OPTIMAL CONTROL APPLICATIONS AND METHODS
Optim. Control Appl. Meth. 2012; **33**:576–594
Published online 12 August 2011 in Wiley Online Library (wileyonlinelibrary.com). DOI: 10.1002/oca.1015

Towards global solutions of optimal discrete-valued control problems

Siew Fang Woon[1], Volker Rehbock[2,*,†] and Ryan Loxton[2]

[1]*Physical Sciences, College of Arts and Sciences, Universiti Utara Malaysia, Sintok, 06010 Kedah, Malaysia*
[2]*Department of Mathematics and Statistics, Curtin University, GPO Box U1987, Perth, WA 6845, Australia*

SUMMARY

This paper proposes a new heuristic approach for solving optimal discrete-valued control problems. We illustrate the approach with an existing hybrid power system model. The problem of choosing an operating schedule to minimize generator, battery, and switching costs is first posed as a mixed discrete dynamic optimization problem. Then, a discrete filled function method is employed in conjunction with a computational optimal control technique to solve this problem. Computational results indicate that this approach is robust, efficient, and can successfully identify a near-global solution for this complex applied optimization problem despite the presence of multiple local optima. Copyright © 2011 John Wiley & Sons, Ltd.

Received 3 June 2009; Revised 17 June 2011; Accepted 17 June 2011

KEY WORDS: optimal control; hybrid power system; global optimization; discrete filled function; mixed discrete optimization

1. INTRODUCTION

The aim of this paper is to develop a new algorithm for finding global solutions of optimal discrete-valued control problems. We demonstrate the applicability of the proposed algorithm in the context of a nonlinear constrained optimal discrete-valued control problem that is known to readily yield locally optimal solutions. The problem is to determine an optimal operating schedule for a hybrid power system.

A hybrid power system is a stand-alone electrical power system incorporating conventional (i.e., hydrocarbon powered) generators, renewable energy sources, and energy storage devices. Such systems are vital for electrification in remote areas, where grid-connected infrastructure is not available and fuel is expensive. Renewable energy sources, such as photovoltaic (PV) arrays and wind turbines are used to supplement the energy produced by the generators, thereby reducing fuel demand and maintenance costs. However, the contribution of the renewable sources to the total energy output varies considerably throughout the day. For this reason, battery banks are typically used to store excess energy generated from both conventional and renewable resources [1].

Apart from the start-up costs, the dominant running costs of a hybrid power system are associated with diesel generators and battery banks. The operating cost of a diesel generator depends on fuel consumption, maintenance costs, and loading. Frequent starts of the diesel generator from cold and running the generator for long hours at a low load increase engine wear and reduce fuel efficiency. On the other hand, incomplete charging and prolonged operation of a battery bank at a low charge state are two of the major factors limiting the battery bank life span. In fact, studies have shown that diesel generators and battery banks are likely to have significantly shortened lifetimes

*Correspondence to: Volker Rehbock, Department of Mathematics and Statistics, Curtin University, GPO Box U1987, Perth, WA 6845, Australia.
†E-mail: rehbock@maths.curtin.edu.au

when operated under non-ideal conditions [1–3]. Hence, an efficient generator operating schedule is required to ensure a continuous electricity supply at the load, while at the same time keeping operating costs to a minimum. We adopt the model developed in [3], which is based on a hybrid power system consisting of a diesel generator as the main component, with a PV array providing additional energy and a battery bank for storage. The work in [3] concentrated on developing a mathematical model for hybrid power system operation and the application of a specialized optimal control technique to optimize the operation of the model. Further investigation has revealed that this optimization problem has many local minimizers.

In this paper, we propose a modified time scaling transformation that results in a new problem with less decision variables than the problem formed by the approach in [3]. Therefore, we expect that it has fewer local minimizers and that a global solution can be obtained with less computational effort. We then develop a new algorithm that is able to bypass locally optimal solutions and thus yield a solution closer to the global one. This algorithm is partly based on the concept of a discrete filled function, which has recently been proposed for the global optimization of a broad class of discrete optimization problems. Note that similar algorithms for nonlinear mixed discrete optimization problems have been suggested in [4, 5]. However, [5] deals with an entirely different application problem and [4] only presents a very simple numerical example. Both of these papers also employ different, less effective, filled functions. For other global methods for optimal control problems, see reference [6].

Our proposed approach is based on a time scaling transformation initially developed in [7]. The earlier transformation, which aims to capture a large variety of possible switching sequences, is discussed in [3]. However, this approach introduces a large number of artificial switches, many of which are not utilized in the optimal solution. Consequently, the resulting optimization problem has many local minima and some of these local minima have high objective values. Our proposed transformation, like the one in [7], results in known and fixed switching instants that allow more accurate numerical integration [7, 8].

The remainder of this paper is organized as follows. We first review the hybrid power system model and problem formulation reported in [3]. We then propose a new transformation, which converts the original problem into an equivalent mixed discrete optimization problem. In Section 3, we outline a discrete filled function method and, based on this, develop a new heuristic algorithm to solve the problem at hand. Numerical results from the implementation of this algorithm are presented in Section 4. Conclusions and some suggestions for future work are made in Section 5.

2. PROBLEM FORMULATION

2.1. *Hybrid power system*

In this section, we briefly review the dynamic model of a hybrid power system discussed in [3]. Figure 1 illustrates the configuration of the hybrid power system under consideration. It consists

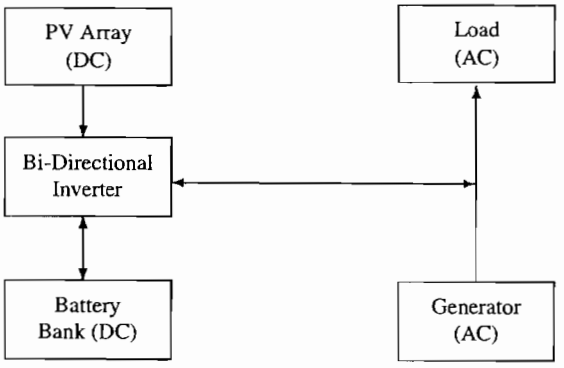

Figure 1. Schematic diagram of a hybrid power system.

DOI: 10.1002/oca

of an alternating current (AC) diesel generator, a bi-directional inverter, a PV array, and a battery bank for energy storage. The inverter is used to convert the direct current (DC) voltage used by the PV array and the battery bank into AC, and vice versa. It also acts as the battery charger. The diesel generator is connected directly to the load to avoid conversion losses and thus increase the efficiency of the power system. Other configurations involving multiple generators and/or renewable energy sources can be readily modeled with an approach similar to that used below. The assumed load demand profile (Figure 2) is based on data provided by the Centre for Renewable Energy and Sustainable Technologies Australia (CRESTA) [3]. The total daily load demand is approximately 340 kWh.

We model the hybrid power system over the time horizon $[0, t_f]$, where t_f is the given terminal time. At each $t \in [0, t_f]$, there are three possible scenarios:

(i) The diesel generator is producing sufficient energy to meet the load demand, and any excess power from the generator or PV array is directed to the battery bank.
(ii) The power from the generator is insufficient to meet the load demand, so energy produced from the PV array is also used to supply the load. Any excess is directed to the battery bank.
(iii) The combined power output from the diesel generator and PV array is insufficient to meet the load demand, so energy from the battery bank is required to make up the shortfall.

Let $C(t)$ denote the charge state of the battery bank at time $t \in [0, t_f]$. Then, on the basis of the above operating principles, the rate of change of the charge state is governed by the following dynamic system:

$$\dot{C}(t) = \begin{cases} \dfrac{K_1 K_3 [P_R(t) + P_G(t) - P_L(t)]}{K_1 + C(t)}, & \text{if } P_G(t) \geq P_L(t), \\ \dfrac{K_1 [K_3 P_R(t) + P_G(t) - P_L(t)]}{K_1 + C(t)}, & \text{if } P_G(t) + K_3 P_R(t) \geq P_L(t) > P_G(t), \\ K_2 \left[P_R(t) - \dfrac{P_L(t) - P_G(t)}{K_3} \right] & \text{if } P_G(t) + K_3 P_R(t) < P_L(t), \end{cases} \tag{1}$$

and

$$C(0) = C_0, \tag{2}$$

where C_0 is the given initial charge state, $P_R(t)$ is the power generated by the PV array at time t, $P_G(t)$ is the power produced by the diesel generator at time t, and $P_L(t)$ is the load demand at time t. K_1 and K_2 are constants of the battery model related to the charging and discharging efficiency, respectively. For example, assuming that the maximum capacity of the battery bank is

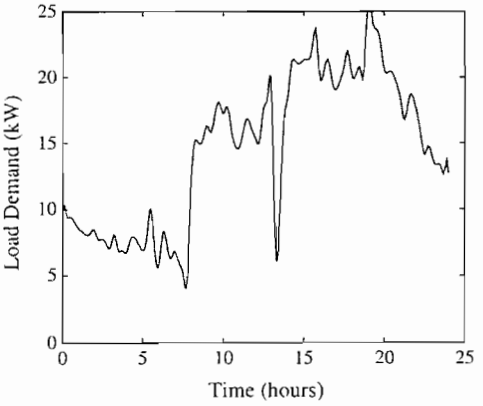

Figure 2. A typical load demand profile [3].

