Applied Energy 151 (2015) 192–205

Contents lists available at ScienceDirect

Applied Energy

journal homepage: www.elsevier.com/locate/apenergy

Metaheuristic optimization methods for a comprehensive operating schedule of battery, thermal energy storage, and heat source in a building energy system

Shintaro Ikeda [a,*], Ryozo Ooka [b]

[a] Department of Architecture, The University of Tokyo, 4-6-1 Komaba, Meguro-ku, Tokyo 153-8505, Japan
[b] Institute of Industrial Science, The University of Tokyo, 4-6-1 Komaba, Meguro-ku, Tokyo 153-8505, Japan

HIGHLIGHTS

- We proposed metaheuristic optimization methods for energy systems.
- The proposed method, m-PSO can calculate the optimal solution quickly and accurately.
- The proposed method can find a solution 62,068 times as fast as previous method.
- The proposed methods can solve nonlinear and non-differentiable problems quickly.

ARTICLE INFO

Article history:
Received 14 October 2014
Received in revised form 27 February 2015
Accepted 8 April 2015

Keywords:
Metaheuristics
Cuckoo search
m-PSO
Dynamic programming
Battery
Thermal energy storage

ABSTRACT

Storage equipment, such as batteries and thermal energy storage (TES), has become increasingly important recently for peak-load shifting in energy systems. Mathematical programming methods, used frequently in previous studies to optimize operating schedules, can always be used to derive a theoretically optimal solution, but are computationally time consuming. Consequently, we use metaheuristics, such as genetic algorithms (GAs), particle swarm optimization (PSO), and cuckoo search (CS), to optimize operating schedules of energy systems that include a battery, TES, and an air-source heat pump. In this paper, we used a GA, differential evolution (DE), our own proposed mutation-PSO (m-PSO), CS, and the self-adaptive learning bat algorithm (SLBA), of which m-PSO was the fastest, and CS was the most accurate. CS obtained the semi-optimal solution 135 times as fast as dynamic programming (DP), a mathematical programming method with 0.22% tolerance. Thus, we showed that metaheuristics, especially m-PSO and CS, have advantages over DP for optimization of the operating schedules of energy systems that include a battery and TES.

1. Introduction

In recent years, renewable power generators, such as wind turbines (WTs) and photovoltaics (PVs), have been increasingly installed in energy grids owing to feed-in tariffs and declining installation costs. The number of installations of renewable power generators is expected to increase [1]. Storage equipment has been installed with WT or PV to avoid electricity grid fluctuation and intermittency [2]. In addition, batteries have a significant role in reducing operating costs in the building sector. Thermal energy storage (TES) with combined heat and power (CHP) and heat pump has a similar role in that sector. Although optimal operation is important in maximizing their roles, it is a complex problem, because there are many things to consider when optimizing their operation, such as outdoor temperature, machine characteristics, and the price of electricity. Therefore, it is important to study energy system optimization.

There have been many previous studies of this topic [3–10]. Omu et al. [3] used mixed-integer linear programming (MILP) to minimize annual investment and operating costs of a distributed energy system. Basu and Chowdhury [4] used the cuckoo search (CS) algorithm to optimize economic dispatch problems of generators on a microgrid. Chandrasekaran and Simon [5] used CS to solve the unit commitment problem (UCP) and economic dispatch problem (EDP) using a fuzzy algorithm. Fazlollahi and Marechal [6] proposed a hybrid method with an evolutionary algorithm and MILP

* Corresponding author. Tel.: +81 3 5452 6434; fax: +81 3 5452 6432.
E-mail addresses: s-ikeda@iis.u-tokyo.ac.jp, shintaro.uni.pg@gmail.com (S. Ikeda).

http://dx.doi.org/10.1016/j.apenergy.2015.04.029
0306-2619/

Nomenclature

acd^t	amount of charging/discharging of electricity at tth time interval (kW)
acr^t	amount of storing/releasing of thermal energy at tth time interval (kW)
c_1	coefficient of returning to the past personal best position of PSO
c_2	coefficient of moving to the best position in all individuals of PSO
C_b	capacity of a battery (kW h)
C_{TES}	capacity of TES (kW h)
D_{max}	maximum demand in all time horizons (kW)
D_e^t	electricity demand at tth time interval (kW)
D_c^t	cooling demand at tth time interval (kW)
ec_{AHP}^t	electricity consumption for operating an AHP at tth time interval (kW)
ec_{Pump1}^t	electricity consumption for operating Pump 1 at tth time interval (kW)
ec_{Pump2}^t	electricity consumption for operating Pump 2 at tth time interval (kW)
$ecoe^t$	electricity consumption for operating a battery and meeting electricity demand at tth time interval (kW)
ef_b	efficiency of charging/discharging of electricity (–)
ef_{TES}	efficiency of storing/releasing of thermal energy (–)
ep^t	price of electricity per kW h at tth time interval (yen/kW h)
$epoe^t$	price of electricity for operating a battery and meeting electricity demand at tth time interval (yen/h)
$epes^t$	price of electricity for operating an AHP and TES and meeting cooling demand at tth time interval (yen/h)
f_i^t	frequency of ith individual at tth time interval
f_{min}	minimum value of frequency (=0.0)
f_{max}	maximum value of frequency (=2.0)
$loss_{TES}$	loss of energy of TES per an hour (–)
$macd$	maximum amount of charging/discharging of electricity (kW)
$macr^t$	maximum amount of storing/releasing of thermal energy at tth time interval (kW)
$max\_P_{AHP}^t$	maximum power output of an AHP at tth time interval (kW)
$\overrightarrow{Mean^t}$	position vector of mean individual at tth time interval
n	population size
nd	number of dimensions
nc	number of children
P_{AHP}^t	power output of an AHP at tth time interval (kW)
pe	assumed SCOP based on primary energy (=0.77)
r_1, r_2	random number with uniformly distribution
R_b^t	rate of charging/discharging of electricity at tth time interval (–)
R_{TES}^t	rate of storing/releasing of thermal energy at tth time interval (–)
S_b^t	state of charge at tth time interval (kW h)
S_{TES}^t	Stored thermal energy at tth time interval (kW h)
t	time interval (=1 h)
$time$	time horizon in each calculation period (=30 h)
$\overrightarrow{v_i^t}$	ith velocity vector at tth time interval
w	coefficient of inertia of PSO
$\overrightarrow{x_i^t}$	position vector of ith individual at tth time interval of PSO and SLBA
$\overrightarrow{x_p^t}$	position vector of parent of DE
$\overrightarrow{x_i^{pare}}$	position vector of ith parent of GA
$\overrightarrow{x^{pare\_g}}$	position vector of all parents' center of gravity of GA
$\overrightarrow{x_i^{child}}$	position vector of ith child individual of GA
$\overrightarrow{xPbest_i^t}$	vector of the best position found by ith individual at tth time interval of PSO and SLBA
$\overrightarrow{xGbest^t}$	vector of the best position in all individuals at tth time interval of PSO and SLBA
$\overrightarrow{x_{d1}^t}, \overrightarrow{x_{d2}^t}$	position vector of differentiable individuals of DE
$\overrightarrow{x_{new}^{t+1}}$	position vector of a new individual of DE
$\overrightarrow{xWorst^t}$	vector of the worst position in all individuals at tth time interval of SLBA
ζ_i	random number of a uniformly distribution with mean 0 and variance $\sigma_\zeta^2 = 1/(nd+k)$

to solve a multi-objective problem of energy systems that include biomass energy. Fong et al. [7] applied a non-revisiting strategy to a genetic algorithm (GA) and particle swarm optimization (PSO) to minimize life cycle costs in centralized air-conditioning systems. Lee and Kung [8] used PSO to minimize life cycle costs by optimizing the capacity and volume of melted ice of the ice storage in an air-conditioning system. Moradi et al. [9] applied a hybrid method combining PSO with fuzzy linear programming to optimize heat production and electricity dispatch of CHP. Wang et al. [10] used a GA to optimize the capacity and operation of combined cooling, heating, and power (CCHP) in comparison to a separation production system.

Although these previous studies provided effective optimization methods, they dealt with energy systems without storage equipment. On the other hand, the number of studies that have considered storage equipment has increased in recent years [11–33]. Although there are many optimization methods, we can divide them into two categories, mathematical programming, such as MILP and dynamic programming (DP), and metaheuristic optimization or metaheuristics. MILP [11–20] and DP [21–23] are often used in previous studies, because those methods can always derive a theoretically optimal solution. However, their computation time is very long, when many decision variables and discrete points are included. In contrast, metaheuristics, such as neural networks [24], the bat algorithm (BA) [25], GAs [26], PSO [26–31], CS [32], and simulated annealing (SA) [33], first determine all variables at once, and then change each decision variable using a specific method to minimize (or maximize) an objective function. Thus, an optimal solution can be obtained fast, even if the problem is complex. Additionally, there are no limitations on the use of metaheuristics, in contrast to mathematical programming, which has such limitations as linearity, non-linearity, convexity, differentiability, and continuity. Therefore, metaheuristics have substantial versatility for optimizing nearly all functions. In this paper, we apply five metaheuristics to optimize an operating schedule of energy systems and compare the results with those obtained using DP. The metaheuristics used are GA, differential evolution (DE), CS, mutation-PSO (m-PSO), developed by the authors to improve the original PSO, and the self-adaptive learning bat algorithm (SLBA) because of their efficiency.

