This was tested against the alternative hypotheses

\[
H_1: J_{PSOGSA} > J_{PSO}
\]

Here, the comparisons were performed between PSOGSA with other three optimization methods (PSO, APSO, and GSA) and shown in Table 15. From the table it is clear that the significant level p-value is 0.01 (less than 0.05). The lower the p-value is, the less likely the null hypothesis is true. Thus we reject the null hypothesis and draw the conclusion that there is significant evidence that PSOGSA can achieve better results in terms of average best fitness.

By following a similar procedure as described before, p-values for the other cases (APSO-PSOGSA and GSA-PSOGSA) were obtained less than 0.05. Therefore, the null hypotheses were rejected and concluded that PSOGSA method is robust than other methods in solving this specific problem.

7. Suggestions for future work
This section suggests future directions of optimization techniques and procedures. The specific research field is relatively new and possible future perspectives have to be emphasized, so that new techniques can be realized.

The concepts proposed here may be utilized and tested for a wider range of multi-objective problems with a variety of problems characteristics in the future (Jiménez et al., 2013). Besides, objective function, \( J(k) \) should also be tested with other swarm intelligence algorithms like ACO, ABC optimization, Firefly Algorithm (FA). Although swarm intelligence-based methods have established their capability to explore large search spaces, they are comparatively incompetent in fine-tuning the
solution. This weakness is usually avoided by means of local search method that is applied to the 
individuals of the population (Derrac et al., 2011). In our case, further studies can be carried out by 
hybridizing PSO or GSA (swarm intelligence-based algorithm) with local search method. The future 
optimization tools should be capable of performing parallel processing evaluations on the same 
computer by using modern multi-core processor technology or to distribute the calculations to a 
cluster of computers. Such ability will substantially improve the simulation runtime. Advanced con-
trolling mechanisms (Su, Wang, & Hu, 2015) are necessary for allocating sufficient energy to a par-
ticular charging station in order to facilitate large-scale PHEV penetration in upcoming years (Su, 
Zeng, & Chow, 2012). The future optimization tools should have the capability of stable convergence 
and thus provide good solution to the desired fitness functions. Exploration and exploitation of the 
search space is essential in order to get desired solution within acceptable computation time. Finally, 
optimization of charging station needs proper assortment of available resources as well as efficient 
available technique implementation. Proper charging infrastructure management can assist the 
larger participation of PHEVs.

At the same time, researchers should try to improve available device mechanism for the infra-
structure with a view to simplify future PHEVs dispersion in roads and highways. In future, more 
vehicles should be considered for intelligent power allocation strategy as well as other single and 
hybrid techniques should be applied to ensure higher fitness value and low computational time.

8. Prototype model
Researchers are trying to design efficient controller for charging station and several literatures on 
optimization-based methods were published in this wake. These vehicles will help the government 
in its role of promoting energy security and environmental protection, when successfully marketed 
to consumers [123]. Efforts are also to be taken for provision of affordable and accessible infrastruc-
ture for recharging (Su et al., 2015). Hence, thrust in research and development on the aforemen-
tioned design considerations and technological challenges coupled with government support in 
terms of incentives to the automobile owners and to the manufacturers will go a long way in accel-
erating the deployment of large-scale PHEVs. The Figure 17 shows the prototype of digital testbed 
for Large-scale PHEV/PEV Charging Infrastructure from Future Renewable Electric Energy Delivery 
and Management (FREEDM) Systems Center with Advanced Diagnosis Automation and Control 
(ADAC) Lab at North Carolina State University and Advanced Transportation Energy Center (ATEC) 
(Su et al., 2012). The applied swarm intelligence-based algorithms are a step towards real-life imple-
mentation of such controller for PHEV charging stations.

Proper charging infrastructure management can assist the larger participation of PHEVs. At the 
same time, researchers should try to improve available device mechanism for the infrastructure with 
a view to simplify future PHEVs dispersion in roads and highways. In future, more vehicles should be 
considered for intelligent power allocation strategy as well as other single and hybrid techniques 
should be applied to ensure higher fitness value and low computational time.