DOI: 10.1002/oca

100 kWh, a value of $K_1 = 250$ means that the charging efficiency near full battery charge is just over 70% of the corresponding charging efficiency at a near empty battery state. For realistic operation, $K_2 = 1.4$, that is, only about 70% of power stored in the battery can be converted for load use. These values of K_1 and K_2 assume the use of deep cycle lead acid batteries. The constant K_3 reflects the inverter efficiency, typically about 90% ($K_3 = 0.9$). Both P_R and P_L are given functions derived from actual data supplied by CRESTA. On the other hand, P_G is the control function that is chosen by the system operator in practice.

Because the charge state must operate within a certain range, we have the following constraints:

$$C_{\min} \leq C(t) \leq C_{\max}, \qquad \forall t \in [0, t_f] \tag{3}$$

and

$$C(t_f) = C_f, \tag{4}$$

where C_f is the desired final charge state, and $C_{\min}$ and $C_{\max}$ are given constants.

Because it is difficult to continuously modify the power produced by the generator, we assume that the generator can only operate at certain fixed fractions of its capacity. Suppose that there are M such levels. Then we require

$$P_G(t) \in S = \{s_1, \ldots, s_M\}, \qquad \forall t \in [0, t_f],$$

where, for each $i = 1, \ldots, M$, s_i denotes the power produced by the generator in mode i. According to [3], the operating cost of the diesel generator is given by

$$\int_0^{t_f} P_G(t) g_1\left(\frac{100 P_G(t)}{P_{G,\max}}\right) \mathrm{d}t,$$

where $P_{G,\max}$ is the maximum power produced by the generator and

$$g_1(x) = 2\left((0.2x + 0.5)^{0.4} - 0.5^{0.4}\right) e^{-0.1x} + 0.15(1 - e^{-0.1x})$$

is a function that reflects the fuel efficiency of a typical diesel generator. Its shape (Figure 3) is based on data from [9], which itself is illustrated in Figure 4. Note that g_1 was chosen to satisfy the requirement $g_1(0) = 0$ and with tail behavior similar to that of the efficiency curve in Figure 4. Note also that $g_1(x)$ values for x between 0% and 40% are not utilized because the generator is not assumed to operate in this range.

The model in [3] also proposes the following cost term for the usage of the battery bank:

$$\int_0^{t_f} (C(t) - K_4)^2 \, \mathrm{d}t,$$

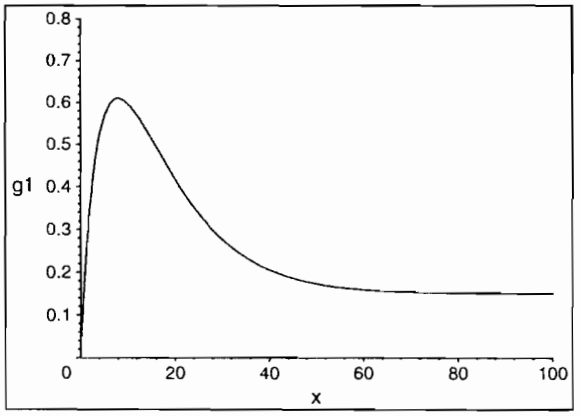

Figure 3. Profile of function $g_1(x)$.

DOI: 10.1002/oca

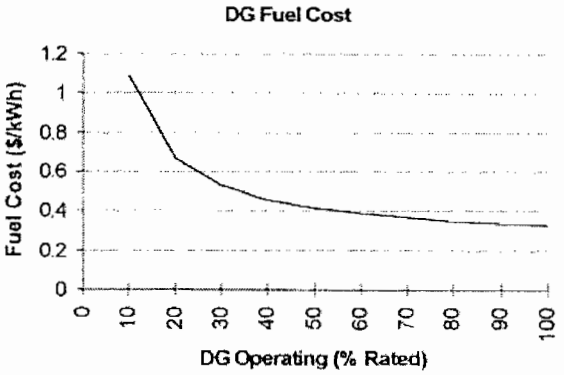

Figure 4. Typical fuel efficiency of diesel generator.

where K_4 is a constant that represents a desired battery charge level. A desired charge level of about 80%, for example, would significantly penalize deep cycle discharges (defined as those where the charge state of a battery drops to below 20% of its nominal capacity) that are the most damaging in terms of battery longevity. Note that a desired charge level near 100% is not reasonable as it would discourage the actual use of the battery bank altogether.

With these cost terms in mind, we can state the following optimal control problem.

Problem (A). Choose a discrete-valued control $P_G : [0, t_f] \to S$ such that the cost function

$$\alpha \int_0^{t_f} P_G(t) g_1 \left(\frac{100 P_G(t)}{P_{G,max}} \right) \, dt + \beta \int_0^{t_f} (C(t) - K_4)^2 \, dt$$

is minimized subject to the dynamics (1)–(2) and the constraints (3)–(4), where α and β are non-negative weights.

Problem (A) is a discrete-valued optimal control problem in which the control is restricted to take values in a discrete set. To determine the optimal discrete-valued control, we need to determine the order in which the different power levels are implemented (the switching sequence) and the times at which the power levels are changed (the switching times). However, conventional computational optimal control techniques are designed for problems in which the control takes values in a connected set, such as an interval, and hence they cannot solve Problem (A) directly. Moreover, variable switching times are known to cause problems in the implementation of any numerical algorithm [7,8]. In the next subsection, we propose a new transformation to overcome these difficulties. This transformation introduces a new discrete variable to represent the switching sequence and a new continuous variable to represent the switching times. Using this transformation, we derive a new problem that is equivalent to Problem (A).

2.2. *A modified time scaling transformation*

Suppose that we allow the control to switch N times over the time horizon. Define a new time variable $\tau \in [0, N+1]$ with the partition $P_N = \{0, 1, 2, \ldots, N, N+1\}$. For each $i = 1, \ldots, N+1$, let

$$\upsilon_i \in \{1, \ldots, M\}$$

be a discrete variable representing the mode of the generator during the i-th subinterval. Let

$$\mathbf{v} = [\upsilon_1, \ldots, \upsilon_{N+1}]^T$$

and V be the set of all such vectors. For each $i = 1, \ldots, N+1$, we define a new control function $U_G(\tau, \mathbf{v})$ by

$$U_G(\tau, \mathbf{v}) = s_{\upsilon_i}, \qquad \tau \in [i-1, i).$$

DOI: 10.1002/oca

Hence, $U_G(\tau, \mathbf{v})$ represents $P_G(t)$ in the new time scale. Next, $u(\tau)$, the time scaling control, is defined as a piecewise constant function with possible discontinuities at $1, 2, \ldots, N$ and satisfying

$$0 \leqslant u(\tau) \leqslant t_f, \qquad \tau \in [0, N+1]. \tag{5}$$

Let U denote the class of all valid time scaling controls satisfying (5). The original time horizon $[0, t_f]$ is transformed into the new time horizon $[0, N+1]$ through the differential equation

$$\dot{t}(\tau) = u(\tau) \tag{6}$$

and

$$t(0) = 0, \tag{7}$$

and with the additional constraint

$$t(N+1) = t_f. \tag{8}$$

Therefore, the original dynamics (1)–(2) are transformed into

$$\dot{\bar{C}}(\tau) = \begin{cases} \dfrac{K_1 K_3 [P_R(t(\tau)) + U_G(\tau, \mathbf{v}) - P_L(t(\tau))]}{K_1 + \bar{C}(\tau)} u(\tau), & \text{if } U_G(\tau, \mathbf{v}) \geqslant P_L(t(\tau)), \\ \dfrac{K_1 [K_3 P_R(t(\tau)) + U_G(\tau, \mathbf{v}) - P_L(t(\tau))]}{K_1 + \bar{C}(\tau)} u(\tau) & \text{if } U_G(\tau, \mathbf{v}) + K_3 P_R(t(\tau)) \\ & \quad \geqslant P_L(t(\tau)) > U_G(\tau, \mathbf{v}), \\ K_2 \left[P_R(t(\tau)) - \dfrac{P_L(t(\tau)) - U_G(t, \mathbf{v})}{K_3} \right] u(\tau) & \text{if } U_G(\tau, \mathbf{v}) + K_3 P_R(t(\tau)) < P_L(t(\tau)), \end{cases} \tag{9}$$

and

$$\bar{C}(0) = C_0. \tag{10}$$

Similarly, constraints (3) and (4) are transformed into

$$C_{\min} \leqslant \bar{C}(\tau) \leqslant C_{\max}, \quad \forall \tau \in [0, N+1], \tag{11}$$

and

$$\bar{C}(N+1) = C_f. \tag{12}$$

After the transformation, the terms measuring the fuel cost and the operating cost of the battery are

$$\int_0^{N+1} U_G(\tau, \mathbf{v}) g_1 \left(\frac{100 U_G(\tau, \mathbf{v})}{P_{G,\max}} \right) u(\tau) \mathrm{d}\tau$$

and

$$\int_0^{N+1} (\bar{C}(\tau) - K_4)^2 u(\tau) \mathrm{d}\tau,$$

respectively. On the basis of the above discussion, we have the following problem, which is equivalent to Problem (A).