2. Materials and methods

2.1. Energy system and load profiles

2.1.1. Modeling energy systems

We considered a simple energy system consisting of a battery, an air-source heat pump (AHP), and TES, as shown in Fig. 1.

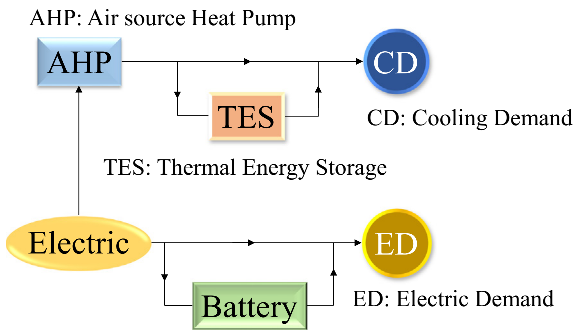

Fig. 1. Energy system.

Electric power for all equipment is supplied from an electric grid. Because a battery cannot sell electricity to the grid, there are three strategies for operating a battery while meeting electricity demand: from (1) grid to electricity demand, (2) grid to battery, and (3) battery to electricity demand. The charging/discharging efficiency of the battery was assumed to be 90% [34]. The self-discharge rate was not considered, because the amount of self-discharge of recent Li-ion batteries is low [35]. For operating an AHP and TES, we positioned two pumps, Pump 1 and Pump 2, as shown in Fig. 2, with four operating modes: (1) charging storage (Pump 1: ON, Pump 2: OFF), (2) meeting loads from storage-only (Pump 1: OFF, Pump 2: ON), (3) meeting loads from storage and direct AHP operation (Pump 1: ON, Pump 2: ON), and (4) meeting loads from direct AHP operation-only (Pump 1: ON, Pump 2: OFF). The inlet and outlet water temperatures of the AHP were fixed at 7 and 12 °C, respectively. An amount of chilled water was varied in relation to the machine load rate in order to consider partial load operation, because the capacity of each pump was fixed. The TES storing/releasing efficiency was assumed to be 80% [36], and the rate of self-loss energy was assumed to be 0.2% per hour. The characteristics of each piece of machinery are shown in Table 1. In general, maximum power output of an AHP depends on outdoor temperature and chilled water inlet temperature [37]. AHP capacity assumed to be 1000 kW, and the relation between machine load rate and power output rate is shown in Fig. 3.

2.1.2. Load profiles

We considered an office building in Tokyo with a total floor space of 16531.1 m^2 and two calculation periods as case studies. The warming-up period was from 12:00 a.m. on August 14 to 6:00 a.m. on August 15, and the analyzed period from 6:00 p.m. on August 14 to 12:00 a.m. on August 16. Time horizons and intervals were 30 h and 1 h, respectively. A portion of the analyzed period overlaps with a portion of the warming-up period, because we can recalculate the operating schedule of overlapped time horizons, even if the schedule cannot be optimized sufficiently. Electricity demand representing electricity consumption by lighting and PCs was determined using Computer-Aided Simulation for Cogeneration Assessment & Design III (CASCADE III), provided by the Society of Heating, Air-Conditioning and Sanitary Engineers of Japan (SHASE) [38]. Cooling demand was calculated using the New Heating, Air-conditioning, Sanitary Program/Air-conditioning Load (NewHASP/ACLD) [39]. Electricity demand, cooling demand, and outdoor temperature are shown in Fig. 4.

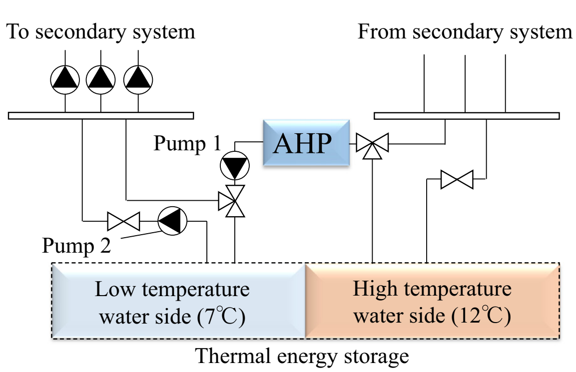

Fig. 2. Piping flow.

Table 1
Characteristics of each piece of machinery.

Air-source heat pump	Cooling capacity (kW)	1000
	Rate of energy consumption (kW)	278.6
Thermal energy storage	Thermal storage capacity (kW h)	3000
	Maximum permitted amount of storing/releasing of thermal energy during a time interval (kW)	Depending on power output from an AHP
	Efficiency of storing/releasing of thermal energy (–)	0.8
	Rate of heat loss (1/h)	0.002
Battery	Battery capacity (kW h)	500
	Maximum permitted amount of charging/discharging of electricity during a time interval (kW)	100
	Efficiency of charging/discharging of electricity (–)	0.9

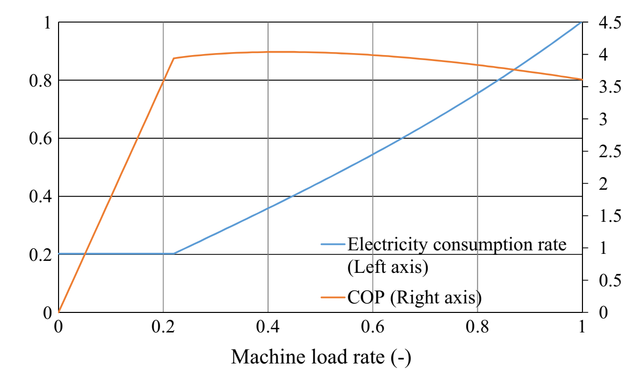

Fig. 3. Characteristics of an AHP.

2.1.3. The price of electricity

The price of electricity in each time interval varies with the total hourly electricity consumption. In this paper, we considered the merit order as a price decision method of simulating an imaginary electricity market. Thus, we adopt four assumptions to obtain the price of electricity per kW h: (1) the type of power plant, (2) the number of each type of power plant, (3) the composition rate of each power plant, and (4) the power generation costs of each type of power plant. First, the type of power plant was set to nuclear, liquid natural gas (LNG) fired, oil fired, coal fired, and hydroelectric, because those are the typical types. Second, the number of each type of power plant in the same order as mentioned above was set to 5, 16, 6, 10, and 8, respectively. The composition rate of each power plant was set to 27%, 40%, 17%, 2%, and 14%, respectively, as in [40]. It is important to consider the total capacity of all of the

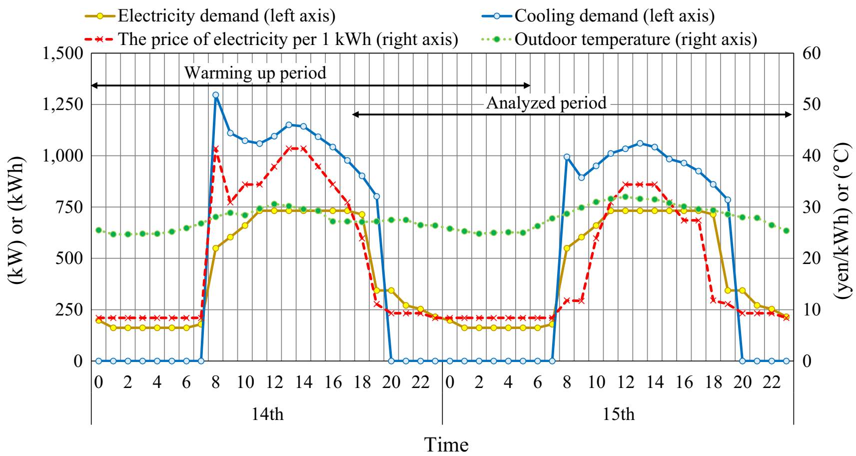

Fig. 4. Cooling and electricity demand and price of electricity.

plants in order to investigate a power shortage. Thus, we used Eq. (1) to obtain that total.

$$D_{max} = \max\{D_e^t + D_c^t \cdot pe\} \tag{1}$$

The assumed maximum electric consumption is 2226 kW at 8:00 a.m. in the warming-up period. Thus, the capacity of each type of power plant was set to 592.1, 899.3, 369.5, 55.7, and 309.4 kW, respectively. Finally, the power generation costs were taken from [41]. The merit order are shown in Fig. 5. In Fig. 4, 8:00 a.m., 1:00 p.m., and 2:00 p.m. in the warming-up period represent the peak times, because the price of electricity in the time intervals is 41.4 yen/kW h, the highest price, as shown in Fig. 5. The exchange rate of Japanese yen against US-$ is 1 yen = $0.009.

