

Atomic Scheduling of Appliance Energy Consumption in Residential Smart Grid

Kyeong Soo Kim, *Member, IEEE*, Sanghyuk Lee, Tiew On Ting, *Member, IEEE*, and Xin-She Yang

Abstract—The current formulation of the optimal scheduling of appliance energy consumption uses as optimization variables the vectors of appliances' scheduled energy consumption over equally-divided time slots of a day, which does not take into account the *atomicity* of appliances' operations (i.e., the *unsplittable* nature of appliances' operations and resulting energy consumption). In this paper, we provide a new formulation of atomic scheduling of energy consumption based on the *optimal routing* framework; the flow configurations of users over multiple paths between the common source and destination nodes of a ring network are used as optimization variables, which indicate the starting times of scheduled energy consumption, and optimal scheduling problems are now formulated in terms of the user flow configurations. Because the atomic optimal scheduling results in a *Boolean-convex* problem for a convex objective function, we propose a *successive convex relaxation* technique for efficient calculation of an approximate solution, where we iteratively drop fractional-valued elements and apply convex relaxation to the resulting problem until we find a feasible suboptimal solution. Numerical results for the cost and peak-to-average ratio minimization problems demonstrate that the successive convex relaxation technique can provide solutions close to, often identical to, global optimal solutions.

Index Terms—Atomic scheduling, convex relaxation, demand-side management, energy consumption scheduling, optimal routing, smart grid.

I. INTRODUCTION

WE study the problem of scheduling electrical appliance energy consumption in residential smart grid. Our goal in revisiting this well-known problem of energy consumption scheduling (e.g., [1]–[9]) is to establish a new formulation of optimal scheduling problems where we take into account the *atomicity* of operations by household appliances, and to provide efficient solution techniques for the formulated optimal scheduling problems. By atomicity, we mean the *unsplittable* nature of appliances' operations and resulting energy consumption.

Note that the scheduling of electrical appliance energy consumption is a key to the autonomous demand-side management (DSM) for residential smart grid in optimizing energy production and consumption; the scheduling is based on smart meters installed at users' premises and the two-way digital communications between a utility company and users through the smart meters, and the typical goals of DSM includes

consumption reducing and shifting, which lead into lower peak-to-average ratio (PAR) and energy cost [10].

Since the energy consumption scheduling was formulated as an optimization problem using energy consumption scheduling vectors as optimization variables representing appliances' scheduled hourly energy consumption over a day [1], there have been published a number of papers on the subjects of appliance energy consumption scheduling and related billing/pricing mechanisms based on this formulation. For instance, the issue of optimality and fairness in autonomous DSM is studied in relation with billing mechanisms in [4], while the same issue is studied but in the context of user privacy in [6]. The cost and PAR minimization problems, which are separately formulated in [1], are integrated into a PAR-constrained cost minimization problem in [7]. This integration is also further extended to take into account consumers' preference on operation delay and power gap using multiple objective functions. In [9], instead of typical concave n -person games, Rubinstein-Stahl bargaining game model is used to capture the interaction between the supplier and the consumers through a retail price vector in lowering PAR to a certain desired value.

In most existing works on the energy consumption scheduling for autonomous DSM in smart grid, however, no serious attention has been given to the microstructure of scheduled energy consumption over time slots; their major focus is on the optimal value of an objective function that is only based on the aggregate load from the scheduled energy consumption. Few exceptions in this regard include the works on the integration of consumers' preference [7] and the use of load consumption curves in the objective function [2].

Below are some scenarios illustrating the importance of the atomicity of appliance operations.

- When a user washes clothes, the washing machine should be continuously on for a certain period depending on the amount of laundry, e.g., for two hours, during which the operation of the clothes washer cannot be interrupted and the supplied power cannot be reduced arbitrarily. As the washing task can be activated anytime within a specified period, e.g., from 9 AM to 3 PM when the user is out to work, the major goal of autonomous DSM is to determine the optimal two-hour time slot to complete the task when the energy price is lowest during the specified period.
- For heavier-duty tasks like charging the battery of a plug-in hybrid electric vehicle (PHEV), again the continuity of the operation is important, e.g., four hours of uninterrupted charging to maintain the lifetime of PHEV's battery.

K. S. Kim, S. Lee, and T. O. Ting are with the Department of Electrical and Electronic Engineering, Xi'an Jiaotong-Liverpool University, Suzhou 215123, Jiangsu Province, P. R. China (e-mail: {Kyeongsoo.Kim,Sanghyuk.Lee,Toting}@xjtlu.edu.cn).

Xin-She Yang is with the School of Science and Technology, Middlesex University, London NW4 4BT, United Kingdom (e-mail: x.yang@mdx.ac.uk).

- Appliances such as a rice cooker can significantly contribute to the overall cost saving if handled properly. Some rice cookers' function may take more than one hour for completion, e.g., the slow cooking function for delicate soup, whereby the cooking process can be scheduled within any time-slot during the day.

Considering that most of the operations subject to DSM are either atomic as such (e.g., laundry cleaning by a clothes washer) or consist of atomic suboperations (e.g., house heating by a heater operating in the morning and in the evening) [8], therefore, we provide a new formulation of the optimization problem for atomic scheduling of appliance energy consumption and efficient solution techniques for resulting problems in this paper. Atomic scheduling has been mostly discussed in the context of concurrent task scheduling on a multi-processor/core system on a chip (SOC) (e.g., [11]) or transaction processing (e.g., [12]). To the best of our knowledge, our work is the first attempt to formulate atomic scheduling of appliance energy consumption in the autonomous DSM for residential smart grid.

The rest of the paper is organized as follows: In Section II, we review and discuss the issues of the current formulation of appliance energy consumption scheduling based on energy consumption scheduling vectors defined over time slots. In Section III, we describe a new formulation of the appliance energy consumption scheduling based on the optimal routing framework, which guarantees the atomicity of appliance operations, and a successive convex relaxation technique for the efficient solution of the Boolean-convex problem resulting from the new formulation for a given convex objective function. In Section IV, we demonstrate that the performance of successive convex relaxation technique through numerical results for the cost and PAR minimization problems. Section V concludes our work and discusses topics for further study.

II. REVIEW OF CURRENT FORMULATION OF APPLIANCE ENERGY CONSUMPTION SCHEDULING

We first review the current formulation of appliance energy consumption scheduling problem by formally describing it. The formulation described here is largely based on [1] but with some modifications and extensions for clarity and better handling of scheduling intervals over a day boundary. Many notations and definitions in this section are applicable to the formulation of atomic scheduling problems in Section III as well.

