

Distribution Loss Minimization With Guaranteed Error Bound

Takeru Inoue, *Member, IEEE*, Keiji Takano, Takayuki Watanabe, Jun Kawahara, Ryo Yoshinaka, Akihiro Kishimoto, Koji Tsuda, Shin-ichi Minato, *Member, IEEE*, and Yasuhiro Hayashi, *Member, IEEE*

Abstract—Determining loss minimum configuration in a distribution network is a hard discrete optimization problem involving many variables. Since more and more dispersed generators are installed on the demand side of power systems and they are reconfigured frequently, developing automatic approaches is indispensable for effectively managing a large-scale distribution network. Existing fast methods employ local updates that gradually improve the loss to solve such an optimization problem. However, they eventually get stuck at local minima, resulting in arbitrarily poor results. In contrast, this paper presents a novel optimization method that provides an error bound on the solution quality. Thus, the obtained solution quality can be evaluated in comparison to the global optimal solution. Instead of using local updates, we construct a highly compressed search space using a binary decision diagram and reduce the optimization problem to a shortest path-finding problem. Our method was shown to be not only accurate but also remarkably efficient; optimization of a large-scale model network with 468 switches was solved in three hours with 1.56% relative error bound.

Index Terms—Distribution network, loss minimization, network reconfiguration, zero-suppressed binary decision diagram.

I. INTRODUCTION

DISTRIBUTION networks consist of several feeders and many switches. They are operated to minimize resistive line losses while satisfying operational constraints on line ca-

Manuscript received August 13, 2012; revised April 14, 2013; accepted October 17, 2013. K. Takano is supported by the FIRST Program, JST CREST and JSPS Kakenhi 25106005. Y. Hayashi is supported by MEXT-Supported Program for the Strategic Research Foundation at Private Universities and JST CREST. Date of current version December 24, 2013. Paper no. TSG-00496-2012.

T. Inoue is with Minato Discrete Structure Manipulation System Project, ERATO, Japan Science and Technology Agency, Sapporo 060-0814, Japan.

K. Takano is with Graduate School of Information Science and Engineering, Tokyo Institute of Technology, Tokyo 226-0026, Japan.

T. Watanabe is with NTT Communications, Tokyo 108-8118, Japan.

J. Kawahara is with Graduate School of Information Science, Nara Institute of Science and Technology, Nara 630-0192, Japan.

R. Yoshinaka is with Graduate School of Informatics, Kyoto University, Kyoto 606-8501, Japan.

A. Kishimoto is with IBM Research, Dublin, Ireland.

K. Tsuda is with Minato Discrete Structure Manipulation System Project, ERATO, Japan Science and Technology Agency, Sapporo, Japan, and also with Computational Biology Research Center, National Institute of Advanced Industrial Science and Technology, Tokyo 135-0064, Japan. (e-mail: koji.tsuda@aist.go.jp).

S.-I. Minato is with Minato Discrete Structure Manipulation System Project, ERATO, Japan Science and Technology Agency, Sapporo, Japan, and also with Graduate School of Information Science and Technology, Hokkaido University, Sapporo, Japan.

Y. Hayashi is with Faculty of Science and Engineering, Waseda University, Tokyo 169-8050, Japan.

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TSG.2013.2288976

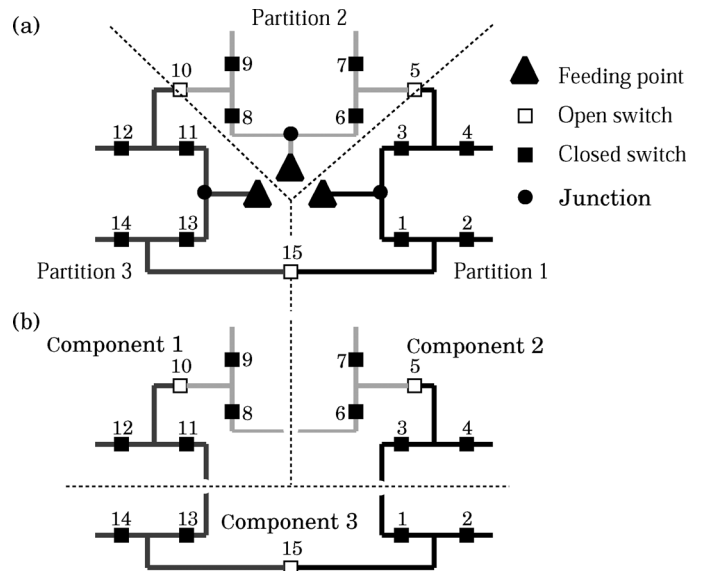


Fig. 1. Distribution network. (a) By configuring switches appropriately, sections are divided to several partitions, each of which is connected to a feeding point. (b) Removing feeding points and first junctions, the network is divided to components.