2.2. Problem formulation

In this paper, the aim of the optimization is to minimize operating costs for 30 h in each calculation period, as follows:

Objective function:

$$\min f = \sum_{t=1}^{time}(epoe^t + epes^t) \tag{5}$$

Constraints:

$$0 \leqslant S_b^t \leqslant C_b \tag{6}$$

$$0 \leqslant acd^t \leqslant macd \tag{7}$$

$$|acd^t \cdot ef_b| \leqslant D_e^t(acd^t < 0) \tag{8}$$

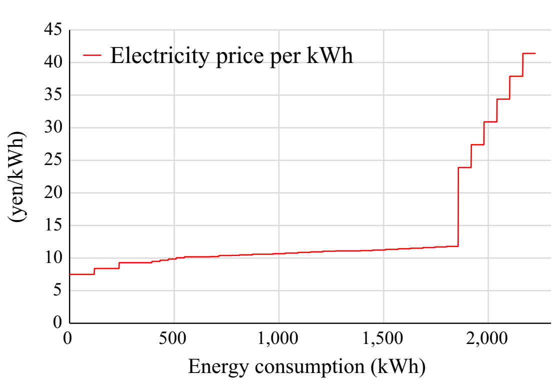

Fig. 5. Merit order of the price of electricity.

$$0 \leqslant S_{TES}^t \leqslant C_{TES} \tag{9}$$

$$0 \leqslant acr^t \leqslant macr^t \tag{10}$$

$$0 \leqslant P_{AHP}^t \leqslant max\_P_{AHP}^t \tag{11}$$

where

$$epoe^t = ep^t \cdot ecoe^t \tag{12}$$

$$ecoe^t = acd^t + D_e^t(acd^t \geqslant 0) \tag{13}$$

$$ecoe^t = acd^t \cdot ef_b + D_e^t(acd^t < 0) \tag{14}$$

$$epes^t = ep^t(ec_{AHP}^t + ec_{Pump1}^t + ec_{Pump2}^t) \tag{15}$$

$$S_b^{t+1} = S_b^t + acd^t \cdot ef_b(acd^t \geqslant 0) \tag{16}$$

$$S_b^{t+1} = S_b^t + acd^t(acd^t < 0) \tag{17}$$

$$acd^t = macd \cdot R_b^t \tag{18}$$

$$S_{TES}^{t+1} = S_{TES}^t(1 - loss_{TES}) + ef_{TES} \cdot acr^t(acr^t \geqslant 0) \tag{19}$$

$$S_{TES}^{t+1} = S_{TES}^t(1 - loss_{TES}) + acr^t(acr^t < 0) \tag{20}$$

$$acr^t = macr^t \cdot R_{TES}^t \tag{21}$$

$$macr^t = max\_P_{AHP}^t \tag{22}$$

$$P_{AHP}^t = acr^t + D_c^t(acr^t \geqslant 0) \tag{23}$$

$$P_{AHP}^t = acr^t \cdot ef_{TES} + D_c^t(acr^t < 0) \tag{24}$$

Decision variables:

R_b^t: rate of charging/discharging of electricity for 30 h
R_{TES}^t: rate of storing/releasing of thermal energy for 30 h

The decision variables are of two types, the rates of charging/discharging of electricity and the storing/releasing of thermal energy for 30 h. Thus, there are sixty decision variables in the optimization problem. The rate of power output of an AHP need not be included in the decision variables, because it is determined automatically by the relation between the amounts of storing/releasing of thermal energy and the cooling demand, using Eqs. (23) and (24).

2.3. Methodology

2.3.1. DP

The computational accuracy of metaheuristics is difficult to determine, because the use of random numbers can prevent these methods from deriving the optimal solution. Thus, we estimated the accuracy of the metaheuristics in comparison with the optimal solution obtained by DP. We set discrete points to 1%. There are two calculation methods in DP, backward programming and forward programming. Backward programming determines the optimal route, whereas forward programming determines optimal discrete points, as shown in Fig. 6.

2.3.2. GA

Although a GA generally involves binary variables, we applied a real-coded genetic algorithm (RCGA) with Real-coded Ensemble Crossover (REX) + Just Generation Gap (JGG) [42] to optimize continuous decision variables. RCGA with REX+JGG is more efficient than RCGA with Unimodal Normal Distribution Crossover (UNDX) + Minimal Generation Gap (MGG), which was already known to be an efficient method [43]. In the RCGA with REX+JGG procedure, first, the population size is generally assigned to n ($nc = 6nd \sim 22nd$). Second, $nd + k$ parents are selected randomly from the population of size n. Third, nc ($nc = 5nd \sim 10nd$) children are generated using the REX method with Eq. (25). Finally, nc children replace all selected parents. In this paper, we set k to 1.0 and nc to $6nd$. No mutation method is required in RCGA with REX+JGG.

$$\overrightarrow{x_i^{child}} = \overrightarrow{x^{pare\_g}} + \sum_{i=1}^{nd+k} \zeta_i \left(\overrightarrow{x_i^{pare}} - \overrightarrow{x^{pare\_g}} \right) \tag{25}$$

2.3.3. PSO

PSO, which imitates the collective behavior of birds and fish, is used frequently [7–9,26–31]. An individual of PSO has three types of vector, the current velocity vector, the best position vector for all particles, and the past best position vector for itself, as shown in Fig. 7. Each individual moves to optimize the objective function by using Eq. (26).

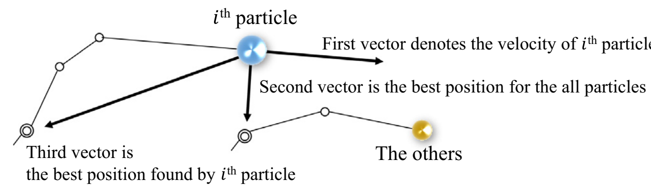

Fig. 7. Three vectors of PSO.

$$\overrightarrow{v_i^t} = w\overrightarrow{v_i^{t-1}} + c_1 r_1 \left(\overrightarrow{xPbest_i^t} - \overrightarrow{x_i^t} \right) + c_2 r_2 \left(\overrightarrow{xGbest^t} - \overrightarrow{x_i^t} \right) \tag{26}$$

PSO involves three parameters, w, c_1, and c_2, as shown in Eq. (26). Bergh and Engelbrecht [44] showed that PSO can have an advantage in finding the best solution, when Eq. (27) is specified.

$$w > \frac{1}{2}(c_1 + c_2) - 1 \tag{27}$$

In this paper, $w = 0.7298$ and $c_1 = c_2 = 1.49618$ were applied from [44]. Although PSO was reported as an efficient method in prior studies, it can become trapped in a local minimum of a multi-modal function. To improve the performance of PSO, Miranda [45] developed evolutionary self-adapting PSO (EPSO), which added an evolutionary method to classical PSO (c-PSO). In this paper, we develop m-PSO, which adds a mutation method to c-PSO. Although both EPSO and m-PSO are improved in terms of evolution, they still differ. In EPSO, a mutation method with a Gaussian distribution was added to the velocity calculation in Eq. (26). In contrast, in m-PSO, we adopted a mutation method with uniformly distributed positions of individuals by Eq. (28).

$$x_{i,j} = \begin{cases} \mathscr{U} & \text{if } rand \leqslant mrate \\ x_{i,j} & \text{if } rand > mrate \end{cases} \tag{28}$$

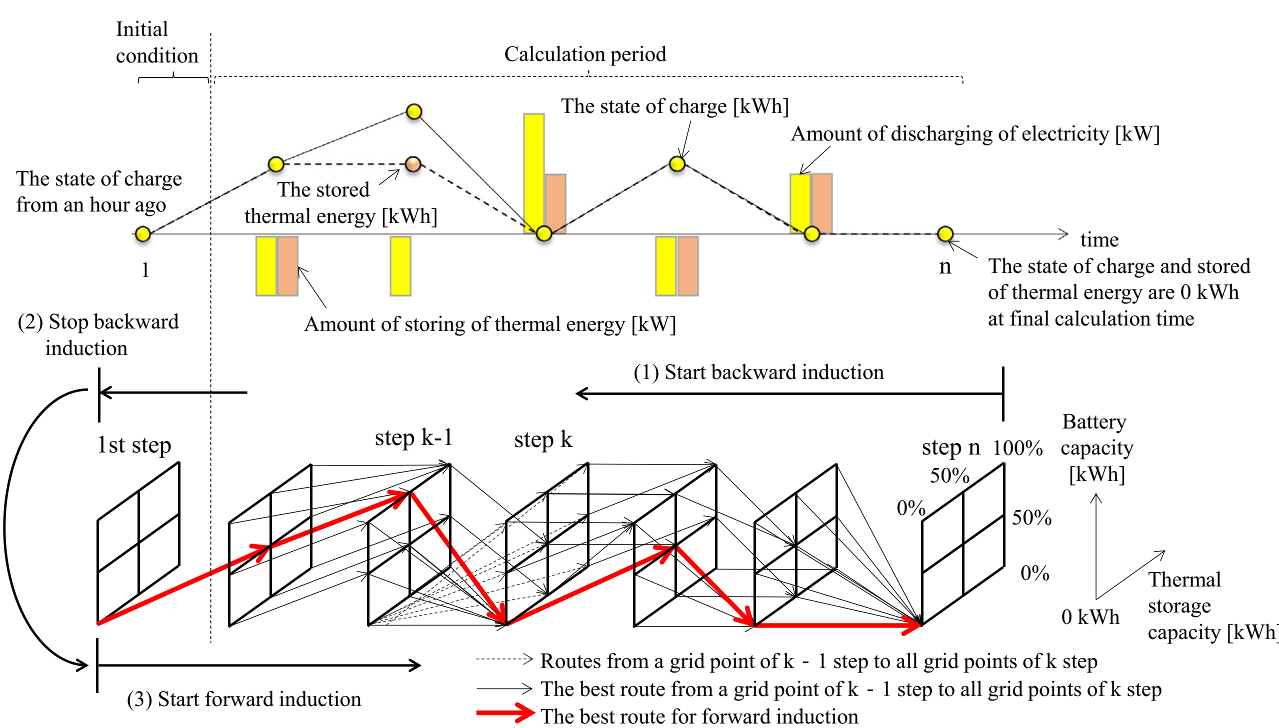

Fig. 6. Backward and forward programming in DP.

where $x_{i,j}$ denotes each element of individual's vector, $\mathcal{U} \in [Lb, Ub]$ denotes an uniformly distributed number that range is lower bound Lb and upper bound Ub of each decision variables. These values are in $[-1, 1]$ because all decision variables are normalized in $[-1, 1]$ in this paper. Thus, each element of an individual's vector is changed by *mrate* (mutation rate) with the specified probability after Eq. (26). We set it to 5%. Although m-PSO is a simpler improvement than EPSO, it showed a substantial advantage over both c-PSO and EPSO. In Section 3.3, the differences among c-PSO, EPSO, and m-PSO are discussed.