9. Conclusion
SoC is needed to be optimized in order to develop the future charging infrastructures for PHEVs. The 
objective function is highly nonlinear which makes the optimization problem as a complex one. 
Simple Linear programming (LP) is not useful for solving this kind of problem. Swarm intelligence 
methods are within the group of metaheuristic algorithms. A metaheuristic is high-level problem-
independent algorithmic framework that provides a set of guidelines or strategies to develop heuris-
tic optimization algorithms. The aim of this work was to a comprehensive framework to solve 
single-objective optimization problems by analyzing the effects of best fitness and computational 
time using four swarm intelligence techniques: PSO, APSO, GSA, and Hybrid PSOGSA. In addition, this 
work was also targeted to analyze and compare the performance of swarm intelligence algorithms 
for PHEV charging station. The effects of convergence and robustness on the performance of applied 
techniques have been studied rigorously for solving charging problems.
Among the all four swarm intelligence-based optimization methods, three are single optimization techniques (PSO, APSO, and GSA) and one is hybrid version (PSOGSA). The convergence speed is the fastest in PSOGSA whereas GSA takes more iteration to be converged. In order to optimize the objective function J(k), a hybrid technique (PSOGSA) has been introduced for the first time along with other three single techniques such as PSO, APSO, and GSA for comparative study. One of the comparison scales is best fitness value. Maximum best fitness, average best fitness, and minimum best fitness are shown in order to evaluate the performance of applied optimization techniques. Total five scenarios in terms of PHEVs number in the charging station are taken into consideration for the simulation starting from 50 PHEVs up to 1,000. These five scenarios show how the best fitness value and computational time changes with the increase of PHEVs in a charging station per day.

Acknowledgment
The authors would like to thank Universiti Teknologi PETRONAS (UTP) for supporting the research under UTP Graduate Assistantship (GA) scheme.

Funding
This work was supported by Graduate Assistance Scheme of Universiti Teknologi PETRONAS.

Author details
Pandian M. Vasant1
E-mails: pandian_m@utp.edu.my, pvasant@gmail.com
Imran Rahman1
E-mail: imran.uitoic@gmail.com
ORCID ID: http://orcid.org/0000-0002-6732-8751
Balbir Singh Mahinder Singh2
E-mail: balbir@petronas.com.my
M. Abdullah-Al-Wadud1
E-mail: mwadud@ksu.edu.sa

1 Department of Fundamental and Applied Sciences, Universiti Teknologi PETRONAS 32610 Seri Iskandar, Perak, Malaysia.
2 Department of Software Engineering, College of Computer and Information Sciences, King Saud University, Riyadh, Saudi Arabia.

Citation information
Cite this article as: Optimal power allocation scheme for plug-in hybrid electric vehicles using swarm intelligence techniques, Pandian M. Vasant, Imran Rahman, Balbir Singh Mahinder Singh & M. Abdullah-Al-Wadud, Cogent Engineering (2016), 3: 1203083.