Problem (B). Choose $\mathbf{v} \in V$ and $u \in U$ such that the cost function

$$\alpha \int_0^{N+1} U_G(\tau, \mathbf{v}) g_1 \left(\frac{100 U_G(\tau, \mathbf{v})}{P_{G,\max}} \right) u(\tau) \mathrm{d}\tau + \beta \int_0^{N+1} (\bar{C}(\tau) - K_4)^2 u(\tau) \mathrm{d}\tau$$

is minimized subject to the dynamics (6)–(10) and the constraints (11)–(12), where α and β are non-negative weights.

DOI: 10.1002/oca

2.3. *Penalizing frequent switching*

Frequent switching is undesirable in practice because it significantly increases mechanical wear. However, there is no mechanism in Problem (B) to discourage a control schedule that frequently switches between generator modes. Hence, we would also like to minimize the term

$$\int_0^{N+1} g_2(u(\tau))\mathrm{d}\tau, \tag{13}$$

where

$$g_2(x) = ((x+0.01)^{0.25} - 0.01^{0.25})e^{-5x}.$$

The function g_2 is illustrated in Figure 5. Note that its shape is chosen so that the objective term (13) severely penalizes an operating schedule that runs any generator mode for a nonzero duration of less than 15 min, while not penalizing a zero mode duration and only lightly penalizing a duration of 1 h or greater. Our new problem is stated below.

Problem (C). Choose $\mathbf{v} \in V$ and $u \in U$ such that the cost function

$$\int_0^{N+1} \left\{ \alpha U_{\mathrm{G}}(\tau, \mathbf{v}) g_1\left(\frac{100 U_{\mathrm{G}}(\tau, \mathbf{v})}{P_{\mathrm{G,max}}}\right) u(\tau) + \beta(\bar{C}(\tau) - K_4)^2 u(\tau) + \gamma g_2(u(\tau)) \right\} \mathrm{d}\tau$$

is minimized subject to dynamics (6)–(10) and the constraints (11)–(12), where α, β, and γ are non-negative weights. Calibration of the model with a real hybrid power system would allow appropriate values for these weights to be chosen, but the means for doing so were not available to the authors of [3] at the time. Instead, values for α, β, and γ were chosen so that each respective objective term had a significant influence on the final solution.

Note that the hybrid power system model developed in [3] and presented here attempts to capture most of the significant cost issues present in real hybrid power systems with the aim of identifying the best possible operating policies that may be achieved in an idealized deterministic environment. These optimal policies can then be used as guidance for the design and control of actual systems. In contrast, most other hybrid power system models in the literature focus mainly on the control of the system with little emphasis on optimization (see, for example, [10] and the references cited therein).

In summary, Problem (C) is a mixed discrete dynamic optimization problem. To facilitate the application of a global optimization technique, we decompose it into a bi-level optimization problem in the next subsection.

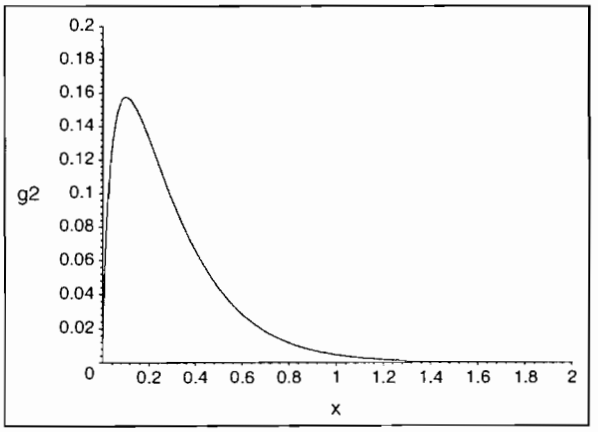

Figure 5. Profile of function $g_2(x)$.

DOI: 10.1002/oca

2.4. *Decomposition of Problem (C)*

In our numerical experiments, we have observed that multiple locally optimal solutions are found when different initial switching times are used to solve the model developed in [3] using the transformation and solution technique suggested there. Many practical discrete-valued optimal control problems exhibit similar behavior. Thus, with the transformation leading to Problem (C), we intend to apply a global optimization technique, known as the discrete filled function method, in an attempt to determine a global optimal solution. For this purpose, we restructure Problem (C) by decomposing it into a bi-level optimization problem as follows.

Problem $\mathbf{C}_1$. Given $\mathbf{v} \in V$, choose a $u \in U$ such that the cost function

$$g_0(u|\mathbf{v}) = \int_0^{N+1} \left\{ \alpha U_{\mathrm{G}}(\tau, \mathbf{v}) g_1 \left(\frac{100 U_{\mathrm{G}}(\tau, \mathbf{v})}{P_{\mathrm{G,max}}} \right) u(\tau) + \beta (\bar{C}(\tau) - K_{\mathrm{P}} 4)^2 u(\tau) + \gamma g_2(u(\tau)) \right\} \mathrm{d}\tau \tag{14}$$

is minimized subject to the dynamics (6)–(10) and the constraints (11)–(12), where α, β, and γ are non-negative weights.

Problem (C_1) is known as the lower level problem or the subproblem. It is simply a standard optimal control problem where the optimal value of g_0 in (14) can be determined using an optimal control software. There are many suitable softwares available for this purpose. We choose to use MISER3.3 [11] because it is based on the concept of control parametrization and caters directly for the piecewise constant control formulation that we employ here. It also allows the user to readily incorporate a wide range of constraints. The second problem in the decomposition is defined as follows.

Problem $\mathbf{C}_2$. Choose $\mathbf{v} \in V$ such that the cost function

$$J(\mathbf{v}) = \min_{u \in U} g_0(u|\mathbf{v}) \tag{15}$$

is minimized.

Problem (C_2) is the upper level of Problem (C). Clearly, Problem (C_2) is a purely discrete optimization problem. To compute the value of the objective function at $\mathbf{v} \in V$, we need to solve the subproblem (C_1) corresponding to $\mathbf{v} \in V$. Next, we propose a combined algorithm in which Problem (C_2) will be solved using a discrete filled function method to determine a global solution and subproblem (C_1) is solved with MISER3.3. For our numerical computations, we have been able to incorporate the discrete filled function method into the MISER3.3 software. The details of the discrete filled function approach are discussed in the next section.

3. DISCRETE FILLED FUNCTION METHOD

Complex nonlinear discrete optimization problems are typically NP-hard problems. This means that there are no efficient algorithms with polynomial-time complexity for determining an optimal solution. Thus, an intensive heuristic approach is often employed to solve them. Below, we recall some basic definitions from the field of discrete optimization, which are given in [12, 13].

3.1. *Preliminaries*

Let $\mathbf{e}_i$ denote the i-th standard unit basis vector of a Euclidean vector space with its i-th component equal to one and all other components equal to zero.

(i) For any $\mathbf{v} \in V$, the *neighborhood* of $\mathbf{v}$ is defined by

$$N(\mathbf{v}) = \{\mathbf{w} \in V : \mathbf{w} = \mathbf{v} \pm \mathbf{e}_i,\ i = 1, 2, \ldots, N+1\}.$$

(ii) The *set of all feasible directions at* $\mathbf{v} \in V$ is defined by

$$\mathcal{D}(\mathbf{v}) = \{\mathbf{d} \in \mathbb{R}^{N+1} : \mathbf{v} + \mathbf{d} \in N(\mathbf{v})\} \subset \{\pm\mathbf{e}_1, \ldots, \pm\mathbf{e}_{N+1}\}.$$

(iii) $\mathbf{d} \in \mathcal{D}(\mathbf{v})$ is a *descent direction* of J at $\mathbf{v}$ if $J(\mathbf{v} + \mathbf{d}) < J(\mathbf{v})$.