2.3.4. DE

An individual in DE [46,47], which has been reported to be better than GA [48], has the same vectors as in PSO. In the DE procedure, first, $\overrightarrow{x_p^t}$ is selected randomly as parent vector. Second, $\overrightarrow{x_{d1}^t}$ and $\overrightarrow{x_{d2}^t}$ are selected as differential vectors to create a new individual ($\overrightarrow{x_{new}^{t+1}}$). Third, a direction vector is created using Eq. (29), as shown in Fig. 8.

$$\overrightarrow{v_i^{t+1}} = \overrightarrow{x_p^t} + M\left(\overrightarrow{x_{d1}^t} - \overrightarrow{x_{d2}^t}\right) \tag{29}$$

Finally, $\overrightarrow{v_i^{t+1}}$ is replaced with $\overrightarrow{x_p^t}$, using Eq. (30).

$$\begin{aligned} \overrightarrow{x_{new}^{t+1}} &= \overrightarrow{v_i^{t+1}} (\text{when rand} < crossover\,rate) \\ \overrightarrow{x_{new}^{t+1}} &= \overrightarrow{x_p^t} (\text{when rand} > crossover\,rate) \end{aligned} \tag{30}$$

where rand $\in [0,1]$ denotes a uniformly distributed random number, and the crossover rate is set to 0.9.

2.3.5. CS

CS was developed by Yang and Deb [49] in 2009. They showed that CS is superior to GA and c-PSO, and Civicioglu and Besdok [50] showed that CS is superior to c-PSO and is as efficient as DE. CS was used in [4,5,32,51–53]. The CS algorithm is based on the brood parasitism of a cuckoo. Initial individuals are selected randomly. Lévy flight is performed when generating a new individual in the next iteration. The best individual at each iteration carries over to the next iteration. Brood parasitic behavior is formulated in CS in terms of a single parameter, *pa*, representing the probability of eggs being discovered by the host bird.

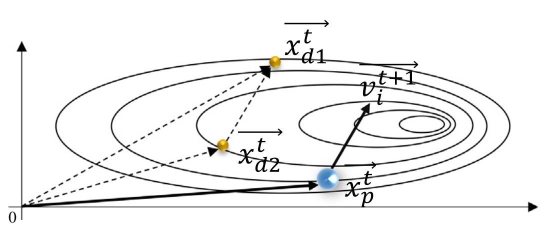

Fig. 8. Creating a direction vector $\overrightarrow{v_i^{t+1}}$.

2.3.6. SLBA

Bahman and Azizipanah-Abarghooee [25] developed SLBA, which combined BA [54] and self-learning [55], in 2010. BA imitates literally the gulping behavior of a bat. The difference between BA and SLBA is in how the velocity is calculated. In BA, only Eq. (32) is used to create a new individual, as follows:

$$f_i^{t+1} = f_{min} + (f_{max} - f_{min}) \cdot rand \tag{31}$$

$$\overrightarrow{v_l^{t+1}} = \overrightarrow{v_l^t} + \left(\overrightarrow{x_l^t} - \overrightarrow{xGbest^t}\right) \cdot f_i^{t+1} \tag{32}$$

$$\overrightarrow{x_{new}^{t+1}} = \overrightarrow{x_l^t} + \overrightarrow{v_l^{t+1}} \tag{33}$$

where f_{max} and f_{min} are set to two and zero, respectively. In contrast, four velocity-updating strategies are used in SLBA, as follows [25].

Velocity updating strategy 1:

$$\begin{aligned} \overrightarrow{v_l^{t+1}} = \overrightarrow{v_l^t} &+ (0.3 \cdot f_i^{t+1} + 0.4)\left(\overrightarrow{xGbest^t} - \overrightarrow{x_l^t}\right) \\ &+ (0.6 \cdot rand + 0.4)\left(\overrightarrow{xGbest^t} - \overrightarrow{xWorst^t}\right) \end{aligned} \tag{34}$$

Velocity updating strategy 2:

$$\overrightarrow{v_l^{t+1}} = \overrightarrow{v_l^t} + \left\{\overrightarrow{x_{p1}^t} + rand\left(\overrightarrow{x_{p2}^t} - \overrightarrow{x_{p3}^t}\right)\right\} \tag{35}$$

Velocity updating strategy 3:

$$\overrightarrow{v_l^{t+1}} = rand \cdot \overrightarrow{v_l^t} + (0.3 \cdot rand + 0.2) f_i^{t+1}\left(\overrightarrow{xGbest^t} - \overrightarrow{x_l^t}\right) \tag{36}$$

Velocity updating strategy 4:

$$\begin{aligned} \overrightarrow{v_l^{t+1}} &= rand \cdot \overrightarrow{v_l^t} + 0.5(0.3 \cdot rand + 0.2) \\ &\times f_i^t\left(\overrightarrow{xGbest^t} - round(1 + rand) \cdot \overrightarrow{Mean^t}\right) \end{aligned} \tag{37}$$

The strategy to be used at each step is selected using a roulette wheel mechanism (RWM).

2.3.7. Individual modeling and constraint handling

An individual has sixty decision variables, as shown in Fig. 9. Positive and negative numbers represent charging/storing and discharging/releasing modes, respectively.

There are some constraints in this paper, as mentioned in connection with Eqs. (6)–(11). Constraint handling is important when applying metaheuristics to a problem with constraints that was developed originally without constraints. The constraint handling procedure is shown as follows.

First, sixty decision variables are determined randomly at once. Then, we check each constraint for each time interval.

Day	14 (Mon)				15 (Tue)			14 (Mon)				15 (Tue)		
Time	6 p.m.	7	8		3 a.m.	4	5	6 p.m.	7	8		3 a.m.	4	5
	0.51	0.83	0.67	……	0.60	0.60	0.45	0.51	0.83	-0.52	……	0.51	0.83	-0.52
Data num.	1	2	3		28	29	30	31	32	33		58	59	60

Data number from 1 to 30, rates of charging or discharging of electricity

Data number from 31 to 60, rates of storing or releasing of thermal energy

Fig. 9. Coding an individual in all of the metaheuristics.

(1) Checking constraints, Eq. (6).

If S_b^t exceeds C_b, acd^t is decreased to set S_b^t equal to C_b, using Eq. (16). If S_b^t is less than zero, acd^t is revised to set S_b^t equal to zero, using Eq. (17).

(2) Checking constraints, Eq. (8).

If acd^t in discharging mode is greater than D_e^t, acd^t is set equal to the value of D_e^t, because the inability to sell electricity from the battery to the grid renders excessive discharging useless.

(3) Checking constraints, Eq. (9) right side.

If S_{TES}^t exceeds C_{TES}, acr^t is decreased to set S_{TES}^t equal to C_{TES}, using Eq. (19) and the power output of an AHP is also revised, using Eq. (23).

(4) Checking constraints, Eq. (11) right side.

If P_{AHP}^t is greater than $max\_P_{AHP}^t$, when TES is in the charging mode, acr^t is decreased to set P_{AHP}^t equal to $max\_P_{AHP}^t$. On the other hand, if P_{AHP}^t is greater than $max\_P_{AHP}^t$, when TES is in the discharging mode, acr^t, a negative number, must be increased to meet the cooling demand, because there is an energy shortage, even if an AHP generates maximum cooling thermal energy.

(5) Checking constraints, Eq. (9) left side.

If S_{TES}^{t+1} is less than zero, acr^t, a negative number, is decreased to set S_{TES}^{t+1} equal to zero, using Eq. (20), and P_{AHP}^t is increased, using Eq. (24), because the amount of discharging is reduced.

(6) Checking constraints, Eq. (11) left side.

If P_{AHP}^t is negative, acr^t, which is in discharging mode, is increased using Eq. (20), because acr^t is greater than D_c^t.

(7) Re-checking constraints, Eq. (11) right side.

We use a death penalty method when further change is needed in this phase.

Installing TES reduces the capacity of an AHP, which contributes to reducing energy consumption and costs. However, it, causes an energy shortage, if there is insufficient energy stored in TES at peak time intervals. Further change indicates that there is an energy shortage in the time interval. We must revise the amount of charging in the previous time interval to meet the cooling demand in the current interval. That indicates that high computation costs are required. Consequently, we use a death penalty method in the interval, when re-checking is required.