References
Abro, A. G., & Mohamad-Saleh, J. (2012). An enhanced artificial bee colony optimization algorithm. Recent Advances in Systems Science and Mathematical Modelling (pp. 222–227).
Amini, M. H., & Arif, I. (2014). Allocation of electric vehicles’ parking lots in distribution network. IEEE PES 5th Innovative Smart Grid Technologies Conference (ISGT 2014), February 19–22. Washington, DC.
Amini, M. H., & Parsa Moghaddam, M. (2013). Probabilistic modelling of electric vehicles’ parking lots charging demand. Electrical Engineering (ICEE), 2013 21st Iranian Conference on. IEEE.
Boyle, G. (2007). Renewable electricity and the grid: the challenge of variability. Earthscan.
Chang, W.-Y. (2013). The state of charge estimating methods for battery: A review. ISRN Applied Mathematics, 2013.
Chassaing, J., & Varanmohan, B. (2009). Estimating the state of charge of a battery. IEEE Transactions on Control Systems Technology, 13, 465–470. http://dx.doi.org/10.1109/TCST.2004.839571
Contestabile, M., Offer, G. J., Slade, R., Joeger, F., & Thoennes, M. (2011). Battery electric vehicles, hydrogen fuel cells and biofuels. Which will be the winner? Energy & Environmental Science, 4, 3754–3772.
Derrac, J., Garcia, S., Molina, D., & Herrera, F. (2011). A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms. Swarm and Evolutionary Computation, 1, 3–18. http://dx.doi.org/10.1016/j.swevo.2011.02.002
Dorigo, M. (2006). Ant colony optimization and swarm intelligence. 5th International Workshop ANTS 2006, Brussels, Belgium, September 6–7, 2006, Proceedings vol. 4150. Springer.
Dubey, H. M., Pandit, M., Panighrahi, B., & Udigr, M. (2013, July). Economic load dispatch by hybrid swarm intelligence based gravitational search algorithm. International Journal of Intelligent Systems & Applications, 5, 21–32, doi:10.5815/jija.2013.08.03.
Duman, S., Güvenç, U., Sönmez, Y., & Yörüküren, N. (2012). Optimal power flow using gravitational search algorithm. Energy Conversion and Management, 59, 86–95. http://dx.doi.org/10.1016/j.enconman.2012.02.024
Eberhart, R. C., & Yuhui, S. (2001). Particle swarm optimization: Developments, applications and resources. In Evolutionary Computation, 2001. Proceedings of the 2001 Congress on, (Vol. 1, pp. 81–86). http://dx.doi.org/10.1109/ICEC.2001.934374
El-Fergany, A. (2013, July). Accelerated particle swarm optimization-based approach to the optimal design of substation grounding grid. Przegląd Elektrotechniczny, 49, 30–34.
Gandomi, A. H., Yun, G. J., Yang, X.-S., & Talatahari, S. (2013). Chaotic enhanced accelerated particle swarm optimization. Communications in Nonlinear Science and Numerical Simulation, 18, 327–340. http://dx.doi.org/10.1016/j.cnsns.2012.07.017
Ganesan, T., Elamvazuthi, I., Ku Shaari, K. Z., & Vasant, P. (2013). Swarm intelligence and gravitational search algorithm for multi-objective optimization of synthesis gas production. Applied Energy, 103, 368–374, 3/2013. http://dx.doi.org/10.1016/j.apenergy.2012.09.059
Ganesan, T., Vasant, P., & Elamvazuthi, I. (2011). Hopfield neural networks approach for design optimization of hybrid power systems with multiple renewable energy sources in a fuzzy environment. Journal of Intelligent and Fuzzy Systems, 26, 2143–2154.
Hannan, M. A., Azidin, F. A., & Mohamed, A. (2014). Hybrid electric vehicles and their challenges: A review. Renewable and Sustainable Energy Reviews, (Vol. 29, pp. 135–150), 1/2014. http://dx.doi.org/10.1016/j.rser.2013.08.097
Hendtlass, T. (2007). Fitness estimation and the particle swarm: Recent developments, applications and resources. In Proceeding of the Congress on, (Vol. 1, pp. 81–86). http://dx.doi.org/10.1109/ICEC.2007.4425028
Hess, A., Malandrin, F., Reinhardt, M. B., Cosetti, C., Hummel, K. A., & Barceló-Ordinas, J. M. (2012, December). Optimal deployment of charging stations for electric vehicular networks. In Proceedings of the first workshop on Urban networking (pp. 10–13). Nice, France. http://dx.doi.org/10.1145/2413236
Hota, A. R., Juvvaniapudi, M., & Bajpai, P. (2014). Issues and solution approaches in PHEV integration to smart grid Renewable and Sustainable Energy Reviews, 30, 217–229, 2/2014. http://dx.doi.org/10.1016/j.rser.2013.10.008

Jiménez, F., Sánchez, G., & Vasant, P. (2013). A multi-objective evolutionary approach for fuzzy optimization in production planning. Journal of Intelligent & Fuzzy Systems: Applications in Engineering and Technology, 25, 441–455.