capacity and voltage drop. As more and more dispersed generators such as fuel cells and solar cells are installed, the reconfiguration of switches would be more frequently needed to avoid violating constraints and to preserve resistive loss within an admissible range. Fig. 1 a shows a typical distribution network. Reconfiguration amounts to optimizing the configuration of switches such that the power loss is minimized. Since each switch configuration is represented as a binary variable (open/closed), this task is formulated as a discrete optimization problem with a set of binary variables. As a result of optimization, the network is divided to several partitions, where a partition represents a set of sections connected to a feeding point through closed switches. Usable configurations must satisfy both *topological* and *electrical* constraints. The topological constraint ensures that each section is connected to only one feeding point, and there is no loop in any partition. The electrical constraint keeps line current and voltage drop within admissible ranges. The loss minimization is a highly complex combinatorial, nondifferentiable, and nonconvex optimization problem with a large number of variables [1], [2].

Several optimization methods have been recently presented to solve this problem. Most of them rely on approximate techniques such as heuristics [1], [3]–[5] and metaheuristics [6]–[9]. Although these methods scale well with large distribution networks, they provide no guarantee on the quality of the solution. That is, because the solution can be arbitrarily worse than the optimal solution, these approaches may fail to reduce the running

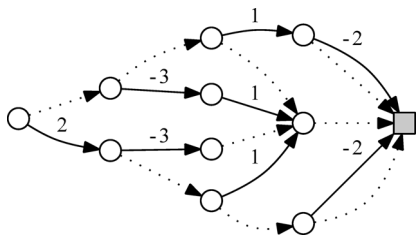


Fig. 2. Binary decision diagram (BDD) corresponding to a set of bit vectors satisfying the constraints (2). Each bit vector is represented as a path from the root node to the terminal. The nodes in a BDD are organized in several levels. Solid and dotted arrows from level i to $i + 1$ indicate $x_i = 1$ and $x_i = 0$, respectively. For example, the path that consists of dotted, solid, solid, and dotted arrows in this order corresponds to $\mathbf{x} = 0110$ and its cost is $0 - 3 + 1 + 0 = -2$. Note that this is the shortest path from the root to the terminal in the BDD.

cost for managing the network. Although the brute force method presented in [10] guarantees optimality, its scalability is currently limited to a network with at most one hundred switches. In contrast, practical networks usually include several hundred switches [1], [11], [12].

Heuristics and metaheuristics employ local update rules of configuration that gradually lead to a smaller loss. Since the search space is discrete and non-convex, they eventually get stuck at local minima. The local minima problem can be solved, however, by organizing the search space in an appropriate way as in our approach. Consider the following example problem with four binary variables,

$$\min_{\mathbf{x}} 2x_1 - 3x_2 + x_3 - 2x_4 \quad (1)$$

subject to the constraints

$$\text{HamDist}(\mathbf{x}, 0000) \leq 2, \text{HamDist}(\mathbf{x}, 0101) \geq 2, \quad (2)$$

where HamDist denotes the Hamming distance. The optimal solution is achieved at $\mathbf{x} = 0110$ with optimal value -2 , but it also has local minima at $\mathbf{x} = 1100$ and $\mathbf{x} = 0011$ with sub-optimal value -1 . This problem can be reduced to a simple shortest path-finding problem by means of a binary decision diagram (BDD) [13], which is a compact data structure that represents a set of bit vectors. The BDD corresponding to the constraints is shown in Fig. 2, where each bit vector satisfying the constraints is represented as a path from the root node to the terminal node. The nodes in a BDD are organized in several levels. Solid and dotted arrows from level i to $i + 1$ indicate $x_i = 1$ and $x_i = 0$, respectively. The main advantage of BDD is that, in certain settings, the BDD size grows only polynomially even if the number of represented bit vectors grows exponentially [13]. Let us assign the coefficients in the objective function (1) to solid arrows as edge weights. Zero weights are assigned to dotted arrows. Now, the optimal solution corresponds to the shortest path from the root node to the terminal and can easily be obtained by invoking search algorithms such as Dijkstra's algorithm.

If BDD is used to solve the loss minimization problem, we must overcome the following two difficulties. First, BDDs representing topological and electrical constraints have to be constructed efficiently. We present novel algorithms for BDD construction specifically designed for electrical networks. Second, since the power loss is not a linear function, more measures are necessary to reduce it to the shortest path-finding problem. When the network is small enough and has only one feeding point, the power loss can be computed for every path in the

BDD and thus the global optimal solution is attained. We will demonstrate in Section V that this simple approach is actually feasible for the 32-bus network introduced by Baran and Wu [4]. For larger networks, however, the distribution network is divided into several *components* [Fig. 1(b)], where the total loss is tightly approximated as the sum of those of individual components. The BDD is transformed into a component-level diagram by aggregating binary variables into categorical variables. Notably, an error bound of our solution can be derived, thus one can always evaluate the quality of our solution in comparison to the global optimal solution. So far, such a guarantee is not available in the other methods except brute-force.