2.3.8. Stopping criteria

Stopping criteria must be specified, because metaheuristics use iteration to find the best solution. The specified number of generations is generally used as the stopping criterion. However, we used three types of stopping criteria to clarify their function, tolerance (Stopping criterion (SC)-I), number of generations (SC-II), and computation time (SC-III). For SC-I, the iteration is stopped, when the tolerance between the results of DP and the metaheuristic is less than 1%, 0.5%, or 0.1%. We analyzed the computational accuracy of each metaheuristic. We set maximum computation time to an hour, because metaheuristics might not converge within the tolerance, when especially the tolerance is 0.1%. For SC-II, we set the maximum number of iterations and the number of individuals to 10,000 and 100, respectively. We analyzed the convergence and calculation speeds of each metaheuristic. For SC-III, we set the computation time to 5 min, 10 min, and 30 min, respectively. We analyzed the advantages of each metaheuristic for use in practical optimization, for instance, energy system management for actual buildings. For all stopping criteria, we performed the same calculation 30 times, because the semi-optimal solution obtained using metaheuristics varies every time, as a result of using random numbers. The computer used during all optimizations had the following characteristics: Windows 7 64 bit, 3.40 GHz Core i7-4700 CPU, and 32 GB RAM. All optimizations were performed using MATLAB R2014a.

3. Results and discussion

3.1. The theoretically optimal solution

The theoretically optimal solution from DP is 305,335 yen/30 h in the analyzed period, and its computation time is 5 h, 1 min, 54 s. This value, 305,335 yen/30 h, is taken as the standard value, when the metaheuristics are compared with DP.

3.2. Sensitivity analysis of the CS pa parameter

We conducted a sensitivity analysis of the CS parameter, *pa*, to determine suitable values for it. Although Yang and Deb [49] set *pa* to 0.25, they recommended using either 0.25 or 0.75, when applying CS to problems with small or huge domains [56], respectively. Then, five values were used with SC-II, 0.75, 0.8, 0.85, 0.9, and 0.95, with the results shown in Fig. 10. Although the minimum value when *pa* equals 0.85 is the smallest, a *pa* value of 0.9 is the most suitable in terms of average value and convergence. Consequently, we use a *pa* value of 0.9 after this section.

3.3. The advantage of m-PSO

The performance of c-PSO, EPSO, and m-PSO are shown in Fig. 11. Although adding a mutation method to c-PSO is simple, m-PSO is more accurate than c-PSO and EPSO. Therefore, mutation for an individual's position has an advantage in finding the optimal solution for our metaheuristics.

3.4. Results of SC-I (Stopping criterion of tolerance)

The results of SC-I, which includes three kinds of tolerance criteria, are shown in Fig. 12: 0.1%, 0.5%, and 0.1%. m-PSO converges faster than the others, because m-PSO is the fastest with 1% and 0.5% tolerance. DE is the second fastest with 0.5% tolerance.

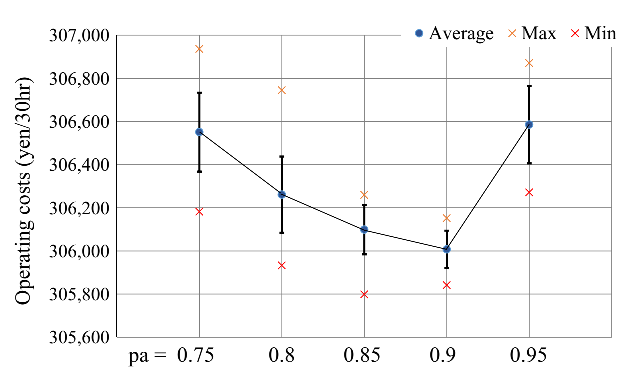

Fig. 10. Results of sensitivity analysis of *pa*.

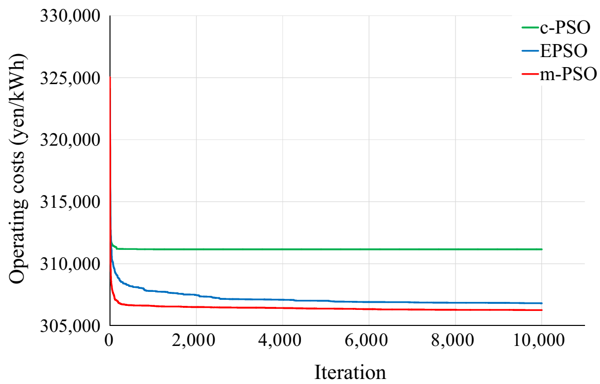

Fig. 11. Comparison of c-PSO, EPSO, and m-PSO.

However, DE never converges with 0.1% tolerance. Although SLBA is the second fastest with 1% tolerance, SLBA converges with a 73% probability with 0.5% tolerance and never converges with 0.1% tolerance. GA never converges with 0.5% and 0.1%, and its computational time is longest of all of the methods with 1% tolerance. Although CS is not faster with 1% and 0.5% tolerance, the success rate of CS with 0.1% is highest, 86%, all of the methods. Therefore, we can conclude that CS is the most accurate method.

3.5. Results of SC-II (Stopping criteria of a specific number of generations)

We can see the convergence and computational speed of each metaheuristic method in Fig. 13. Five lines represent the average performance of 30 repetitions. The order of convergence speed is m-PSO, SLBA, DE, CS, and GA. This result is associated with the results of SC-I, for which m-PSO is the fastest for 1% and 0.5% tolerance. In terms of computational speed, the order is SLBA, m-PSO, CS, DE, and GA. The algorithm of SLBA is the same as that of m-PSO in terms of using vectors and an equation of moving to the next position. Thus, SLBA and m-PSO are faster than the other methods as a consequence of the simplicity of these algorithms.

3.6. Results of SC-III (Stopping criterion of computation time)

CS is the most accurate in all cases of SC-III, as shown in Fig. 14: computation times of 5 min, 10 min, and 30 min. In terms of operating cost minimization, the order of the five methods is CS, m-PSO, DE, SLBA, and GA. The minimum value of CS is close to 0.062%, with a theoretically optimal solution for DP of 30 min. For SC-III, CS is the most accurate, because CS converges throughout its long iterations, the same results as for SC-II, because it is important not to become trapped in a local minimum.

3.7. Optimal operation schedule

3.7.1. Optimal operation schedule of battery

The optimal operation schedules of a battery are shown in Fig. 15. The upper, middle, and lower graphs show the results of DP, m-PSO, and CS, respectively. The discharging operations are conducted when the price of electricity (red straight line) is low in the three methods. In the theoretical optimal solution by DP, Fig. 15(a), the amount of charging electricity are alternately 94 to 100 kW at 3 a.m. to 7 a.m. during the warming up period (blue bar and line charts) and analyzed period (green bar and line charts). Each amount is not 100 kW because a battery causes loss of electricity when it charges or discharges electricity. On the other

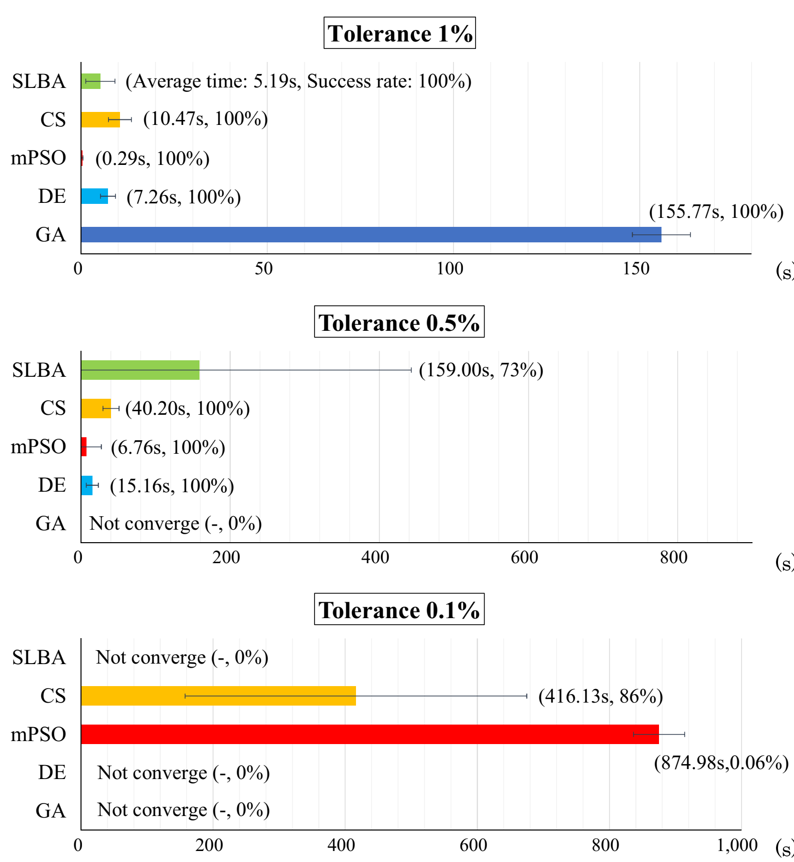

Fig. 12. Convergence speeds of five methods.

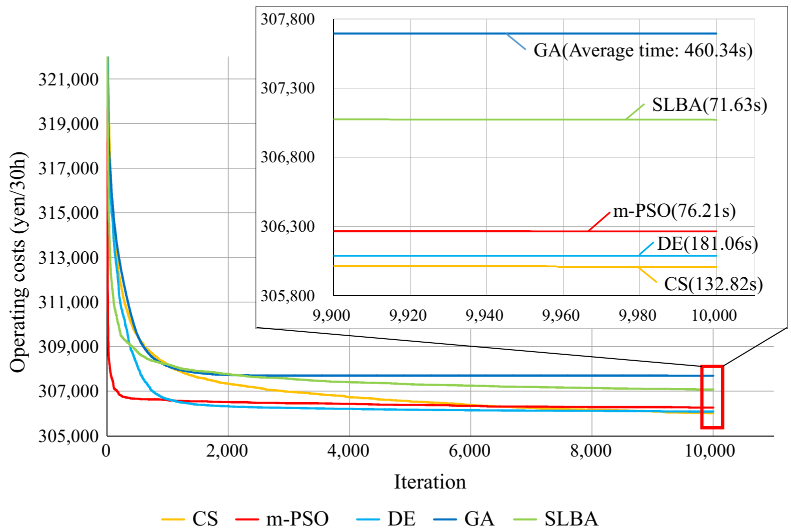

Fig. 13. Performance and computation times of five methods.