Karaboga, D., & Basturk, B. (2007). A powerful and efficient algorithm for numerical function optimization—Artificial bee colony (ABC) algorithm. Journal of Global Optimization, 39, 459–471. http://dx.doi.org/10.1007/s10898-007-9149-x

Kulshrestha, P., Wang, L., Chow, M.-Y., & Lukic, S. (2008). Integration of renewable energy into the transport and electricity sectors through V2G. Energy Policy, 36, 3578–3587. http://dx.doi.org/10.1016/j.enpol.2008.06.007

Mallick, S., Ghoshal, S. P., Acharjee, P., & Thakur, S. (2013). Optimal static state estimation using improved particle swarm optimization and gravitational search algorithm. International Journal of Electrical Power & Energy Systems, 52, 254–265.

Martens, D., Boensens, B., & Fawcett, T. (2011). Editorial survey: swarm intelligence for data mining. Machine Learning, 82, 1–42. http://dx.doi.org/10.1007/s10994-010-5216-5

Mirjalili, S., & Hashim, S. M. (2010). A new hybrid PSOGSA algorithm for function optimization. In Computer and Information Application (ICCIA), 2010 International Conference on (pp. 374–377). Tianjin: IEEE.

Mirjalili, S., Mohd Hashim, S. Z., & Moradian Sardroudi, H. (2012). Training feedforward neural networks using hybrid particle swarm optimization and gravitational search algorithm. Applied Mathematics and Computation, 218, 11125–11137. http://dx.doi.org/10.1016/j.amc.2012.04.069

Mohamed, A. Z., Lee, S. H., Hsu, H. Y., & Nath, N. (2012). A faster path planner using accelerated particle swarm optimization. Artificial Life and Robotics, 17, 233–240. http://dx.doi.org/10.1007/s10462-012-9572-3

Morrow, K., Karner, D., & Frantock, J. (2008). Plug-in hybrid electric vehicle charging infrastructure review (Final Report Battelle Energy Alliance Contract No. 58517). The Idaho National Laboratory is a U.S. Department of Energy National Laboratory Operated by Battelle Energy Alliance Contract No. 58517). The Idaho National Laboratory Operated by Battelle Energy Alliance.

Mwasili, F., Justo, J. I., Kim, E.-K., Do, T. D., & Jung, J.-W. (2014). Electric vehicles and smart grid interaction: A review on vehicle-to-grid and smart grid interaction. A review on vehicle-to-grid and renewable energy sources integration Renewable and Sustainable Energy Reviews, 34, 501–516, 6/2014. http://dx.doi.org/10.1016/j.rser.2014.03.031

Piller, S., Perrin, M., & Jossen, A. (2001). Methods for state-of-charge determination and their applications. Journal of Power Sources, 96, 113–120. http://dx.doi.org/10.1016/S0378-7753(01)00560-2

Projin, K., Rao, G. S. B., Reddy, K., & Matheswari, R. U. (2014). A new dual channel speech enhancement approach based on accelerated particle swarm optimization (APSO). International Journal of Intelligent Systems and Applications (IJISA), 6, 1. http://dx.doi.org/10.5815/ija

Rahman, I., Vasant, P. M., Singh, B. S. M., Abdullah-Al-Wadud, M., & Adnan, N. (2016). Review of recent trends in optimization techniques for plug-in hybrid, and electric vehicle charging infrastructures. Renewable and Sustainable Energy Reviews, 58, 1039–1047. http://dx.doi.org/10.1016/j.rser.2015.12.353

Rahman, I., Vasant, P. M., Singh, B. S. M., & Abdual-Al-Wadud, M. (2016c). Novel metaheuristic optimization strategies for plug-in hybrid electric vehicles: A holistic review Intelligent Decision Technologies, 10, 149–163. http://dx.doi.org/10.3233/IDT-150245

Rashedi, E., Nezamabadi-pour, H., & Saryazdi, S. (2009). GSA: A Gravitational Search Algorithm. Information Sciences, 179, 2232–2248. http://dx.doi.org/10.1016/j.ins.2009.03.004

Reddy, B. R. (2012, August). Performance Analysis of MIMO Radar Waveform using Accelerated Particle Swarm Optimization Algorithm. Signal & Image Processing : An International Journal (SIPIJ). arXiv preprint arXiv:1209.4015, 3, 193–202.