Let $\mathcal{N} \triangleq \{1, \dots, N\}$ denotes a set of users in a residential smart grid, whose appliances share a common energy source and subject to autonomous DSM. Without loss of generality and for ease of presentation, we assume that each user has only one appliance throughout the paper.¹ In this case a daily energy consumption scheduling vector of user n is defined as

$$\mathbf{x}_n \triangleq [x_n^0, \dots, x_n^h, \dots, x_n^{H-1}] \quad (1)$$

where a scalar element x_n^h denotes the energy consumption scheduled for a time slot $h \in \mathcal{H} \triangleq \{0, \dots, H-1\}$.² A feasible energy consumption scheduling set for user n is given by

$$\mathcal{X}_n = \left\{ \mathbf{x}_n \left| \sum_{h \in \mathcal{H}_n} x_n^h = E_n, \right. \right. \\ \left. \left. \gamma_n^{\min} \leq x_n^h \leq \gamma_n^{\max}, \forall h \in \mathcal{H}_n, x_n^h = 0, \forall h \in \mathcal{H} \setminus \mathcal{H}_n \right\} \quad (2)$$

where γ_n^{\min} and γ_n^{\max} are the minimum and the maximum energy levels for a time slot, E_n is the total daily energy consumption of user n 's appliance, and \mathcal{H}_n is a *scheduling interval* defined as follows

$$\mathcal{H}_n \triangleq \{h | h = i \bmod H, \forall i \in [\alpha_n, \beta_n]\} \quad (3)$$

with $\alpha_n \in [0, H-1]$, $\beta_n \in [1, 2H-2]$, and $1 \leq \beta_n - \alpha_n \leq H-1$.

With these definitions of scheduling vectors and feasible sets, the optimal scheduling is formulated as an optimization problem for a given objective function (e.g., total energy cost or PAR) as follows:

$$\underset{\mathbf{x}_n \in \mathcal{X}_n, \forall n \in \mathcal{N}}{\text{minimize}} \quad \psi(\mathbf{x}) \quad (4)$$

where

$$\mathbf{x} \triangleq [\mathbf{x}_1, \dots, \mathbf{x}_N]. \quad (5)$$

As shown in [1], it is better to develop a DSM approach that optimizes the properties of the aggregate load of the users rather than individual user's consumption. In fact, the optimal scheduling problems are formulated in terms of the total load across all users at each time slot h in [1], i.e.,

$$L_h(\mathbf{x}) \triangleq \sum_{n \in \mathcal{N}} x_n^h. \quad (6)$$

So (4) can be expressed as

$$\underset{\mathbf{x}_n \in \mathcal{X}_n, \forall n \in \mathcal{N}}{\text{minimize}} \quad \phi(\mathbf{L}(\mathbf{x})) \quad (7)$$

where

$$\mathbf{L}(\mathbf{x}) \triangleq [L_0(\mathbf{x}), \dots, L_{H-1}(\mathbf{x})]. \quad (8)$$

Note that the objective function becomes

$$\phi(\mathbf{L}(\mathbf{x})) = \sum_{h \in \mathcal{H}} C_h(L_h(\mathbf{x})) \quad (9)$$

for the energy cost minimization problem, where $C_h(\cdot)$ is a cost function indicating the cost of generating or distributing electricity energy by the energy source at a time slot h , and

$$\phi(\mathbf{L}(\mathbf{x})) = \frac{H \max_{h \in \mathcal{H}} L_h(\mathbf{x})}{\sum_{n \in \mathcal{N}} E_n} \quad (10)$$

for the PAR minimization problem, respectively.

A major issue with the formulation of optimal scheduling problems based on the energy consumption scheduling vectors in (1) is that it cannot guarantee the atomicity of appliance

¹We use the term "user" and "appliance" interchangeably.

²Time slot numbering in this paper starts from 0, which makes it easier handling energy consumption scheduling wrap-around the day boundary (i.e., from 11 PM to 6 AM) using modulo operation. See (2) and (3) for details.

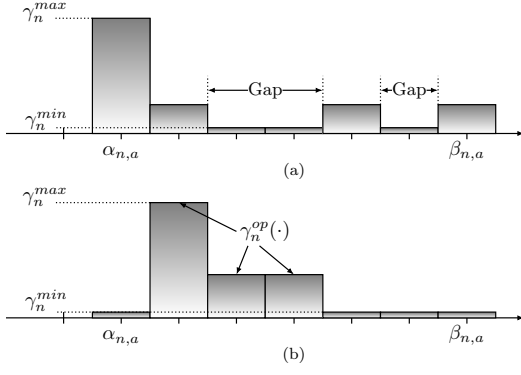


Fig. 1. Examples of (a) non-atomic and (b) atomic scheduling of appliance energy consumption, where γ_n^{\min} , γ_n^{\max} , and $\gamma_n^{\text{op}}(\cdot)$ are the minimum, the maximum, and the operating energy levels of an appliance for a time slot.

operations, which is illustrated in Fig. 1 (a). When a load from other appliances falls on the middle of the scheduling interval, especially that of non-shiftable appliances, the scheduled appliance energy consumption spreads over non-contiguous time slots, resulting in several gaps. Relatedly, the scheduled energy consumption may not provide enough power for appliances to carry out required operations because, during the scheduling, the energy levels over time slots are determined to achieve the optimal solution for the given objective function but not to meet the actual energy consumption requirements of the appliances for the operations. The atomic scheduling that will be described in Section III, on the other hand, assigns contiguous time slots with a predefined pattern of operating energy levels (i.e., $\gamma_n^{\text{op}}(\cdot)$) as shown in Fig. 1 (b), even at the expense of increased penalty in optimization.

III. ATOMIC SCHEDULING OF APPLIANCE ENERGY CONSUMPTION

We assume that all appliance operations are atomic (i.e., no gaps in energy consumption during the operations) and that the operation of the appliance of user n requires δ_n contiguous time slots belonging to a scheduling interval of \mathcal{H}_n with a predefined pattern of operating energy levels³ $\gamma_n^{\text{op}}(h)$ such that $\gamma_n^{\text{op}}(h) > \gamma_n^{\min} \forall h \in [0, \delta_n - 1]$ and $\gamma_n^{\text{op}}(h) = \gamma_n^{\min}$ otherwise. Clearly, $\beta_n \geq \alpha_n + \delta_n - 1$. Without loss of generality, we set γ_n^{\min} to 0; the resulting daily energy consumption E_n , therefore, is given by

$$E_n = \sum_{h=0}^{\delta_n-1} \gamma_n^{\text{op}}(h). \quad (11)$$

Fig. 2 shows an example of operating energy levels.