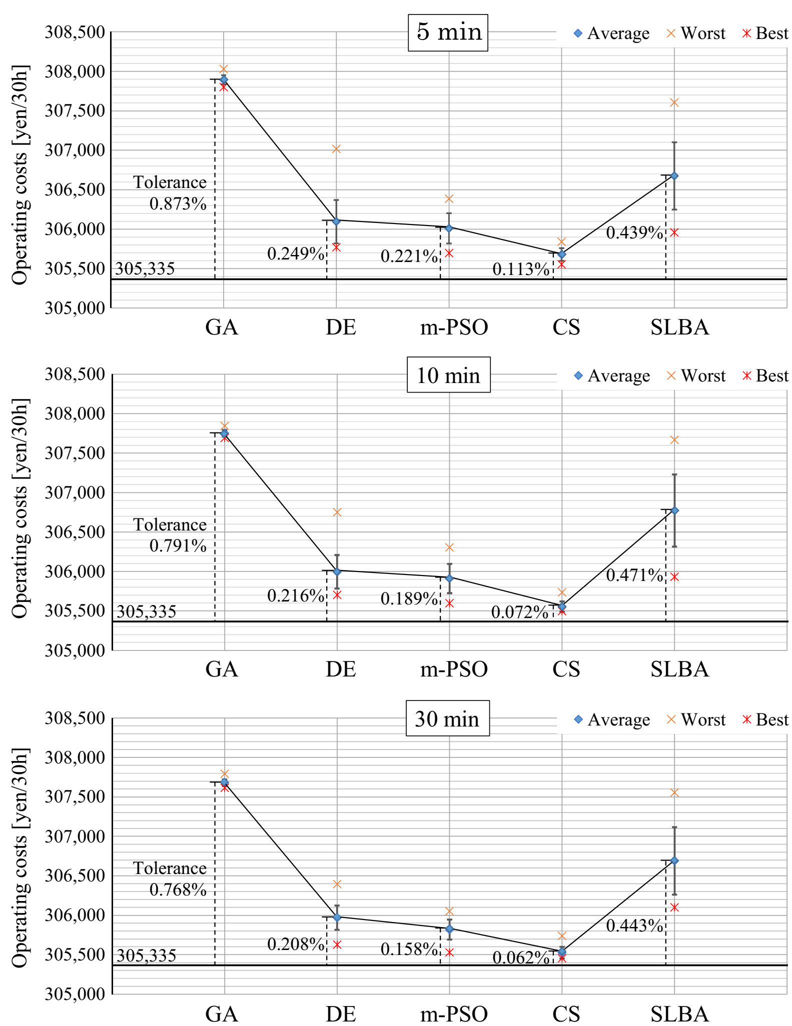

Fig. 14. Computational accuracy of five methods.

hand, in the results of m-PSO and CS, Fig. 15(b) and (c), charging operations are conducted with full charging at 100 kW at night, with the amount of decreasing to meet capacity constraints during other time intervals. For example, in Fig. 15(c), the battery charges electricity of 56.5 kW at 5 a.m., because the remaining battery is already 449 kW at 4 a.m., during the warming up period.

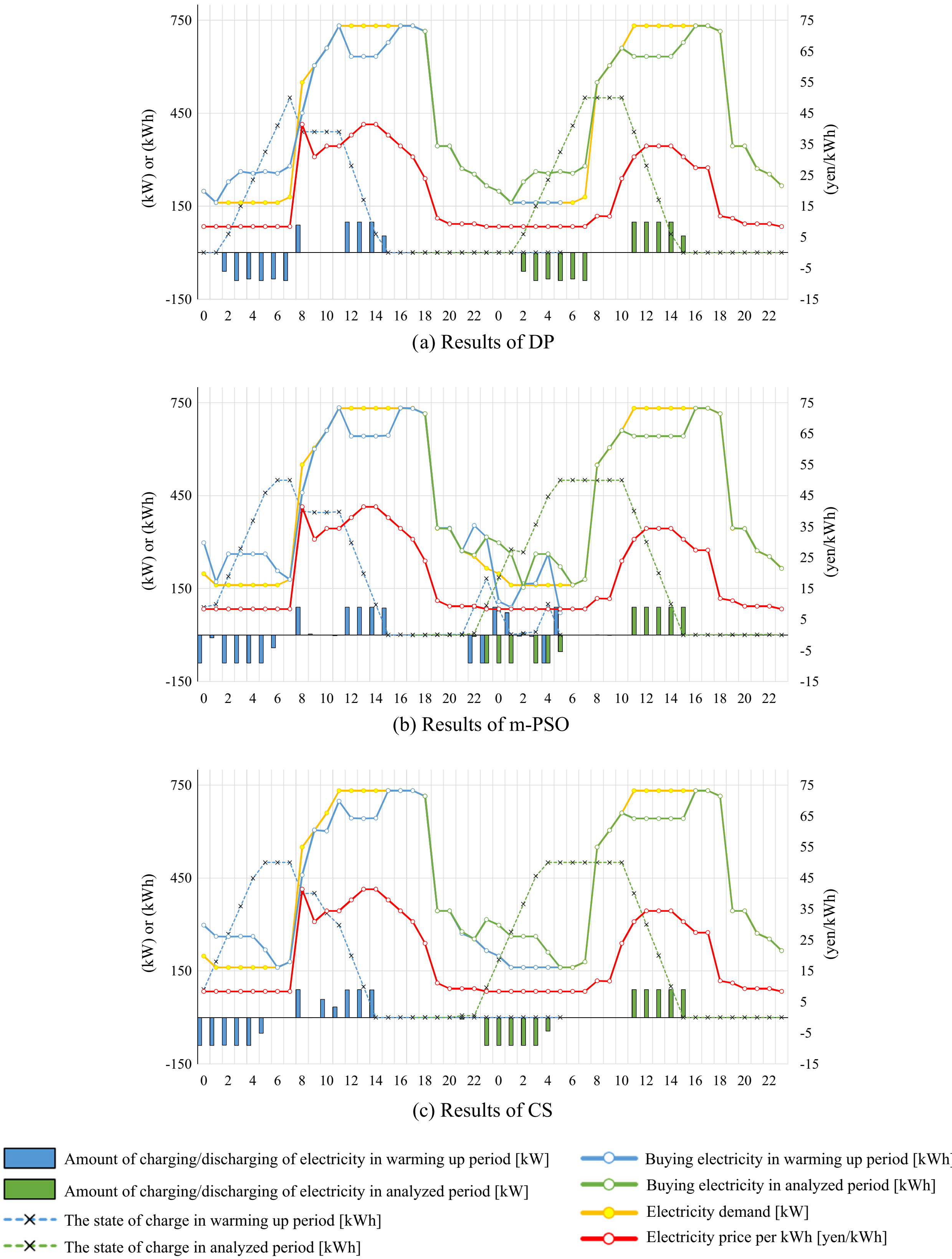

Fig. 15. Optimal operating schedule of a battery.

Although the operating cost is the same for DP and CS, the operation of CS is simple and straightforward for practical energy system engineers, because the amounts of charging electricity are constant. In Fig. 15(b), the results of m-PSO, the charging/discharging operation is conducted between 10 p.m. and 5 a.m. in overlapped time intervals, because finding the solutions of m-PSO is not enough for convergence in the warming up period. However, the discharging operation at 12, 1, and 5 a.m. in warming up period