DOI: 10.1002/oca

(iv) $\mathbf{d}^* \in \mathcal{D}(\mathbf{v})$ is a *steepest descent direction of* J *at* $\mathbf{v}$ if it is a descent direction and $J(\mathbf{v}+\mathbf{d}^*) \leqslant J(\mathbf{v}+\mathbf{d})$ for each $\mathbf{d} \in \mathcal{D}(\mathbf{v})$.
(v) $\mathbf{v}^* \in V$ is a *local minimizer* of V if $J(\mathbf{v}^*) \leqslant J(\mathbf{v})$ for all $\mathbf{v} \in N(\mathbf{v}^*)$. If $J(\mathbf{v}^*) < J(\mathbf{v})$ for all $\mathbf{v} \in N(\mathbf{v}^*)$, then $\mathbf{v}^*$ is a *strict local minimizer* of J.
(vi) $\mathbf{v}^*$ is a *global minimizer* of J if $J(\mathbf{v}^*) \leqslant J(\mathbf{v})$ for all $\mathbf{v} \in V$. If $J(\mathbf{v}^*) < J(\mathbf{v})$ for all $\mathbf{v} \in V \setminus \{\mathbf{v}^*\}$, then $\mathbf{v}^*$ is a *strict global minimizer* of J.
(vii) $\mathbf{v}$ is a *vertex of* V if for each $\mathbf{d} \in \mathcal{D}(\mathbf{v})$, $\mathbf{v}+\mathbf{d} \in V$ and $\mathbf{v}-\mathbf{d} \notin V$. Let $\tilde{V}$ denote the set of vertices of V.
(viii) A sequence $\{\mathbf{v}^{(i)}\}_{i=0}^{k+1}$ between two distinct points $\mathbf{v}^*$ and $\mathbf{v}^{**}$ in V is a *discrete path in* V if $\mathbf{v}^{(0)} = \mathbf{v}^*$, $\mathbf{v}^{(k+1)} = \mathbf{v}^{**}$, $\mathbf{v}^{(i)} \in V$ for all i, $\mathbf{v}^{(i)} \neq \mathbf{v}^{(j)}$ for $i \neq j$, and $\| \mathbf{v}^{(i+1)} - \mathbf{v}^{(i)} \| = 1$ for all i. If such a discrete path exists, then $\mathbf{v}^*$ and $\mathbf{v}^{**}$ are *pathwise connected in* V.

Algorithm 1
Discrete Steepest Descent Method

1. Choose an initial switching sequence $\mathbf{v} \in V$.
2. If $\mathbf{v}$ is a local minimizer of J, then stop. Otherwise, find the discrete steepest descent direction $\mathbf{d}^* \in \mathcal{D}(\mathbf{v})$ of J.
3. Let $\mathbf{v} := \mathbf{v} + \mathbf{d}^*$. Go to Step 2.

3.2. Discrete filled function

The filled function approach is a global optimization method that was introduced in the late 1980s [14]. The filled function concept initially focused on solving continuous global optimization problems. Various filled functions with improved theoretical properties have been developed to enhance computational efficiency [12–21]. Reference [14] appears to be the first publication to adapt the continuous filled function approach to solving discrete optimization problems. However, the filled function proposed in [14] contains an exponential term, which makes it difficult to determine an improved local minimizer [13]. Several improved discrete filled functions have been proposed in [12, 13, 19–21] since then.

In this paper, we employ a discrete filled function method, recently developed in [13], as part of our proposed algorithm. The basic idea of this method is as follows. We choose an initial point and then perform a local search (Algorithm 1) to find an initial local minimizer. Then, we construct an auxiliary function, called a filled function, at this local minimizer. By minimizing the filled function, either an improved local minimizer is found or one of the vertices is reached; otherwise, the parameters of the filled function are adjusted. This process is repeated until no better local minimizer of the corresponding filled function is found. The final local minimizer is then taken as an approximation of the global minimizer.

Definition 1
Let $\hat{V}(\mathbf{v}^*) = \{\mathbf{v} \in V : \mathbf{v} \neq \mathbf{v}^*, J(\mathbf{v}) \geqslant J(\mathbf{v}^*)\}$. A function $G_{\mathbf{v}^*} : V \mapsto \mathbb{R}$ is called a discrete filled function of J at $\mathbf{v}^*$ if it satisfies the following conditions:

(a) $\mathbf{v}^*$ is a strict local maximizer of $G_{\mathbf{v}^*}$ over V;
(b) $G_{\mathbf{v}^*}$ has no local minimizer in the set $\hat{V}(\mathbf{v}^*) \setminus \tilde{V}$;
(c) $\mathbf{v}^{**} \in V \setminus \tilde{V}$ is a local minimizer of J over V if and only if $\mathbf{v}^{**}$ is a local minimizer of $G_{\mathbf{v}^*}$ over V.

Define

$$G_{\mu,\rho,\mathbf{v}^*}(\mathbf{v}) = A_\mu(J(\mathbf{v}) - J(\mathbf{v}^*)) - \rho \| \mathbf{v} - \mathbf{v}^* \|, \tag{16}$$

where

$$A_\mu(y) = y \cdot \mu \left[(1-c) \left(\frac{1-c\mu}{\mu - c\mu} \right)^{-y/\omega} + c \right].$$

DOI: 10.1002/oca

The parameter $\omega > 0$ is a sufficiently small number and $0 < c \leqslant 1$ is a constant. The function $G_{\mu,\rho,\mathbf{v}^*}(\mathbf{v})$ is a discrete filled function when certain conditions on the parameters μ and ρ are satisfied. Hence, as long as the conditions on μ and ρ are satisfied, $G_{\mu,\rho,\mathbf{v}^*}(\mathbf{v})$ possesses properties (a)–(c) in Definition 1.

Definition 2
Let $\mathcal{K}$ be a constant satisfying

$$1 \leqslant \max_{\substack{\mathbf{v}_1,\mathbf{v}_2 \in V \\ \mathbf{v}_1 \neq \mathbf{v}_2}} \| \mathbf{v}_1 - \mathbf{v}_2 \| \leqslant \mathcal{K} < \infty,$$

where $\| \cdot \|$ is the Euclidean norm. Let $0 < \mathcal{L} < \infty$ be the Lipschitz constant such that

$$|J(\mathbf{v}_1) - J(\mathbf{v}_2)| \leqslant \mathcal{L} \| \mathbf{v}_1 - \mathbf{v}_2 \|$$

for any distinct $\mathbf{v}_1, \mathbf{v}_2 \in V$.

Theorem 1
If $\rho > 0$ and $0 < \mu < \min\{1, \frac{\rho}{\mathcal{L}}\}$, then $\mathbf{v}^*$ is a strict local maximizer of $G_{\mu,\rho,\mathbf{v}^*}$. If $\mathbf{v}^*$ is a global minimizer of J, then $G_{\mu,\rho,\mathbf{v}^*}(\mathbf{v}) < 0$ for all $\mathbf{v} \in V \setminus \{\mathbf{v}^*\}$.

Theorem 2
Let $\mathbf{v}^{**}$ be a strict local minimizer of J with $J(\mathbf{v}^{**}) < J(\mathbf{v}^*)$. If $\rho > 0$ is sufficiently small and $0 < \mu < 1$, then $\mathbf{v}^{**}$ is a strict local minimizer of $G_{\mu,\rho,\mathbf{v}^*}$.

Theorem 3
Let $\hat{\mathbf{v}}$ be a strict local minimizer of $G_{\mu,\rho,\mathbf{v}^*}$ and suppose that there exists a $\bar{\mathbf{d}} \in \mathcal{D}(\hat{\mathbf{v}})$ such that

$$\| \hat{\mathbf{v}} + \bar{\mathbf{d}} - \mathbf{v}^* \| > \| \hat{\mathbf{v}} - \mathbf{v}^* \| .$$

If $\rho > 0$ is sufficiently small and $0 < \mu < \min\{1, \frac{\rho}{2\mathcal{K}^2\mathcal{L}}\}$, then $\hat{\mathbf{v}}$ is a local minimizer of J.

Corollary 1
Assume that every local minimizer of J is strict. Suppose that $\rho > 0$ is sufficiently small and $0 < \mu < \min\{1, \frac{\rho}{2\mathcal{K}^2\mathcal{L}}\}$. Then, $\mathbf{v}^{**} \in V \setminus \tilde{V}$ is a local minimizer of J with $J(\mathbf{v}^{**}) < J(\mathbf{v}^*)$ if and only if $\mathbf{v}^{**}$ is a local minimizer of $G_{\mu,\rho,\mathbf{v}^*}$.

Note that a detailed convergence analysis for this method and extensive numerical tests on discrete optimization problems have been given in [13]. Based on the work in [13], we propose the following modified discrete filled function algorithm to solve our optimal control problem.