Note that, if an appliance requires multiple, separate scheduling intervals (e.g., one for 6 AM–11 AM and the other for 1 PM–5 PM), it can be modeled as multiple (virtual) appliances, each of them having only one contiguous scheduling interval (i.e., appliance 1 with 6 AM–11 AM and appliance 2 with 1 PM–5 PM).

³When the time slot duration is an hour (i.e., $H=24$), the energy level becomes a power level.

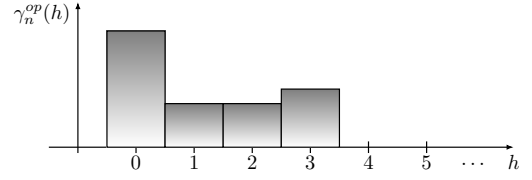


Fig. 2. An example of a predefined pattern of operating energy levels for $\delta_n = 4$.

A simple and straightforward formulation of atomic scheduling is using the starting times of operations as optimization variables, i.e.,

$$\mathbf{s} \triangleq [s_1, \dots, s_N] \quad (12)$$

where a feasible set of starting times for user n is given by

$$\mathcal{S}_n \triangleq \{s_n \mid s_n = i \bmod H, \forall i \in [\alpha_n, \beta_n - \delta_n + 1]\} \quad (13)$$

The total load across all users at each time slot h , therefore, can be expressed in terms of starting times $s_n \in \mathcal{S}_n, \forall n \in \mathcal{N}$, as follows

$$L_h(\mathbf{s}) \triangleq \sum_{n \in \mathcal{N}} \gamma_n^{\text{op}}((h - s_n) \bmod H) I_{\mathcal{R}_n(s_n)}(h) \quad (14)$$

where $\mathcal{R}_n(s_n)$ is a range of user n 's appliance operation for s_n defined by

$$\mathcal{R}_n(s_n) \triangleq \{h \mid h = i \bmod H, \forall i \in [s_n, s_n + \delta_n - 1]\} \quad (15)$$

and $I_{\mathcal{R}_n(s_n)}(h)$ is an indicator function; note that, for a set \mathcal{A} , the indicator function is defined as

$$I_{\mathcal{A}}(a) \triangleq \begin{cases} 1 & \text{if } a \in \mathcal{A}, \\ 0 & \text{otherwise.} \end{cases} \quad (16)$$

Because the feasible set is now discrete, we have to evaluate the objective function for all the elements in the feasible set, which makes the optimization problem impractical for large N and H . For instance, when $N=100$ and $H=24$ with the worst case scenario of $\alpha_n=0, \beta_n=23$, and $\delta_n=1$ for all $n \in \mathcal{N}$, global optimization by direct enumeration would require evaluating the objective function 24^{100} times, which is on the order of 10^{138} times. To address the scalability issue, therefore, we provide an alternative formulation of atomic scheduling based on the optimal routing framework, which is amenable to convex relaxation and enables systematic analysis of suboptimal solutions with upper and lower bounds.

A. Optimal Routing-Based Formulation

We provide a new formulation of the optimal scheduling of appliance energy consumption based on the framework of *optimal routing* in networking [13, Ch. 5]; this new formulation takes into account the atomicity of appliance operations in scheduling without taking any additional objective functions.

As for optimization variables, instead of the energy consumption scheduling vector \mathbf{x}_n in (1), we use *flow configurations of users over multiple paths* between the common source and destination nodes of the network shown in Fig. 3 (a)⁴, which are defined as follows:

⁴This is for the case of hourly time slots, i.e., $H=24$.

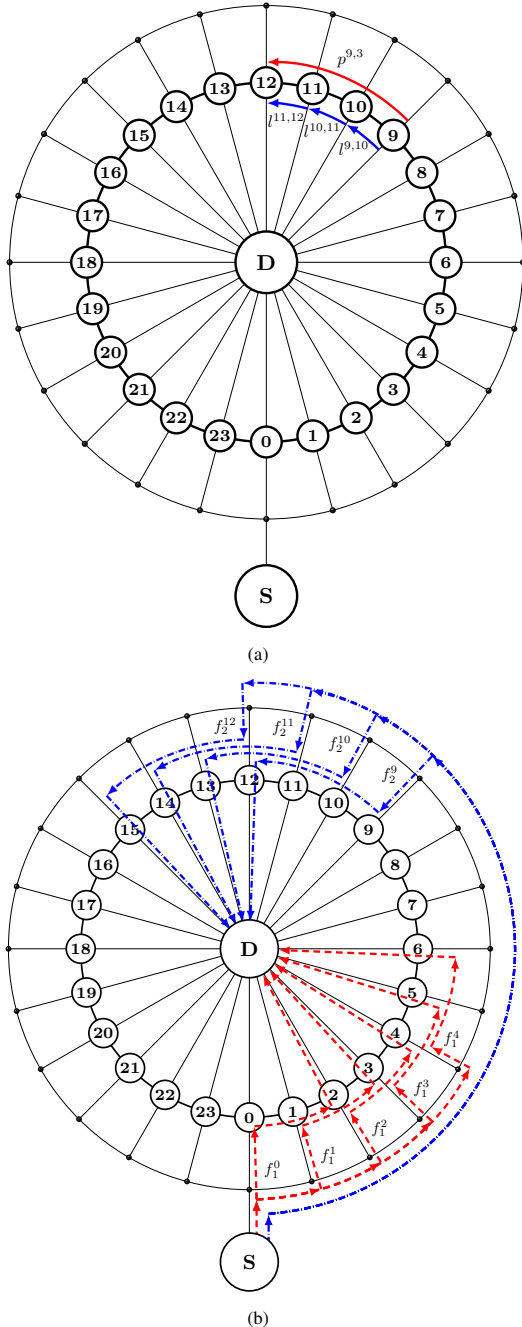


Fig. 3. Atomic energy consumption scheduling based on optimal routing: (a) A network connecting the source (S) and the destination (D) through 24 intermediate nodes with a sample path ($p^{9,3}$) and its constituent links ($l^{9,10}$, $l^{10,11}$, and $l^{11,12}$); (b) mapping of all possible atomic operations of two appliances into two groups of flows (f_1^0, \dots, f_1^4 and f_2^0, \dots, f_2^{12}) over multiple paths on the network.