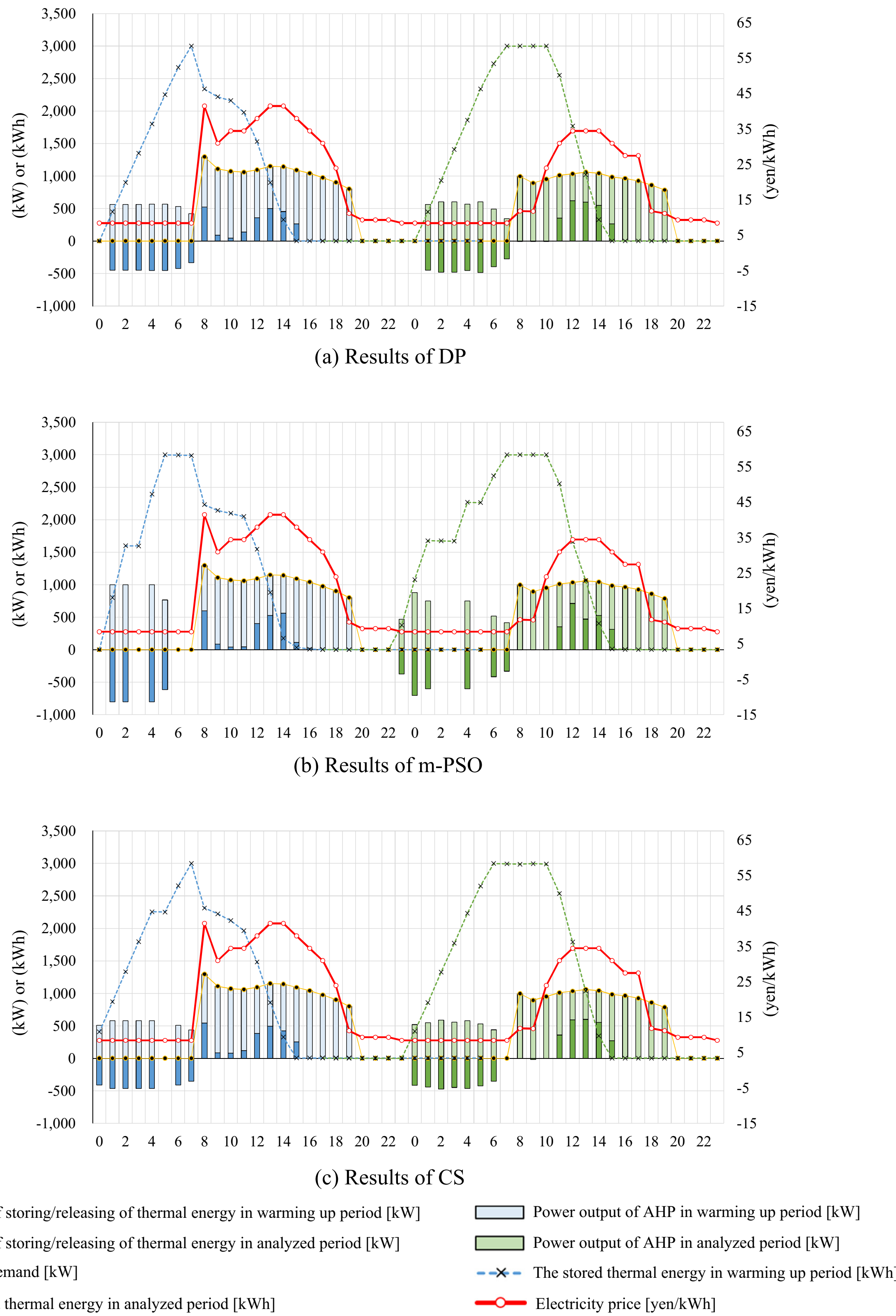

Fig. 16. Optimal operating schedule of an AHP and TES.

vanished in the analyzed period (green bar charts) because of the recalculation of optimization. In terms of discharging, each of the three optimization methods is conducted according to similar operation schedules, namely discharging is conducted when the price of electricity is high during the day in the warming up and analyzed periods.

3.7.2. Optimal operation schedule of TES and AHP

The optimal operation schedules of TES and AHP are shown in Fig. 16. In the results of DP and CS, Fig. 16(a) and (c), although the capacity of AHP is 1000–1100 kW depending on outdoor temperature, TES stores thermal energy around 450–500 kW at night time in both periods. The reason is that half power outputs of

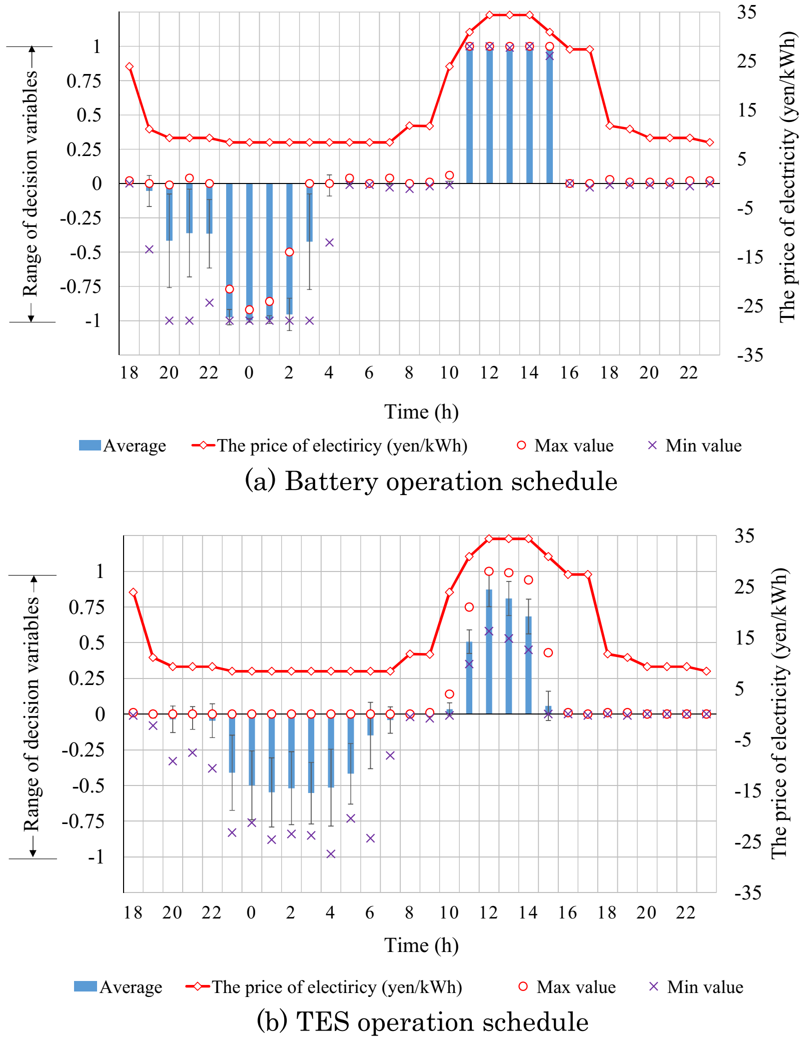

Fig. 17. Operation schedules dispersion.

the capacity are most efficient to generate cooling heat to AHP as shown Fig. 3. On the other hand, AHP generates cooling heat of 600–800 kW in the results of m-PSO as shown in Fig. 16(b). Thus, the results of m-PSO are inferior to DP and CS. In terms of releasing thermal energy, the results of DP and CS are very similar. When the price of electricity is high, for instance at 8 a.m., 1 p.m., and 2 p.m. during the warming up period and at noon, 1 p.m., and 2 p.m. in the analyzed period, the operation is conducted. Moreover, we can find that AHP power outputs are 410–500 kW during those time intervals because of the above mentioned reasons. Thus, DP and CS can find a suitable amount of releasing thermal energy using the relation between the remaining TES and the power generation efficiency of AHP.

3.7.3. Dispersion of optimal operation schedules

In Fig. 17, dispersions of the optimal operation schedules of battery and TES after 30 CS calculations of CS in the analyzed period are shown. Positive numbers on the left axis of each figure indicate the range of decision variables specifying the rates of discharging/releasing by battery and TES. Negative numbers indicate charging/storing. Focusing on the battery, average rates of discharging (blue bar chart) are nearly 1.0 from 11 a.m. to 3 p.m., when the price of electricity (red straight line) is high. There are some dispersions in charging at night, although full charging rates are conducted from 11 p.m. to 1 a.m. These dispersions create the tolerance (0.22%) between the results of DP and CS. Focusing on TES in Fig. 17(b), the average rates of storing/releasing (blue bar) are suitable for minimizing the operational cost, because when the price of electricity is low, storing is conducted, and when it is high, releasing is conducted. Moreover, at night, the average rates are 0.4–0.55 to operate the AHP efficiently.

4. Conclusion

We used a metaheuristic method to optimize operating schedules of an energy system that includes storage equipment, such as a battery and TES, in terms of minimization of operating costs. Metaheuristics have the advantage of computational speed and applied versatility over mathematical programming, which was often used in previous studies. We showed the following through this study:

(1) The most suitable value of the CS parameter, *pa*, was estimated to be 0.9, with 0.75 being recommended for huge problems [56].

(2) We developed m-PSO, which improved on c-PSO by adding a mutation method for individual position and was superior to c-PSO and EPSO, which uses a mutation method for velocity determination.
(3) The computational and convergence speeds of m-PSO were the highest. For SC-I, m-PSO can converge within 1% tolerance with a theoretically optimal solution of DP 62,068 times as fast as DP. For SC-II, we found that m-PSO can converge within 0.304% 23 times as fast as DP.
(4) In terms of computational accuracy, CS is the most accurate. For SC-II, CS converged within 0.22% 135 times as fast as DP. For SC-III, CS converged within 0.062% ten times as fast as DP.
(5) A heat source machine (AHP) that had nonlinear COP characteristic was able to work with its highly efficient load rate (0.4–0.55). Therefore, the proposed methods were able to solve the nonlinear problem quickly, while maintaining the computational accuracy.

Therefore, metaheuristics, particularly m-PSO and CS, have a substantial advantage over DP for optimizing the operating schedule of an energy system that includes a battery and TES.