Algorithm 2
Discrete Filled Function Method

1. Choose an initial sequence $\mathbf{v}_0 \in V$, ρ_0, μ_0, $\rho_L > 0$, $0 < \hat{\rho} < 1$, and $0 < \hat{\mu} < 1$.
   Let $\rho := \rho_0$ and $\mu := \mu_0$.
2. Starting from $\mathbf{v}_0$, minimize $J(\mathbf{v})$ using Algorithm 1 to obtain a local minimizer $\mathbf{v}^*$ of J.
3. (a) List the neighboring sequences of $\mathbf{v}^*$ as $N(\mathbf{v}^*) = \{\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_q\}$. Set $\ell := 1$.
   (b) Set the current switching sequence, $\mathbf{v}_c := \mathbf{w}_\ell$.
4. (a) If there exists a direction $\mathbf{d} \in \mathcal{D}(\mathbf{v}_c)$ such that $J(\mathbf{v}_c + \mathbf{d}) < J(\mathbf{v}^*)$, then set $\mathbf{v}_0 := \mathbf{v}_c + \mathbf{d}$ and go to Step 2. Otherwise, go to (b) below.
   (b) Let $\mathcal{D}_1 = \{\mathbf{d} \in \mathcal{D}(\mathbf{v}_c) : J(\mathbf{v}_c + \mathbf{d}) < J(\mathbf{v}_c)$ and $G_{\mu,\rho,\mathbf{v}^*}(\mathbf{v}_c + \mathbf{d}) < G_{\mu,\rho,\mathbf{v}^*}(\mathbf{v}_c)\}$.
   If $\mathcal{D}_1 \neq \emptyset$, set $\mathbf{d}^* := \arg\min_{\mathbf{d} \in \mathcal{D}_1}\{J(\mathbf{v}_c + \mathbf{d}) + G_{\mu,\rho,\mathbf{v}^*}(\mathbf{v}_c + \mathbf{d})\}$.
   Then, set $\mathbf{v}_c := \mathbf{v}_c + \mathbf{d}^*$ and go to Step 4(a). Otherwise, go to (c) below.
   (c) Let $\mathcal{D}_2 = \{\mathbf{d} \in \mathcal{D}(\mathbf{v}_c) : G_{\mu,\rho,\mathbf{v}^*}(\mathbf{v}_c + \mathbf{d}) < G_{\mu,\rho,\mathbf{v}^*}(\mathbf{v}_c)\}$.
   If $\mathcal{D}_2 \neq \emptyset$, set $\mathbf{d}^* := \arg\min_{\mathbf{d} \in \mathcal{D}_2}\{G_{\mu,\rho,\mathbf{v}^*}(\mathbf{v}_c + \mathbf{d})\}$.
   Then, set $\mathbf{v}_c := \mathbf{v}_c + \mathbf{d}^*$ and go to Step 4(a). Otherwise, go to Step 5.

DOI: 10.1002/oca

5. Let $\hat{\mathbf{v}}$ be the obtained local minimizer of $G_{\mu,\rho,\mathbf{v}^*}$.
   (a) If $\hat{\mathbf{v}} \in \tilde{V}$, set $\ell := \ell + 1$. If $\ell > q$, go to Step 6. Otherwise, go to Step 3(b).
   (b) If $\hat{\mathbf{v}} \notin \tilde{V}$, reduce μ by setting $\mu := \hat{\mu}\mu$ and go to Step 4(b).
6. Reduce ρ by setting $\rho := \hat{\rho}\rho$. If $\rho < \rho_L$, terminate the algorithm. The current $\mathbf{v}^*$ is taken as a global minimizer of the problem. Otherwise, reset $\ell := 1$ and go to Step 3(b).

The mechanism of this algorithm is described as follows. Firstly, the parameters of the discrete filled function $G_{\mu,\rho,\mathbf{v}^*}$ in (16) are initialized to suitable values in Step 1. These parameters will be reduced gradually in Steps 5 and 6 to ensure that $G_{\mu,\rho,\mathbf{v}^*}$ eventually satisfies properties (a)–(c). The reduction factors for each of these parameters are also specified at Step 1.

Secondly, we choose an initial sequence $\mathbf{v}_0$ in the feasible region and minimize the original function J. Recall that the value of J for each switching sequence is computed using MISER3.3 according to the discussion in the previous section. The objective function value at each sequence in the neighborhood of $\mathbf{v}_0$ is calculated. The search direction leading to the most improved objective function value in this neighborhood is chosen in accordance with Algorithm 1. The process is repeated until a local minimizer of J, namely $\mathbf{v}^*$, is found. Next, we identify the neighborhood of $\mathbf{v}^*$ in Step 3. One of the neighboring points of $\mathbf{v}^*$, denoted by $\mathbf{v}_c$, is set to be an initial point from which we minimize the discrete filled function $G_{\mu,\rho,\mathbf{v}^*}$ in the following step. Note that $\mathbf{v}^*$ is a local maximizer of $G_{\mu,\rho,\mathbf{v}^*}$ here.

In Step 4, we first check to see if there exists a neighboring sequence of $\mathbf{v}_c$ that is an improvement over the current minimizer. If such a sequence can be found, then we use it as a starting point to minimize the function J using Algorithm 1. Otherwise, if we can find a direction that results in an improvement in both J and G compared with the values at $\mathbf{v}_c$, then we choose the direction which gives the greatest such improvement. If such a direction does not exist, then we try to find a steepest descent direction such that $G_{\mu,\rho,\mathbf{v}^*}(\mathbf{v}_c + \mathbf{d}^*) < G_{\mu,\rho,\mathbf{v}^*}(\mathbf{v}_c)$. If none of these directions exists, then $\mathbf{v}_c$ must be a local minimizer of $G_{\mu,\rho,\mathbf{v}^*}$, so we go to Step 5.

If the local minimizer of $G_{\mu,\rho,\mathbf{v}^*}$ is found to be a vertex of the feasible region, then we choose the next point in $N(\mathbf{v}^*)$ as a starting point to minimize $G_{\mu,\rho,\mathbf{v}^*}$ in Step 5(a). Note that the minimizer of $G_{\mu,\rho,\mathbf{v}^*}$ must be either an improved point or a vertex provided that the parameters are chosen correctly. Thus, μ is adjusted suitably to satisfy this criteria in Step 5(b).

If no improved sequence is found and the minimization process starting from each neighboring sequence ends up at the vertices, then we reduce ρ, reset $\ell = 1$, and minimize $G_{\mu,\rho,\mathbf{v}^*}$ again with the new value of ρ. The algorithm is repeated until the termination criterion is reached, where ρ reaches its lower bound, ρ_L. In this case, we have minimized the discrete filled function from every search direction from $\mathbf{v}^*$ and failed to find an improved point with a range of parameter settings. $\mathbf{v}^*$ is then taken to be the global solution of J.

To increase the efficiency, we construct a look-up table to store each value of the objective function J computed so far. Thus, we avoid repeated application of the subproblem solution algorithm at the same point. This is vital to the computational efficiency because computing $J(\mathbf{v})$ involves solving a complex optimal control problem, which takes considerable computational time.

Note that for some sequences, the subproblem solution algorithm may report that Problem (C_1) is infeasible. This may be because the subproblem solver (MISER3.3) does not converge or it may actually indicate that the subproblem is infeasible at the current sequence $\mathbf{v}$. In an effort to distinguish between these two possibilities, we re-initialize the optimization of the subproblem several times. When five such attempts fail to yield a feasible solution, it is assumed that no feasible solution of the subproblem exists for this switching sequence $\mathbf{v}$. An artificially high cost is assigned to such a sequence and the algorithm is allowed to continue.

Remark 1

Note that the discrete filled function algorithm we adopted from [13] is designed for a box constrained (or linear inequality constrained) problem where the feasible search region is pathwise connected and has easily identifiable vertices. These properties are not necessarily met in the application of the algorithm to Problem (C_2) because the feasibility of a point is not known until an attempt has been made to solve the corresponding subproblem. Although we do not remove such

DOI: 10.1002/oca

a point from the search region directly, we assign an artificially high cost to it. It may well be the case that the effective feasible region of Problem (C_2) becomes non-convex and non-connected. However, it is difficult to ascertain this behavior beforehand and our application of the algorithm to Problem (C_2) must hence be viewed as a heuristic approach. Nevertheless, numerical results suggest that the proposed approach is capable of determining significantly improved solutions when compared with local methods.

4. NUMERICAL RESULTS

In this section, our algorithm is applied to solve Problem (C) with four, seven, and nine switches. A comparison between our method and the method in [3] is discussed at the end of this section. The results were computed using a modified version of MISER3.3 in which the filled function method repeatedly calls on the standard MISER3.3 algorithm. The experiments were conducted on a Windows-based PC, with a CPU speed of 2.4GHz and 2GB RAM.

4.1. Results for four switches

By setting $N = 4$, $K_1 = 250$, $K_2 = 1.4$, $K_3 = 0.9$, $K_4 = 80$, $C_0 = 80$ kWh, $C_{\min} = 20$ kWh, $C_{\max} = 100$ kWh, $\alpha = 1$, $\beta = 0.01$, $\gamma = 10$, $t_f = 24$, $c = 0.5$, $\mu_0 = 0.1$, $\rho_0 = 0.1$, $\omega = 1$, $\rho_L = 0.001$, $\hat{\rho} = 0.1$, $\hat{\mu} = 0.1$, we solved the resulting Problem (C). There are 3125 potential switching sequences for four switches with $U_G \in \{0, 8, 12, 16, 20\}$.