The users belonging to the set \mathcal{N} share a set \mathcal{P} of paths connecting the source node S and the destination node D, where the path set \mathcal{P} is defined as

$$\mathcal{P} \triangleq \left\{ p^{i,j} \mid i = 0, \dots, H-1, j = 1, \dots, H-1 \right\} \quad (17)$$

where i and j are the starting node number on the ring of a path and the number of hops, respectively; for instance, as illustrated in Fig. 3 (a), $p^{9,3}$ is a three-hop path consisting

of links $l^{9,10}$, $l^{10,11}$, and $l^{11,12}$ on the ring. Note that we do not take into account the radial links connecting the source and the destination nodes to and from the intermediate nodes on the ring in calculating hop counts and path/link costs.⁵ We now define f_n^s as a flow that user n sends on path p^{s,δ_n} , which represents an atomic operation of user n 's appliance with starting time slot s and duration of δ_n slots. Fig. 3 (b) shows mapping of all possible atomic appliance operations of two users (i.e., $\alpha_1=0, \beta_1=5, \delta_1=2$ for user 1 and $\alpha_2=9, \beta_2=14, \delta_2=3$ for user 2) into corresponding groups of flows (i.e., f_1^0, \dots, f_1^4 for user 1 and f_2^0, \dots, f_2^{12} for user 2).

We use the flow configurations of all users as the optimization variables, which are defined as follows:

$$\mathbf{f} \triangleq [f_1, \dots, f_n, \dots, f_N] \quad (18)$$

where

$$f_n \triangleq [f_n^0, \dots, f_n^{H-1}]. \quad (19)$$

A feasible atomic energy consumption scheduling set for user n is given by

$$\mathcal{F}_n = \left\{ \mathbf{f}_n \mid \sum_{s \in \mathcal{S}_n} f_n^s = 1, \right. \\ \left. f_n^s \in \{0, 1\}, \forall s \in \mathcal{S}_n, f_n^s = 0, \forall s \in \mathcal{H} \setminus \mathcal{S}_n \right\} \quad (20)$$

where \mathcal{S}_n is the feasible set of starting times for user n that is already defined in (13) but in terms of s instead of s_n . Note that the constraint of $\sum_{s \in \mathcal{S}_n} f_n^s = 1$ in (20) ensures that an optimal user flow configuration vector \mathbf{f}_n has only one non-zero element representing the starting time of a scheduled atomic operation.

With the user flow configurations, the total load across all users at each time slot $h \in \mathcal{H}$ can be calculated as the sum of all flows passing over the link $l^{h,((h+1) \bmod H)}$, i.e.,

$$L_h(\mathbf{f}) \triangleq \sum_{n \in \mathcal{N}} \gamma_n^{op}((h-s) \bmod H) \left(\sum_{s \in \mathcal{S}_n} f_n^s I_{R_n(s)}(h) \right). \quad (21)$$

Now that we have an expression of the total load at each time slot in terms of flows representing atomic operations of appliances, we can formulate the problem of optimal scheduling of appliance energy consumption for a given objective function of $\phi(\mathbf{L}(\mathbf{f}))$ with $\mathbf{L}(\mathbf{f}) \triangleq [L_0(\mathbf{f}), \dots, L_{H-1}(\mathbf{f})]$ as follows:

$$\underset{\mathbf{f}_n \in \mathcal{F}_n, \forall n \in \mathcal{N}}{\text{minimize}} \quad \phi(\mathbf{L}(\mathbf{f})). \quad (22)$$

For a convex objective function, this problem becomes a *Boolean-convex* problem, since the sum constraint in (20) is linear and the optimization variable f_n^s is restricted to take only 0 or 1 as in [14]. Note that this problem still cannot be solved efficiently but, unlike the problem formulation directly based on starting times in (12), is amenable to the successive convex relaxation technique that we describe in Section III-B.

⁵This means that their link costs are zero.

B. Successive Convex Relaxation

We can obtain the *convex relaxation* of the atomic optimal scheduling problem in (22) by replacing the nonconvex constraints $f_n^s \in \{0, 1\}$ in (20) with the convex constraints $0 \leq f_n^s \leq 1$ as follows:

$$\underset{\mathbf{f}_n \in \hat{\mathcal{F}}_n, \forall n \in \mathcal{N}}{\text{minimize}} \quad \phi(\mathbf{L}(\mathbf{f})) \quad (23)$$

where

$$\hat{\mathcal{F}}_n = \left\{ \mathbf{f}_n \left| \sum_{s \in \mathcal{S}_n} f_n^s = 1, \right. \right. \\ \left. \left. 0 \leq f_n^s \leq 1, \forall s \in \mathcal{S}_n, f_n^s = 0, \forall s \in \mathcal{H} \setminus \mathcal{S}_n \right\}. \quad (24)$$

For a convex objective function, this problem becomes convex because the new feasibility set $\hat{\mathcal{F}}_n$ is now convex where all the equality and inequality constraints on \mathbf{f} are linear. This convex-relaxed problem in (23), therefore, can be solved efficiently, for instance, using the well-known interior-point method [15].

Let $\hat{\mathbf{f}}$ denote a solution to the relaxed problem in (23). The relaxed atomic scheduling problem in (23) is not equivalent to the original problem in (22) because the elements of the optimal solution $\hat{\mathbf{f}}$ can take fractional values (e.g., 0.75). The optimal objective value of the relaxed atomic scheduling problem in (23), however, provides a lower bound on the optimal objective value of the original problem in (22); the optimal value of the relaxed problem is less than or equal to that of the original problem because the feasible set for the relaxed problem contains the feasible set for the original problem.

We can also use the solution of the relaxed problem in (23) to generate a suboptimal solution. Note that we cannot simply choose N largest elements of the optimization vector as is done for the optimal selection of sensor measurements in [14]; because the optimization variable \mathbf{f} is a vector of vectors (i.e., \mathbf{f}_n), the selection of largest elements should be done per component vector. Also, we found that the convex relaxation spreads the starting times of appliance energy consumption over the same scheduling interval when there are several appliances with identical parameter values (i.e., $\alpha_n, \beta_n, \gamma_n^{op}(\cdot)$, and δ_n), which often results in many elements with the same values in the component vectors. If we choose randomly one element among the same-valued ones to break ties in this case, the performance of resulting suboptimal solution is not close to the optimal one.