Roy, P. K. (2013). Solution of unit commitment problem using gravitational search algorithm. International Journal of Electrical Power & Energy Systems, 53, 85–94.

Shaikh, P. H., Nor, N. M., Nallagowden, P., & Elamvazuthi, J. (2013). Intelligent Optimized Control System for Energy and Comfort Management in Efficient and Sustainable Buildings. Procedia Technology, 11, 99–106. http://dx.doi.org/10.1016/j.proct.2013.12.167

Soares, J., Morais, H., & Vole, Z. (2012). Particle swarm optimization based approaches to vehicle-to-grid scheduling. In Power and Energy Society General Meeting, 2012 IEEE (pp. 1–8). San Diego, CA: IEEE.

Soares, J., Sousa, T., Morais, H., Vole, Z., Canizes, B., & Silva, A. (2013). Application specific modified particle swarm optimization for energy resource scheduling considering vehicle-to-grid. Applied Soft Computing, 13, 4264–4280.

Su, W., & Chow, M.-Y. (2011a). Performance evaluation of a PHEV parking station using particle swarm optimization. In Power and Energy Society General Meeting, 2011 IEEE (pp. 1–6). San Diego, CA: IEEE.

Su, W., & Chow, M.-Y. (2011b). Sensitivity analysis on battery model for large-scale PHEV/PEV charging algorithms. In IECON 2011-37th Annual Conference on IEEE Industrial Electronics Society (pp. 3248–3253). Melbourne, VIC: IEEE. http://dx.doi.org/10.1109/IECON.2011.6119831

Su, W., & Chow, M.-Y. (2012). Computational intelligence-based energy management for a large-scale PHEV/PEV enabled municipal parking deck. Applied Energy, 96, 171–182. http://dx.doi.org/10.1016/j.apenergy.2011.11.088

Su, W., Eichi, H., Zeng, W., & Chow, M.-Y. (2012). A survey on the electrification of transportation in a smart grid environment. IEEE Transactions on Industrial Informatics, 8, 1–10. http://dx.doi.org/10.1109/TII.2011.2172454

Su, W., Zeng, W., & Chow, M.-Y. (2012). A digital testbed for a PHEV/PEV enabled parking lot in a smart grid environment. In Innovative Smart Grid Technologies (ISGT), IEEE PES, 2012 (pp. 1–7).
Su, W., Wang, J., & Hu, Z. (2015). Planning, Control, and Management Strategies for Parking Lots for PEVs. In S. Rajakaruna, F. Shahnia, A. Ghosh (Eds.), Plug In Electric Vehicles in Smart Grids (pp. 61–98). Singapore: Springer. doi:10.1007/978-981-287-299-9_3

Talatahari, S., Khatibi, E., & Alaviodeh, S. (2013). Accelerated particle swarm for optimum design of frame structures. Mathematical Problems in Engineering, 2013. Article ID 649857, 6p. http://dx.doi.org/10.1155/2013/649857

Tan, W. S., Hassan, M. Y., Rahman, H. A., Abdullah, M. P., & Hussin, F. (2013). Multi-distributed generation planning using hybrid particle swarm optimisation-gravitational search algorithm including voltage rise issue. IET Generation, Transmission & Distribution, 7, 929–942.

Tie, S. F., & Tan, C. W. (2013). A review of energy sources and energy management system in electric vehicles. Renewable and Sustainable Energy Reviews, 20, 82–102. http://dx.doi.org/10.1016/j.rser.2012.11.077

Wu, X., Cao, B., Wen, J., & Bion, Y. (2008). Particle swarm optimization for plug-in hybrid electric vehicle control strategy parameter. In Vehicle Power and Propulsion Conference, VPPC’08 IEEE, 2008(pp. 1–5). Harbin: IEEE.

Yang, J., He, L., & Fu, S. (2014). An improved PSO-based charging strategy of electric vehicles in electrical distribution grid Applied Energy, 128, 82–92. 9/1/2014. http://dx.doi.org/10.1016/j.apenergy.2014.04.047