We tested the problem with 10 random initial sequences, namely, $[2,3,4,5,4]^T$, $[4,5,3,5,2]^T$, $[5,1,5,1,5]^T$, $[4,3,1,5,2]^T$, $[3,4,4,3,5]^T$, $[2,4,5,4,4,]^T$, $[2,4,5,4,1]^T$, $[5,4,3,2,3]^T$, $[2,3,2,4,5]^T$, and $[4,5,3,4,5]^T$. We found 13 local minimizers during the application of the algorithm on these ten starting points. For each starting sequence, the algorithm successfully identified the assumed discrete global minimizer, $[2,3,4,5,4]^T$, for which the cost function value is $J = 58.7216005$, and the time scaling control is

$$u(\tau) = \begin{cases} 7.50650, & 0 \leqslant \tau < 1, \\ 1.42257, & 1 \leqslant \tau < 2, \\ 5.05190, & 2 \leqslant \tau < 3, \\ 8.70635, & 3 \leqslant \tau < 4, \\ 1.31267, & 4 \leqslant \tau < 5. \end{cases}$$

Table I illustrates the computational results of 10 experiments using the same initial time scaling control set at

$$u(\tau) = \begin{cases} 1, & 0 \leqslant \tau < 1, \\ 6, & 1 \leqslant \tau < 2, \\ 8, & 2 \leqslant \tau < 3, \\ 6, & 3 \leqslant \tau < 4, \\ 3, & 4 \leqslant \tau < 5. \end{cases}$$

The number of original function evaluations and filled function evaluations are denoted by E_J and E_G, respectively. Note that E_J does not include function evaluations that were obtained from the look-up table.

The algorithm terminates when $\mu = 1 \times 10^{-41}$ and $\rho = 1 \times 10^{-3}$, at which point no further improvement can be made. Therefore, $[2,3,4,5,4]^T$ is assumed to be the globally optimal sequence for Problem (C) with $N = 4$. At most, 571 switching sequences are computed during the ten applications of the algorithm, which is 18.3% of the total possible sequences. Further experiments with a range of refined parameter values of the discrete filled function were carried out and the results also confirmed $[2,3,4,5,4]^T$ as the best solution.

DOI: 10.1002/oca

Table I. Numerical results for Problem (C) with four switches.

$\mathbf{v}_0$	$\mathbf{v}^*$	J	E_J	E_G
$[4,5,3,5,2]^T$	$[5,5,2,5,4]^T$	6.35559358×10^1		
	$[3,5,2,5,4]^T$	6.35559357×10^1		
	$[5,2,4,5,4]^T$	5.88517382×10^1		
	$[3,2,4,5,4]^T$	5.87626319×10^1		
	$[2,3,4,5,4]^T$	5.87216005×10^1	350	1443
$[5,1,5,1,5]^T$	$[5,2,4,3,5]^T$	6.03626516×10^1		
	$[3,2,4,3,5]^T$	6.02617585×10^1		
	$[2,3,4,3,5]^T$	6.01837944×10^1		
	$[2,4,3,2,5]^T$	6.01491200×10^1		
	$[2,4,5,3,4]^T$	5.88517553×10^1		
	$[2,4,5,4,2]^T$	5.88517431×10^1		
	$[2,3,4,5,4]^T$	5.87216005×10^1	571	2636
$[4,3,1,5,2]^T$	$[3,2,4,5,4]^T$	5.87626319×10^1		
	$[2,3,4,5,4]^T$	5.87216005×10^1	336	1318
$[3,4,4,3,5]^T$	$[2,3,4,3,5]^T$	6.01837944×10^1		
	$[2,4,3,2,5]^T$	6.01491200×10^1		
	$[2,4,5,3,4]^T$	5.88517553×10^1		
	$[2,4,5,4,2]^T$	5.88517431×10^1		
	$[2,3,4,5,4]^T$	5.87216005×10^1	420	1830
$[2,4,5,4,4]^T$	$[2,4,5,4,2]^T$	5.88517431×10^1		
	$[2,3,4,5,4]^T$	5.87216005×10^1	362	1543
$[2,4,5,4,1]^T$	$[2,4,5,4,2]^T$	5.88517431×10^1		
	$[2,3,4,5,4]^T$	5.87216005×10^1	362	1543
$[5,4,3,2,3]^T$	$[2,4,5,4,2]^T$	5.88517431×10^1		
	$[2,3,4,5,4]^T$	5.87216005×10^1	363	1614
$[2,3,2,4,5]^T$	$[2,4,3,5,5]^T$	6.01491301×10^1		
	$[2,4,5,4,2]^T$	5.88517431×10^1		
	$[2,3,4,5,4]^T$	5.87216005×10^1	406	1750
$[4,5,3,4,5]^T$	$[4,5,2,4,5]^T$	6.03626306×10^1		
	$[2,3,4,5,4]^T$	5.87216005×10^1	386	1568
$[2,3,4,5,4]^T$	$[2,3,4,5,4]^T$	5.87216005×10^1	319	1220

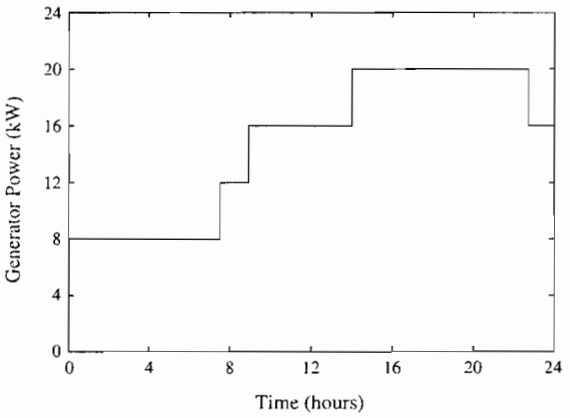

Figure 6. Optimal generator power profile for four switches.

From Table I, most of the local minimizers start with 8 kW, and the generator needs to run at an average of 12 kW to achieve an optimal cost, based on the load demand and PV data. Figure 6 depicts the best operating strategy for the diesel generator: start at a lower load, which is 8 kW for 7.5 h, increase this to 12 kW for another 1.5 h until reaching maximum power at 20 kW, before

DOI: 10.1002/oca

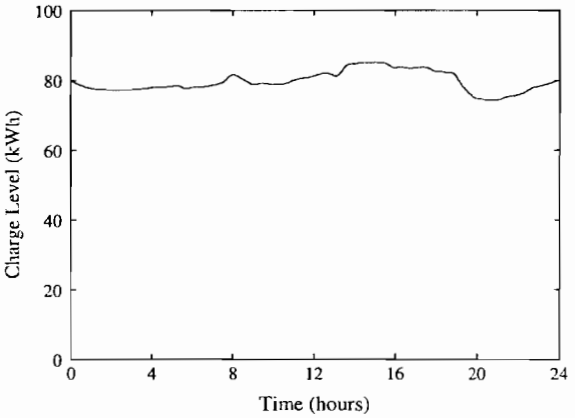

Figure 7. Optimal battery charge profile for four switches.

Table II. Numerical results for Problem (C) with seven switches.

$\mathbf{v_0}$	$\mathbf{v}^*$	J	E_J	E_G
$[3,4,4,3,5,2,3,4]^T$	$[2,3,4,3,5,2,3,4]^T$	6.00318935×10^1		
	$[2,3,4,3,5,2,5,4]^T$	5.87216219×10^1		
	$[2,3,5,3,4,3,5,4]^T$	5.87215981×10^1		
	$[3,4,2,3,4,5,5,4]^T$	5.85886904×10^1		
	$[4,3,2,3,4,5,5,4]^T$	5.85886863×10^1	1733	3825
$[2,3,4,5,2,3,4,5]^T$	$[2,3,4,5,2,3,4,5]^T$	6.01837798×10^1		
	$[2,4,4,5,2,5,4,5]^T$	5.88519036×10^1		
	$[2,4,5,4,2,5,4,5]^T$	5.88518410×10^1		
	$[2,4,4,2,4,5,4,5]^T$	5.88517428×10^1		
	$[3,4,3,2,4,5,4,5]^T$	5.87628543×10^1		
	$[3,5,2,2,4,5,4,5]^T$	5.87626332×10^1		
	$[2,3,5,2,4,5,4,5]^T$	5.87216048×10^1		
	$[4,2,2,3,4,5,4,5]^T$	5.87215986×10^1		
	$[5,2,2,3,4,5,5,4]^T$	5.87215886×10^1		
	$[4,3,2,3,4,5,5,4]^T$	5.85886863×10^1	2550	5717
$[3,4,5,2,3,4,5,2]^T$	$[3,5,5,2,3,4,5,4]^T$	5.87216035×10^1		
	$[5,4,2,3,4,3,5,4]^T$	5.87216030×10^1		
	$[3,4,2,3,4,5,5,4]^T$	5.85886904×10^1		
	$[4,3,2,3,4,5,5,4]^T$	5.85886863×10^1	1659	3605
$[4,5,3,5,2,1,2,3]^T$	$[4,5,2,5,1,1,2,3]^T$	6.53276638×10^1		
	$[5,5,2,5,2,2,3,3]^T$	6.38284739×10^1		
	$[5,3,2,5,4,1,3,2]^T$	6.29162001×10^1		
	$[3,4,2,5,4,5,4,3]^T$	5.92223690×10^1		
	$[4,4,2,3,4,5,4,3]^T$	5.87216018×10^1		
	$[5,3,2,3,4,5,4,2]^T$	5.87215951×10^1		
	$[4,3,2,3,4,5,5,4]^T$	5.85886863×10^1	3115	5923
$[2,1,4,1,5,5,1,3]^T$	$[3,2,4,1,5,5,2,2]^T$	6.01135927×10^1		
	$[3,2,4,3,1,5,4,5]^T$	5.87626426×10^1		
	$[2,3,4,3,1,5,4,5]^T$	5.87216039×10^1		
	$[2,3,4,2,3,5,4,5]^T$	5.87216007×10^1		
	$[2,3,4,2,5,5,4,5]^T$	5.87215952×10^1		
	$[2,3,4,2,5,4,2,5]^T$	5.87215895×10^1		
	$[3,2,5,1,4,5,5,3]^T$	5.87123116×10^1		
	$[3,2,4,3,4,4,5,4]^T$	5.85888513×10^1		
	$[4,3,2,3,4,5,5,4]^T$	5.85886863×10^1	6233	11887