Therefore, here we present a new technique, called *successive convex relaxation*, where we iteratively drop fractional-valued elements and apply convex relaxation to the resulting problem until we find a feasible solution. This technique is similar to the cutting-plane algorithm in mixed integer linear programming [16, Ch. 9] in that, at each iterative step, new constraints are added to refine the feasible region. The proposed technique, however, is much simpler in forming new constraints where it does not introduce any new slack variables and takes into account the structure of the optimization variables. The detailed procedure is described in

```

1  $\mathcal{D} \leftarrow \emptyset$  /* initialize the set of elements to drop */
2 while true
3    $\hat{\mathbf{f}} \leftarrow$  Solution of (23) with the equality constraints replaced
   by " $f_n^s = 0, \forall s \in \mathcal{H} \setminus \mathcal{S}_n \cup \mathcal{D}$ " in (24)
4   Exclude elements that are the maximum of  $\hat{\mathbf{f}}_n, \forall n \in \mathcal{N}$ ,
   arrange in ascending order the remaining elements of  $\hat{\mathbf{f}}$ 
   whose values are less than one, and let  $i_1, i_2, \dots$  denote
   their indexes
5    $\hat{f}_{i_1} \leftarrow 0$  /* always drop the minimum-valued element */
6    $\mathcal{D} \leftarrow \mathcal{D} \cup i_1$ 
7   for  $j = 2, \dots, N_D$ 
8     if  $\hat{f}_{i_j} < \theta_D$ 
9        $\hat{f}_{i_j} \leftarrow 0$  /* drop it */
10       $\mathcal{D} \leftarrow \mathcal{D} \cup i_j$ 
11   else
12     Break /* return to the while loop */
13   if there is only one nonzero element in  $\hat{\mathbf{f}}_n, \forall n \in \mathcal{N}$ 
14     Break /* stop here; solution found */

```

Algorithm 1: Successive convex relaxation.

Algorithm 1, where we introduce two variables, i.e., θ_D , the threshold value for dropping, and N_D , the maximum number of fractional-valued elements that can be dropped per iteration. Note that the overall complexity of the interior-point method to solve the relaxed convex optimization problem is $\mathcal{O}((NH)^3)$ operations [15], which is needed per iteration and a dominating factor of the complexity of the proposed successive convex relaxation technique. With these variables, therefore, we can do a fine control of the number of fractional-valued elements dropped per iteration and the number of iterations to finish the procedure: A reasonable value of θ_D (e.g., 0.1) can prevent many high fractional-valued elements (e.g., ≥ 0.5) from being dropped unnecessarily per iteration when N_D is set to a rather large number. On the other hand, when a relaxed convex optimization problem gives a solution with a large number of fractional-valued elements smaller than θ_D , we can drop them up to N_D . In this way we can speed up the whole procedure while not sacrificing the quality of the resulting suboptimal solution.

C. Examples

Below we provide specific examples of the optimal atomic scheduling for frequently used objective functions for DSM in residential smart grid.

1) *Energy Cost Minimization:* For energy cost minimization, we can formulate it as the following optimization problem similar to [1]:

$$\underset{\mathbf{f}_n \in \mathcal{F}_n, \forall n \in \mathcal{N}}{\text{minimize}} \quad \sum_{h \in \mathcal{H}} C_h(L_h(\mathbf{f})) \quad (25)$$

where $C_h(\cdot)$ is a cost function indicating the cost of generating or distributing electricity energy by the energy source at a time slot h .

Replacing the feasible set \mathcal{F}_n with $\hat{\mathcal{F}}_n$ in (25), we obtain the relaxed energy cost minimization problem as follows:

$$\underset{\mathbf{f}_n \in \hat{\mathcal{F}}_n, \forall n \in \mathcal{N}}{\text{minimize}} \quad \sum_{h \in \mathcal{H}} C_h(L_h(\mathbf{f})). \quad (26)$$

TABLE I
APPLIANCE ENERGY CONSUMPTION REQUIREMENTS (ADAPTED FROM [1])

Appliance	Parameters			
	α [h]	β [h]	γ^{op} [kWh]	δ [h]
Dish Washer	0	23	0.7200	2
Washing Machine (Energy-Star)	0	23	0.4967	3
Washing Machine (Regular)	0	23	0.6467	3
Clothes Dryer	0	23	0.6250	4
Plug-in Hybrid Electric Vehicle (PHEV)	22*	29*	3.3000	3

* Scheduling interval of 10 PM–5 AM.

2) *Peak-to-Average Ratio Minimization*: Considering the total energy consumption $\sum_{n \in \mathcal{N}} E_n$ is fixed, we can formulate the PAR minimization problem as follows:

$$\underset{\mathbf{f}_n \in \mathcal{F}_n, \forall n \in \mathcal{N}}{\text{minimize}} \quad \max_{h \in \mathcal{H}} (L_h(\mathbf{f})). \quad (27)$$

Note that (27) is difficult to directly solve due to the $\max(\cdot)$ term in the objective function. As mentioned in [1], however, this optimization problem can be turned into a Boolean-linear program, a special case of Boolean-convex optimization, by introducing a new auxiliary variable Γ as follows:

$$\begin{aligned} & \underset{\Gamma, \mathbf{f}_n \in \mathcal{F}_n, \forall n \in \mathcal{N}}{\text{minimize}} \quad \Gamma \\ & \text{subject to} \quad \Gamma \geq L_h(\mathbf{f}), \quad \forall h \in \mathcal{H}. \end{aligned} \quad (28)$$

Again, replacing the feasible set \mathcal{F}_n with $\hat{\mathcal{F}}_n$ in (28), we obtain the relaxed PAR minimization problem, i.e.,

$$\begin{aligned} & \underset{\Gamma, \mathbf{f}_n \in \hat{\mathcal{F}}_n, \forall n \in \mathcal{N}}{\text{minimize}} \quad \Gamma \\ & \text{subject to} \quad \Gamma \geq L_h(\mathbf{f}), \quad \forall h \in \mathcal{H}. \end{aligned} \quad (29)$$

Like the convex optimization, the linear program can be efficiently solved by either the simplex method or the interior-point method [15]. Note that, when applying the successive convex relaxation technique described in Algorithm 1, the auxiliary variable Γ is not subject to dropping unlike \mathbf{f} .