References

[1] International Energy Agency (IEA). World Energy Outlook 2013 Chapter 6 Renewable Energy Outlook, 2013. p. 197–232.
[2] Toshiba Corporation. Smart Grid Progress in Japan & US, US-Japan Renewable Energy Policy Business Roundtable, 2012. p. 10.
[3] Omu A, Choudhary R, Boies A. Distributed energy resource system optimisation using mixed integer linear programming. Energy Policy 2013;61:249–66.
[4] Basu M, Chowdhury A. Cuckoo search algorithm for economic dispatch. Energy 2013;60:99–108.
[5] Chandrasekaran K, Simon SP. Multi-objective scheduling problem: hybrid approach using fuzzy assisted cuckoo search algorithm. Swarm Evol. Comput. 2012;5:1–16.
[6] Fazlollahi S, Marechal F. Multi-objective, multi-period optimization of biomass conversion technologies using evolutionary algorithms and mixed integer linear programming (MILP). Appl Therm Eng 2013;50:1504–13.
[7] Fong KF, Yuen SY, Chow CK, Leung SW. Energy management and design of centralized air-conditioning systems through the non-revisiting strategy for heuristic optimization methods. Appl Energy 2010;87:3494–506.
[8] Lee WS, Kung CK. Optimization of heat pump system in indoor swimming pool using particle swarm algorithm. Appl Therm Eng 2008;28:1647–53.
[9] Moradi MH, Hajinazari M, Jamasb S, Paripour M. An energy management system (EMS) strategy for combined heat and power (CHP) systems based on a hybrid optimization method employing fuzzy programming. Energy 2013;49:86–101.
[10] Wang JJ, Jing YY, Zhang CF. Optimization of capacity and operation for CCHP system by genetic algorithm. Appl Energy 2010;87:1325–35.
[11] Ashouri A, Fux SS, Benz MJ, Guzzella L. Optimal design and operation of building services using mixed-integer linear programming techniques. Energy 2013;59:365–76.
[12] Buoro D, Pinamonti P, Reini M. Optimization of a Distributed Cogeneration System with solar district heating. Appl Energy 2014;124:298–308.
[13] Dai R, Mesbahi M. Optimal power generation and load management for off-grid hybrid power systems with renewable sources via mixed-integer programming. Energy Convers Manage 2013;73:234–44.
[14] Ikegami T, Iwafune Y, Ogimoto K. Optimal Demand Controls for a Heat Pump Water Heater under Different Objective Functions, IEEE International Conference on Power System Technology (POWERCON), 2012. p. 1–6.
[15] Vetterli J, Benz M. Cost-optimal design of an ice-storage cooling system using mixed-integer linear programming techniques under various electricity tariff schemes. Energy Build 2012;49:226–34.
[16] Wakui T, Kinoshita T, Yokoyama R. A mixed-integer linear programming approach for cogeneration-based residential energy supply networks with power and heat interchanges. Energy 2014;68:29–46.
[17] Wakui T, Wada N, Yokohama R. Energy-saving effect of a residential polymer electrolyte fuel cell cogeneration system combined with a plug-in hybrid electric vehicle. Energy Convers Manage 2014;77:40–51.
[18] Wakui T, Yokoyama R. Optimal sizing of residential gas engine cogeneration system for power interchange operation from energy-saving viewpoint. Energy 2011;36:3816–24.
[19] Wakui T, Yokoyama R. Optimal structural design of residential cogeneration systems in consideration of their operating restrictions. Energy 2014;64:719–33.
[20] Yun H, Li W. Optimization and analysis of distributed energy system with energy storage device. Energy Procedia 2011;12:958–65.
[21] Facci AL, Andreassi L, Ubertini Stefano, Sciubba Enrico. Analysis of the influence of thermal energy storage on the optimal management of a trigeneration plant. Energy Procedia 2014;45:1295–304.
[22] Chen HJ, Wang DWP, Chen SL. Optimization of an ice-storage air conditioning system using dynamic programming method. Appl Therm Eng 2005;25:461–72.
[23] Li J, Danzer MA. Optimal charge control strategies for stationary photovoltaic battery systems. J Power Sources 2014;258:365–73.
[24] Abbey C, Strunz K, Joós G. A knowledge-based approach for control of two-level energy storage for wind energy systems. IEEE Trans Energy Convers 2009;24–2:539–47.
[25] Bahmani-Firouzi B, Azizipanah-Abarghooee R. Optimal sizing of battery energy storage for micro-grid operation management using a new improved bat algorithm. Electrical Power Energy Systems 2014;56:42–54.
[26] Daud MZ, Mohamed A, Ibrahim AA, Hannan MA. Heuristic optimization of state-of-charge feedback controller parameters for output power dispatch of hybrid photovoltaic/battery energy storage system. Measurement 2014;49:15–25.
[27] Baziar A, Kavousi-Fard A. Considering uncertainty in the optimal energy management of renewable micro-grids including storage devices. Renewable Energy 2013;59:158–66.
[28] Lee WS, Chen YT, Wu TH. Optimization for ice-storage air-conditioning system using particle swarm algorithm. Appl Energy 2009;86:1589–95.
[29] Moghaddam AA, Seifi A, Niknam T, Pahlavani MRA. Multi-objective operation management of a renewable MG (micro-grid) with back-up micro-turbine/fuel cell/battery hybrid power source. Energy 2011;36:6490–507.
[30] Pedrasa MAA, Spooner TD, MacGill IF. Coordinated scheduling of residential distributed energy resources to optimize smart home energy services. IEEE Transactions Smart Grid 2010;1–2:134–43.
[31] Wang JJ, Zhai ZJ, Jing Y, Zhang C. Particle swarm optimization for redundant building cooling heating and power system. Appl Energy 2010;87:3668–79.
[32] Berrazouane S, Mohammedi K. Parameter optimization via cuckoo optimization algorithm of fuzzy controller for energy management of a hybrid power system. Energy Convers Manage 2014;78:652–60.
[33] Ekren O, Ekren BY. Size optimization of a PV/wind hybrid energy conversion system with battery storage using simulated annealing. Appl Energy 2010;87:592–8.
[34] Valøen LO, Shoesmith MI. The Effect of PHEV and HEV duty cycles on battery and battery pack performance. In: Proc. Plug-in Highway Electric Vehicle Conference, 2007. p. 1–9.
[35] BatteryPlusForLife. Battery Technology Comparison, 2014. <http://batteryplusforlife.com/research.html>.
[36] Dincer I, Rosen M. Thermal energy storage: systems and applications. Wiley; 2002.
[37] Ministry of Land, Infrastructure, Transport and Tourism, LCEM tool ver.3.10, 2014.
[38] The Society of Heating, Air-Conditioning and Sanitary Engineers of Japan (SHASE). Computer Aided Simulation for Cogeneration Assessment & Design III, Maruzen Publishing, 2003.
[39] The Society of Heating, Air-Conditioning and Sanitary Engineers of Japan (SHASE). NewHASP/ACLD Ver. 20091117, Japanese Association of Building Mechanical and Electrical Engineers, 2009.
[40] TEPCO. ILLUSTRATED 2013, TEPCO, 2013.
[41] Energy and Environment Council. Final Report, 2011 (in Japanese).
[42] Takahashi T, Kawai K, Nakai H, Ema Y. Development of the automatic modeling system for reaction mechanisms using REX+JGG. Phys Procedia 2013;46:239–47.
[43] Ono I, Kobayashi S, Yoshida K. Optimal lens design by real-coded genetic algorithms using UNDX. Comput Methods Appl Mech Eng 2000;186:483–97.
[44] Van den Bergh F, Engelbrecht AP. A study of particle swarm optimization particle trajectories. Inf Sci 2006;176:937–71.
[45] Miranda V, Fonseca N. EPSO – Evolutionary Particle Swarm Optimization, a New Algorithm with Applications in Power Systems, Transmission and Distribution Conference and Exhibition 2002: Asia Pacific. IEEE/PES (Volume: 2) 2002. p. 745–50.
[46] Abou El Ela AA, Abido MA, Spea SR. Optimal power flow using differential evolution algorithm. Electric Power Syst Res 2010;80:878–85.
[47] Mallipeddi R, Jeyadevi S, Suganthan PN, Baskar S. Efficient constraint handling for optimal reactive power dispatch problems. Swarm Evol Comput 2012;5:28–36.
[48] Karaboğa D, Ökdem S. A simple and global optimization algorithm for engineering problems: differential evolution algorithm. Turk J Elec Eng 2004;12–1:53–60.
[49] Yang XS, Deb S. Cuckoo Search via Lévy Flights, Proc. of World Congress on Nature & Biologically Inspired Computing (NaBIC 2009). IEEE Publications, 2009. p. 210–14.
[50] Civicioglu P, Besdok E. A conceptual comparison of the Cuckoo-search, particle swarm optimization, differential evolution and artificial bee colony algorithms. Artif Intell Rev 2013;39–4:315–46.
[51] Piechocki J, Ambroziak D, Palkowski A, Redlarski G. Use of Modified Cuckoo Search algorithm in the design process of integrated power systems for modern and energy self-sufficient farms. Appl Energy 2014;114:901–8.
[52] Ahmed J, Salam Z. A maximum Power Point Tracking (MPPT) for PV system using Cuckoo Search with partial shading capability. Appl Energy 2014;119:118–30.

[53] Moravej Z, Akhlaghi A. A novel approach based on cuckoo search for DG allocation in distribution network. Electrical Power Energy Syst 2014;44:672–9.
[54] Yang XS. A new metaheutistic bat-inspired algorithm, nature inspired cooperative strategies for optimization (NISCO 2010). Stud Comput Intell 2010;284:65–74.
[55] Wang Y, Li B, Weise T, Wang J, Yuan B, Tian Q. Self-adaptive learning based particle swarm optimization. Inf Sci 2011;181:4515–38.
[56] Yang XS, Deb S. Cuckoo Search: recent advances and applications. Neural Comput Appl 2014;24–1:169–74.