DOI: 10.1002/oca

reducing it to 16 kW. Note that the generator is maintained at a minimum of 12 kW for almost two thirds of the day (16.5 h) to achieve optimal performance. Figure 7 shows that the charge level of the battery bank remains almost constant for the first 8 h, before fluctuating between 75 kWh and 85 kWh for the rest of the day.

4.2. Results for seven switches

We apply the same algorithm to find the optimal switching sequence of Problem (C) for $N = 7$ switches. By using $u(\tau) = 3$, $\tau \in [0, 8]$, for five experiments, Table II indicates that $[4, 3, 2, 3, 4, 5, 5, 4]^T$ is likely to be the global minimizer as it was obtained by using five different initial sequences. Thirty local minimizers were found with the proposed algorithm. Indeed, $[3, 2, 3, 4, 5, 4]^T$ is actually the optimal switching sequence for seven switches when we take into account the optimal u is zero over one interval of its defining partition. The optimal solution is 58.5886863, an improvement of 0.23% compared with the $N = 4$ case. The algorithm terminates when $\mu = 1 \times 10^{-5}$ and $\rho = 1 \times 10^{-2}$.

The plots of the generator output and battery charge level are shown in Figures 8 and 9, respectively. Figure 8 shows that the generator should run at 12 kW for the first 42 min before following the profile of the solution in Figure 6. No significant differences are observed for the battery bank profiles between four switches and seven switches. The computational results indicate that the filled function algorithm is robust and efficient in solving a large scale problem with up to 390,625

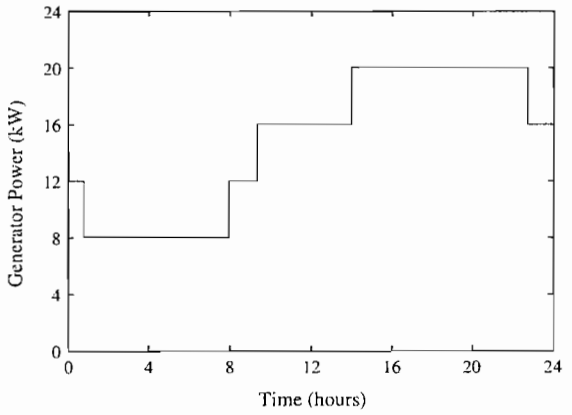

Figure 8. Optimal generator power profile for seven switches.

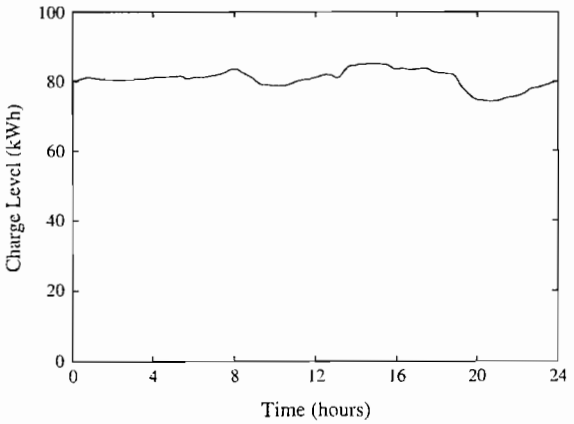

Figure 9. Optimal battery charge profile for seven switches.

DOI: 10.1002/oca

potential switching sequences, given that the objective was evaluated at less than 1.6% of these potential sequences.

4.3. Results for nine switches

Table III depicts the numerical results of solving Problem (C) with $N = 9$ switches, which leads to 9,765,625 possible sequences. Five experiments were carried out using $u(\tau) = 2.4,\ \tau \in [0, 10]$, as the initial guess. Only 0.11% of all potential switching sequences are computed, and the algorithm identifies 40 local minimizers. However, the algorithm fails to identify a unique global minimizer of Problem (C) in this case, and objective function values in the range from 58.2438575 to 58.55886706 are generated. The best solution from Table III is 58.2438575, which is an

Table III. Numerical results for Problem (C) with nine switches.

$\mathbf{v_0}$	$\mathbf{v}^*$	J	E_J	E_G
$[5,1,5,1,5,1,5,1,5,1]^T$	$[5,1,4,1,5,1,5,1,5,1]^T$	7.41514672×10^1		
	$[5,2,4,2,5,1,4,1,5,1]^T$	6.03626730×10^1		
	$[5,4,4,2,5,1,4,1,5,2]^T$	6.01994522×10^1		
	$[4,4,4,2,5,1,4,3,5,3]^T$	5.92987984×10^1		
	$[4,5,4,2,4,2,4,3,5,4]^T$	5.88517493×10^1		
	$[5,4,5,2,4,2,4,3,5,4]^T$	5.88517395×10^1		
	$[4,5,3,2,4,3,4,3,5,4]^T$	5.87626361×10^1		
	$[5,4,3,2,4,3,4,3,5,4]^T$	5.87626283×10^1		
	$[2,5,3,2,4,3,4,5,5,4]^T$	5.87284007×10^1		
	$[3,4,3,2,4,3,4,5,5,4]^T$	5.85886872×10^1		
	$[4,4,3,2,3,4,5,4,5,4]^T$	5.85886768×10^1		
	$[4,3,4,2,3,4,5,4,5,4]^T$	5.85886703×10^1		
	$[5,3,5,2,1,4,5,5,4,3]^T$	5.82438612×10^1	9398	16548
$[3,4,3,4,3,4,3,4,3,4]^T$	$[2,4,5,4,3,4,3,3,2,4]^T$	5.88517547×10^1		
	$[2,4,5,4,3,4,3,3,4,3]^T$	5.88517367×10^1		
	$[2,3,5,4,5,4,1,3,4,1]^T$	5.87216011×10^1		
	$[2,5,3,4,5,4,2,3,3,1]^T$	5.87216006×10^1		
	$[5,2,3,4,5,4,4,2,2,3]^T$	5.87215994×10^1		
	$[5,2,3,4,5,4,4,2,2,5]^T$	5.87215930×10^1		
	$[5,2,3,4,5,4,4,2,4,4]^T$	5.87215911×10^1		
	$[2,3,1,4,5,4,4,1,4,4]^T$	5.82784943×10^1	9380	16177
$[5,4,3,2,5,4,3,2,5,2]^T$	$[5,3,3,2,3,4,3,2,5,4]^T$	5.87215987×10^1		
	$[5,5,2,2,3,4,3,2,5,4]^T$	5.87215868×10^1		
	$[3,4,4,2,3,4,5,2,5,4]^T$	5.85886835×10^1		
	$[4,3,5,2,3,4,5,2,5,4]^T$	5.85886720×10^1		
	$[4,3,4,2,3,4,5,4,5,4]^T$	5.85886703×10^1	6674	11592
$[2,3,4,5,2,3,4,5,2,3]^T$	$[2,3,4,5,3,4,4,5,2,3]^T$	5.91803219×10^1		
	$[2,3,4,5,3,5,5,4,2,3]^T$	5.87225217×10^1		
	$[2,3,4,4,3,5,5,4,2,2]^T$	5.87216606×10^1		
	$[3,5,4,2,3,4,5,4,3,5]^T$	5.85886787×10^1		
	$[4,4,3,2,3,4,5,4,3,5]^T$	5.85886734×10^1		
	$[4,4,3,2,3,4,5,4,3,2]^T$	5.85886723×10^1		
	$[5,5,3,2,1,4,5,4,3,2]^T$	5.82438626×10^1		
	$[5,4,3,2,1,4,5,4,2,3]^T$	5.82438575×10^1	11047	18844
$[2,3,4,5,1,2,3,4,5,2]^T$	$[2,3,4,3,2,3,2,4,5,3]^T$	5.91801836×10^1		
	$[3,2,4,4,3,2,1,4,5,4]^T$	5.87626301×10^1		
	$[2,3,4,4,3,2,1,4,5,4]^T$	5.87216014×10^1		
	$[2,3,4,4,3,4,1,4,5,4]^T$	5.87216012×10^1		
	$[2,3,4,4,2,3,1,4,5,4]^T$	5.87215963×10^1		
	$[3,2,3,4,2,3,1,5,5,4]^T$	5.85886706×10^1	5784	9685

DOI: 10.1002/oca

improvement over the solutions with four and seven switches, by 0.81% and 0.59%, respectively. Clearly, as expected, better solutions are obtained when the number of switches is increased. However, the algorithm appears unable to consistently yield a global solution. This is probably because it cannot guarantee a globally optimal solution of the subproblems.