IV. NUMERICAL RESULTS

We demonstrate the application of the successive convex relaxation technique to the atomic scheduling of appliance energy consumption through numerical examples for the minimization of total energy cost and PAR in the aggregated load. In all the examples, a day is divided into 24 time slots (i.e., $H=24$), and each user is to have an appliance randomly selected from the appliances whose energy consumption requirements are summarized in Table I; for simplicity, we assume constant operating energy levels for all appliances.

For the energy cost minimization problem described in Section III-C1, we assume a simple quadratic hourly cost function as in [1], which is given by

$$C_h(L_h) = a_h L_h^2 [\text{cent}] \quad (30)$$

where

$$a_h = \begin{cases} 0.2 & \text{if } h \in [0, 7], \\ 0.3 & \text{if } h \in [8, 23]. \end{cases} \quad (31)$$

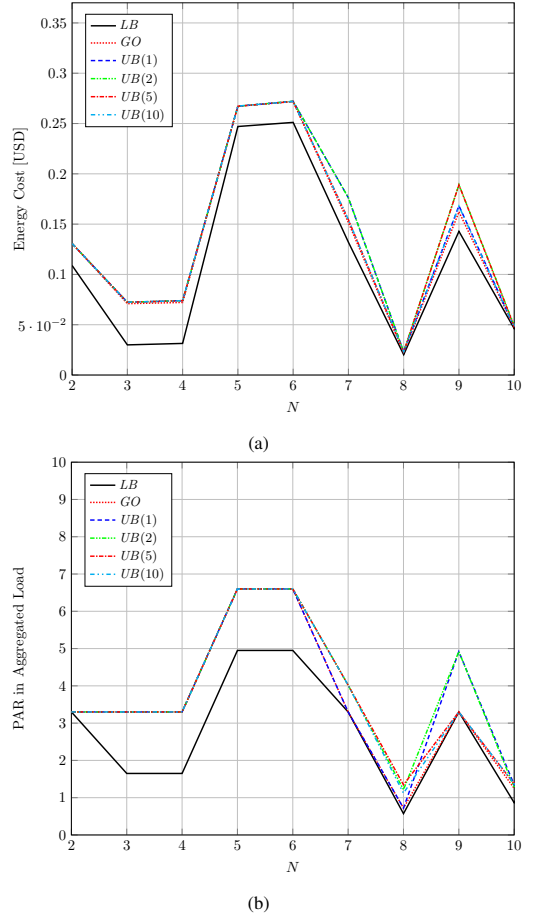


Fig. 4. Comparison of upper (UB) and lower bounds (LB) with true values from global optimization (GO) in atomic energy consumption scheduling for (a) cost minimization and (b) PAR minimization.

The objective function in (26) for this hourly cost function can be expressed as

$$\psi(\mathbf{f}) = \phi(\mathbf{L}(\mathbf{f})) = \sum_{h \in \mathcal{H}} a_h (L_h(\mathbf{f}))^2. \quad (32)$$

Then its first and second derivatives, which are needed for convex optimization, are given as follows: For $n \in \mathcal{N}$ and $s \in \mathcal{H}$, the first derivative (i.e., the *gradient* of $\psi(\mathbf{f})$) is given by

$$\begin{aligned} \frac{\partial \psi(\mathbf{f})}{\partial f_n^s} = & 2 \sum_{h \in \mathcal{H}} a_h \gamma_n^{op}((h-s) \bmod H) L_h(\mathbf{f}) I_{\mathcal{R}_n(s)}(h). \end{aligned} \quad (33)$$

For $n_1, n_2 \in \mathcal{N}$ and $s_1, s_2 \in \mathcal{H}$, the second derivative (i.e., the *Hessian* of $\psi(\mathbf{f})$), which is positive due to the convexity of the objective function in (32), is given by

$$\begin{aligned} \frac{\partial^2 \psi(\mathbf{f})}{\partial f_{n_1}^{s_1} \partial f_{n_2}^{s_2}} = & 2 \sum_{h \in \mathcal{H}} a_h \gamma_{n_1}^{op}((h-s_1) \bmod H) \gamma_{n_2}^{op}((h-s_2) \bmod H) \\ & \times I_{\mathcal{R}_{n_1}(s_1)}(h) I_{\mathcal{R}_{n_2}(s_2)}(h). \end{aligned} \quad (34)$$

To evaluate the performance of the successive convex relaxation technique, we first obtain the lower bound (LB)

using (26) and (29) for the minimization of energy cost and PAR, respectively. Then we obtain suboptimal solutions, which are also upper bounds, using Algorithm 1 with the dropping threshold θ_D fixed to 0.1 and different values of N_D ($UB(N_D)$). We compare the lower and the upper bounds with the optimal objective value from global optimization (GO) using direct enumeration. The results are shown in Fig. 4, where, due to the huge size of the feasible set for direct enumeration given in (20), we limit the maximum value of N to 10; for $N=10$, the size of feasible set is about 2.3×10^{13} , and in case of the energy cost minimization, it took 105.2 hours (i.e., more than 4 days) to obtain the optimal solution from direct enumeration using an OpenMP-based parallelized version of C++ program on a workstation with two Intel® Xeon® processors running at 2.3 GHz providing 20 cores and 40 threads in total, while it took 23.3 seconds to obtain the suboptimal solution from the successive convex relaxation with $N_D=1$ (requiring 207 iterative steps of convex relaxation) using a MATLAB® script with OPTI toolbox [17] on the same machine. Note that due to the random selection of appliances and their different requirements for energy consumption, the resulting energy cost and PAR are not proportional to N .

The results in Fig. 4 show that both lower and upper bounds are very close to the true optimal values. In case of the upper bounds, they are even identical to the true optimal values. For instance, the upper bounds on the energy cost for all values of N_D when $N=2$ and for $N_D=1, 2, 5$ when $N=5$ are identical to true optimal values; in case of PAR, the upper bounds for all values of N_D when $N \leq 6$ are identical to true optimal values. Considering the huge difference in the computational complexity between the two approaches, these results are remarkable. Of the results for energy cost in Fig. 4 (a) and PAR in Fig. 4 (b), we found that the integrality gap in convex relaxation [18] is more visible for the lower bounds on the PAR: Note that, unlike the total cost whose calculation involves all the appliances, the PAR depends on less number of appliances in its calculation, i.e., only those that contribute to a specific time slot whose load is maximum. The relaxation process, however, makes more appliances contribute to the PAR calculation by spreading fractional-valued flows over all possible paths.

To further investigate the quality of lower and upper bounds and the impact of different values of N_D on the performance of successive convex relaxation technique, we obtain the lower and the upper bounds, their gaps defined as the difference between upper and lower bounds, and the number of iterations for upper bounds for the value of N from 2 to 50. Figs. 5 and 6 show the results for energy cost minimization and PAR minimization, respectively.

From the figures, we observe that the overall trend in the results from the minimization of energy cost and PAR are similar to each other, except the relatively larger gaps in the PAR minimization that we already discussed with the results shown in Fig. 4. For both minimization problems, the upper bounds are quite close to the lower bound for a broader range of values of N , and the curves for lower bounds with different values of N_D are hardly distinguishable. The gaps shown in Fig. 5 (b) and Fig. 6 (b) further confirm that the quality of

suboptimal solutions represented by the lower bounds hardly depends on the values of N_D but slightly improves as N increases; even though both upper and lower bound values roughly increase as N increases, the gaps fluctuate but do not show any trend of increasing.

While Fig. 5 (a) and (b) and Fig. 6 (a) and (b) show that N_D does not have any visible impact on the quality of suboptimal solutions, Fig. 5 (c) and Fig. 6 (c) show that increasing the value of N_D can significantly reduce the number of iterations. As discussed in Section III-B, these results show that a larger value of N_D , in combination with a lower dropping threshold value (i.e., $\theta_D = 0.1$), can improve the speed of the successive convex relaxation while maintaining the quality of resulting suboptimal solutions.

V. CONCLUSIONS

The atomicity of appliance operation has never been given serious attention in scheduling of appliance energy consumption in autonomous DSM for residential smart grid. The current dominant approach based on the vector of appliance's hourly energy consumption may result in several gaps in scheduled appliance energy consumption and have problems in providing enough power for appliances to carry out required operations, which, in most cases, are atomic.

In this paper, therefore, we have provided a new formulation of appliance energy consumption scheduling based on the optimal routing framework, which guarantees the atomicity of resulting scheduled energy consumption as such without additional objective functions. Compared to the straightforward problem formulation based on the vector of possible starting times of an appliance operation, the optimal-routing-based formulation provides a Boolean-convex problem for a convex objective function, which is amenable to the successive convex relaxation technique where we can apply well-known interior-point methods for the efficient solution of relaxed convex optimization problem. Unlike approaches based on heuristics like genetic algorithms [19], convex relaxation enables us to carry out systematic analysis of the original problem with both upper and lower bounds.

The numerical results for the cost and PAR minimization problems demonstrate that the proposed successive convex relaxation technique can provide tight upper and lower bounds and, therefore, suboptimal solutions very close to optimal solutions, often identical to true optimal values, in an efficient way. The results also show that, using two control parameters, i.e., N_D for the maximum number of fractional-valued elements that can be dropped per iteration and θ_D for a dropping threshold, we can strike the right balance between the quality of suboptimal solutions and the number of iterations to obtain them in applying the successive convex relaxation technique. Considering that the original problem of atomic scheduling is a very difficult combinatorial problem to solve using direct enumeration due to its huge size of the feasible set, the proposed successive convex relaxation technique makes it practical to implement the atomic scheduling of appliance energy consumption for autonomous DSM in residential smart grid.

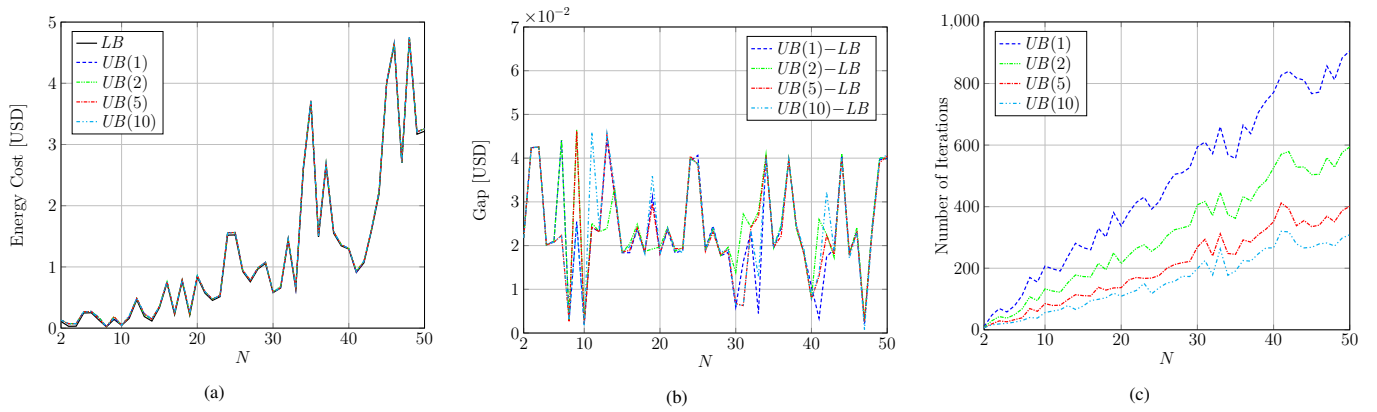


Fig. 5. Atomic energy consumption scheduling for cost minimization: (a) Upper (UB) and lower bounds (LB); (b) gaps; (c) number of iterations.