The characteristics of the generator and battery charge level for the best solution found are plotted in Figures 10 and 11, respectively. In contrast with Figures 6 and 8, where the generator is left running non-stop for 24 h, Figure 10 shows that it is favorable to turn off the generator for 48 min early in the morning to avoid excess energy waste, before re-starting it at 16 kW near 8AM, and increasing the generator output to maximum capacity at 2PM. The suggested operating strategy here is $[3, 2, 1, 4, 5, 4]^T$ (once again, the optimal u was zero over several subintervals of its defining partition).

The findings from Tables I to III support the findings of [2] that diesel generators are inefficient when they operate at a low load factor (around 40% – 50%) of their rated capacity. The findings also indicate that half of the operating time of the generator is spent on generating power during late afternoon and at night when the power source from the PV is not available. In addition, no significant difference is observed for the battery bank profiles among four, seven, and nine switches, where the charge level varies between 75 kWh and 85 kWh. A sharp fall in the battery charge level is also observed when the generator is turned off for a short period, as demonstrated in Figure 11.

Table IV shows the numerical results obtained by solving the transformed problem in [3] starting from 10 random initial guesses. The findings in Table IV, when compared with our algorithm for

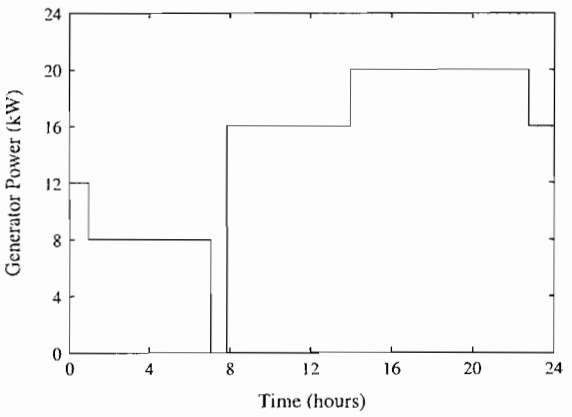

Figure 10. Optimal generator power profile for nine switches.

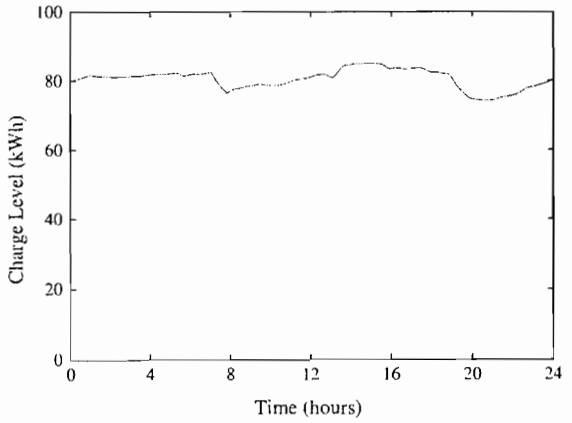

Figure 11. Optimal battery charge profile for nine switches.

DOI: 10.1002/oca

Table IV. Results for solving the model in [3].

Test	Minimum cost function value
1	8.87975339×10^1
2	6.01837893×10^1
3	6.51091281×10^1
4	6.06252775×10^1
5	6.01837944×10^1
6	6.03626756×10^1
7	6.03188550×10^1
8	6.33365059×10^1
9	6.01837982×10^1
10	8.87946446×10^1

four to nine switches, show that our algorithm yields a better result compared with the approach in [3]. The best solution identified by our algorithm is 58.7216005 for four switches, compared with the best local minimum value of 60.1837893 identified from Table IV. Clearly, the method in [3] gets stuck in local minima and cannot determine a globally optimal solution. Note that we are using exactly the same objective function as the one used in [3], including the term penalizing short durations in a particular operating mode.

5. CONCLUSIONS AND FUTURE WORK

A new heuristic approach for solving optimal discrete-valued control problems is proposed. It is based on an efficient time scaling transformation and the incorporation of a discrete filled function algorithm into a standard optimal control software. The computational results for a complex application problem demonstrate that the method is capable of determining a significantly improved solution when compared with earlier results in [3]. Application of the proposed method to other complex discrete-valued optimal control problems is currently being considered.

REFERENCES

1. Maskey RK, Nestmann F. Hydro based renewable hybrid power system for rural electrification: A concept paper. *Retrieved June 12, 2008, from http://www.mtnforum.org/apmn/hybridconcept.htm*, 2008.
2. Wichert B. PV-diesel hybrid energy systems for remote area power generation-a review of current practice and future developments. *Renewable and Sustainable Energy Reviews* 1997; **1**(3):209–228.
3. Ruby T, Rehbock V, Lawrence WB. Optimal control of hybrid power systems. *Dynamics of Continuous, Discrete and Impulsive Systems Series B: Applications & Algorithms* 2003; **10**:429–439.
4. Wu CZ, Teo KL, Rehbock V. A filled function method for optimal discrete-valued control problems. *Journal of Global Optimization* 2008; **44**(2):1–13.
5. Feng ZG, Teo KL. A discrete filled function method for the design of FIR filters with signed-power-of-two coefficients. *IEEE Transactions on Signal Processing* 2008; **56**(1):134–139.
6. Pardalos PM, Yatsenko V. *Optimization and Control of Bilinear Systems: Theory, Algorithms, and Applications*. Springer: New York, 2008.
7. Lee HWJ, Teo KL, Rehbock V, Jennings LS. Control parametrization enhancing technique for optimal discrete-valued control problems. *Automatica* 1999; **35**(8):1401–1407.
8. Loxton RC, Teo KL, Rehbock V, Ling WK. Optimal switching instants for a switched-capacitor DC/DC power converter. *Automatica* 2009; **45**(4):973–980.
9. Ashari M. *Optimisation of photovoltaic/diesel/battery hybrid power system for remote area electrification, Master's Thesis, Department of Electrical and Computer Engineering, Curtin University, Perth, Australia*, 1997.
10. Goya T, Uchida K, Kinjyo Y, Senjyu T, Yona A, Funabashi T. Coordinated control of energy storage system and diesel generator in isolated power system. *International Journal of Emerging Electric Power Systems* 2011; **12**(1):1–22.
11. Jennings LS, Teo KL, Fisher ME, Goh CJ. *MISER3 version 3 Constrained Optimal Control Software, Department of Mathematics and Statistics, University of Western Australia, http://school.maths.uwa.edu.au/~les/miser3.3.html*, 2004.
12. Ng CK, Zhang LS, Li D. Tian WW. Discrete filled function method for discrete global optimization. *Computational Optimization & Applications* 2005; **31**(1):87–115.

DOI: 10.1002/oca

13. Ng CK, Li D, Zhang LS. Discrete global descent method for discrete global optimization and nonlinear integer programming. *Journal of Global Optimization* 2007; **37**(3):357–379.
14. Zhu W. An approximate algorithm for nonlinear integer programming. *Applied Mathematics and Computation* 1998; **93**(2):183–193.
15. Wu ZY, Bai FS, Lee HWJ, Yang YJ. A filled function method for constrained global optimization. *Journal of Global Optimization* 2007; **39**(4):495–507.
16. Zhang LS, Ng CK, Li D, Tian WW. A new filled function method for global optimization. *Journal of Global Optimization* 2004; **28**(1):17–43.
17. Wang W, Shang Y, Zhang L. A filled function method with one parameter for box constrained global optimization. *Applied Mathematics and Computation* 2007; **194**(1):54–66.
18. Xu Z, Huang HX, Pardalos PM, Xu CX. Filled functions for unconstrained global optimization. *Journal of Global Optimization* 2001; **20**(1):49–65.
19. Shang Y, Zhang L. A filled function method for finding a global minimizer on global integer optimization. *Journal of Computational and Applied Mathematics* 2005; **181**(1):200–210.
20. Yang Y, Liang Y. A new discrete filled function algorithm for discrete global optimization. *Journal of Computational and Applied Mathematics* 2007; **202**(2):280–291.
21. Gu YH, Wu ZY. A new filled function method for nonlinear integer programming problem. *Applied Mathematics and Computation* 2006; **173**(2):938–950.

DOI: 10.1002/oca