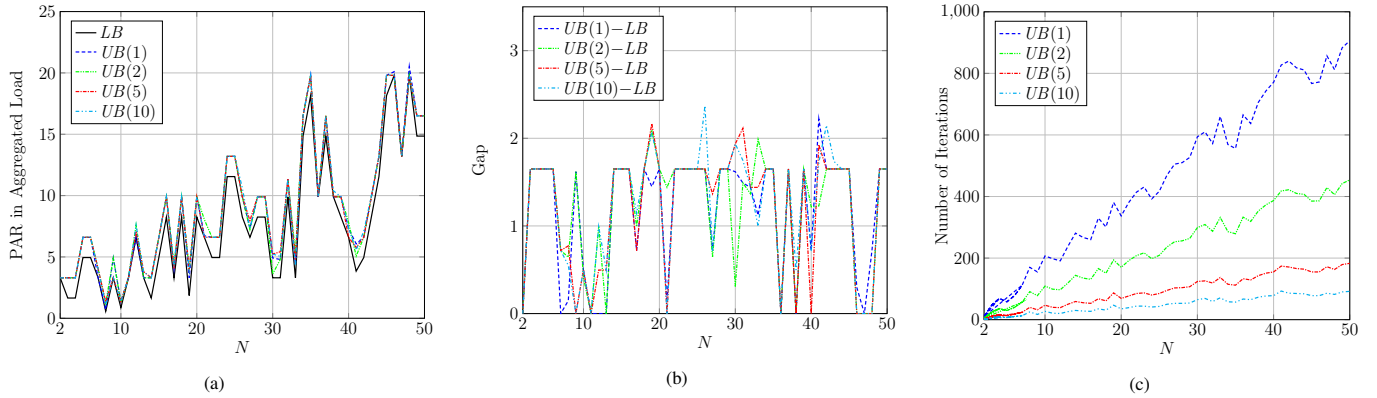


Fig. 6. Atomic energy consumption scheduling for PAR minimization: (a) Upper (UB) and lower bounds (LB); (b) gaps; (c) number of iterations.

Note that our major focus in this paper is on the formulation of atomic scheduling problem and efficient solution techniques based on convex relaxation with adaptive dropping of fractional-valued elements. The extension to distributed atomic energy consumption scheduling and advanced techniques refining the feasible region at each iterative step to reduce the total number of iterations are interesting topics for further study.

ACKNOWLEDGMENT

This work was supported by the Centre for Smart Grid and Information Convergence (CeSGIC) and the Research Development Fund (RDF) of Xi'an Jiaotong-Liverpool University under grant reference number RDF-14-01-25.

REFERENCES

- [1] A.-H. Mohsenian-Rad, V. W. S. Wong, J. Jatskevich, R. Schober, and A. Leon-Garcia, "Autonomous demand-side management based on game-theoretic energy consumption scheduling for the future smart grid," *IEEE Trans. Smart Grid*, vol. 1, no. 3, pp. 320–331, Dec. 2010.
- [2] T. Logenthiran, D. Srinivasan, and T. Z. Shun, "Demand side management in smart grid using heuristic optimization," *IEEE Trans. Smart Grid*, vol. 3, no. 3, pp. 1244–1252, Sep. 2012.
- [3] C. O. Adika and L. Wang, "Autonomous appliance scheduling based on time of use probabilities and load clustering," in *Proc. IPEC 2012*, Dec. 2012, pp. 42–47.
- [4] Z. Baharlouei, M. Hashemi, H. Narimani, and H. Mohsenian-Rad, "Achieving optimality and fairness in autonomous demand response: Benchmarks and billing mechanisms," *IEEE Trans. Smart Grid*, vol. 4, no. 2, pp. 968–975, Jun. 2013.
- [5] X.-M. Hu, Z.-H. Zhan, Y. Lin, Y.-J. Gong, Y. Wei-Jie, Y.-X. Hu, and J. Zhang, "Multiobjective genetic algorithm for demand side management of smart grid," in *Proc. SCIS 2013*, Apr. 2013, pp. 14–21.
- [6] Z. Baharlouei and M. Hashemi, "Efficiency-fairness trade-off in privacy-preserving autonomous demand side management," *IEEE Trans. Smart Grid*, vol. 5, no. 2, pp. 799–808, Mar. 2014.
- [7] Y. Liu, C. Yuen, S. Huang, N. U. Hassan, X. Wang, and S. Xie, "Peak-to-average ratio constrained demand-side management with consumer's preference in residential smart grid," *IEEE J. Sel. Topics Signal Process.*, vol. 8, no. 6, pp. 1084–1097, Dec. 2014.
- [8] M. Khan, N. Javaid, M. Arif, S. Saud, U. Qasim, and Z. Khan, "Peak load scheduling in smart grid communication environment," in *Proc. AINA 2014*, May 2014, pp. 1025–1032.
- [9] Y. Park and S. Kim, "Bargaining-based smart grid pricing model for demand side management scheduling," *ETRI Journal*, vol. 37, no. 1, pp. 197–202, Feb. 2015.
- [10] W. Saad, Z. Han, H. V. Poor, and T. Başar, "Game-theoretic methods for the smart grid: An overview of microgrid systems, demand-side management, and smart grid communications," *IEEE Signal Process. Mag.*, vol. 29, no. 5, pp. 86–105, Sep. 2012.
- [11] P. Yang, C. Wong, P. Marchal, F. Catthoor, D. Desmet, D. Verkest, and R. Lauwereins, "Energy-aware runtime scheduling for embedded-multiprocessor SOCs," *IEEE Design & Test of Computers*, vol. 18, no. 5, pp. 46–58, Sep. 2001.
- [12] J. A. Stankovic and W. Zhao, "On real-time transactions," *ACM SIGMOD Record - Special Issue on Real-Time Database Systems*, vol. 17, no. 1, pp. 4–18, Mar. 1988.
- [13] D. Bertsekas and R. Gallager, *Data networks*, 2nd ed. Upper Saddle River, NJ, USA: Prentice Hall, 1992.
- [14] S. Joshi and S. Boyd, "Sensor selection via convex optimization," *IEEE Trans. Signal Process.*, vol. 57, no. 2, pp. 451–462, Feb. 2009.
- [15] S. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge, U.K.: Cambridge University Press, 2004.
- [16] S. P. Bradley, A. C. Hax, and T. L. Magnanti, *Applied mathematical programming*. Addison-Wesley, Feb. 1977.
- [17] J. Currie and D. I. Wilson, "OPTI: Lowering the barrier between open source optimizers and the industrial MATLAB user," in *Foundations of Computer-Aided Process Operations*, N. Sahinidis and J. Pinto, Eds., Savannah, GA, USA, Jan. 2012.

- [18] E. Chlamtac and M. Tulsiani, “Convex relaxations and integrality gaps,” in *Handbook on Semidefinite, Conic and Polynomial Optimization*, ser. International Series in Operations Research & Management Science, M. F. Anjos and J. B. Lasserre, Eds. Springer US, 2012, vol. 166, pp. 139–169.
- [19] D. E. Goldberg, *Genetic Algorithms in Search, Optimization and Machine Learning*. Reading, Massachusetts: Addison-Wesley, 1989.