The construction of BDD is performed under a full-blown support of a collection of algorithms implemented in BDD software packages such as CUDD¹ and Buddy². They usually support reduction, reordering of variables and all kinds of binary operations. They are highly optimized via extensive use of cache to prevent unnecessary computation.

In experiments, we use a novel large-scale model network³ developed in 2006 by Fukui University and Tokyo Electric Power Company (TEPCO) [14]. It closely models a typical Japanese distribution network including 72 feeding points and 468 switches. The network consists of residential, industrial, and commercial areas. Each section has a different time-course load profile that is deliberately determined by expert curators. To our best knowledge, there are no benchmark networks that compare to this size and specificity. For example, the benchmark networks by IEEE power and energy society⁴ have at most 12 switches, and those by North Dakota State University⁵ have at most 27 switches. Our experimental results showed remarkable efficiency and reliability of the algorithm; by representing 1.5×10^{70} feasible configurations as a compact BDD, the solution was obtained in less than three hours using one CPU core and the relative error bound was about 1–2%. It implies that our novel BDD-based approach to global optimization can be successfully applied to solve general complex problems.

The rest of this paper is organized as follows. Section II introduces the loss minimization problem. Section III describes algorithms to construct BDDs for topological and electrical constraints. Section IV explains the variable aggregation method for reducing the problem to a shortest path-finding problem. Section V reports our experimental results and Section VI concludes the paper.

II. LOSS MINIMIZATION PROBLEM

This section formulates the loss minimization problem. The distribution network is an undirected graph where vertices are either feeding points, switches or junctions and links are sections. A junction has more than two links attached but it has no switching function. Computing the power loss with respect to a given switch configuration is not a trivial problem, because it requires power flow calculation [15]. Thus, most loss minimization studies (e.g., [4], [7], [16]) employ a simplified flow model

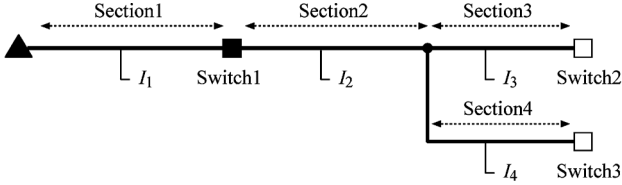
¹<http://vlsi.colorado.edu/~fabio/CUDD/>

²<http://buddy.sourceforge.net/manual/main.html>

³Available from http://www.hayashilab.sci.waseda.ac.jp/RIANT/riant_test_feeder.html

⁴<http://www.ewh.ieee.org/soc/pes/dsacom/testfeeders/>

⁵<http://venus.ece.ndsu.nodak.edu/~kavasseri/reds.html>



Configuration of the three switches is:

$$\mathbf{x} = (1, 0, 0).$$

Sets of downstream sections are:

$$C_1^{\text{down}}(\mathbf{x}) = \{2, 3, 4\}, \quad C_2^{\text{down}}(\mathbf{x}) = \{3, 4\}, \quad C_3^{\text{down}}(\mathbf{x}) = \emptyset, \quad C_4^{\text{down}}(\mathbf{x}) = \emptyset.$$

Line currents are:

$$\begin{aligned} J_1(\mathbf{x}) &= (I_2 + I_3 + I_4) + I_1, & J_2(\mathbf{x}) &= I_3 + I_4 + I_2, & J_3(\mathbf{x}) &= I_3, & J_4(\mathbf{x}) &= I_4. \\ &= J_2(\mathbf{x}), & &= J_3(\mathbf{x}) + J_4(\mathbf{x}) & & & \end{aligned}$$

Fig. 3. An example of the power flow model of [7]. Given a switch configuration $\mathbf{x} = (1, 0, 0)$, it determines downstream sections $C_i^{\text{down}}(\mathbf{x})$ as shown in the figure. Line current $J_i(\mathbf{x})$ is also given by (3). Although section load I_i is assumed as uniformly distributed in the model, it is shown as point load in the figure for readability.

which allows to compute the power loss with relative ease. In this paper, we use the model by Nara *et al.* [7], where each section $i \in \{1, \dots, m\}$ has load I_i , impedance Z_i and resistance R_i . Section load is assumed as uniformly distributed on a section.

The configuration of n switches is described as an n -dimensional binary vector $\mathbf{x} \in \{0, 1\}^n$, where closed switches are denoted as one. Given ℓ feeding points, a valid configuration of switches divides the set of sections into ℓ partitions, each of which is connected exclusively to a feeding point. Additionally, each partition must be loop-free. Once the partition is fixed, the set of upstream sections $C_i^{\text{up}}(\mathbf{x})$ is defined as those on the path from the feeding point to section i (including i). Similarly, the set of downstream sections $C_i^{\text{down}}(\mathbf{x})$ is defined as those farther from section i (excluding i). According to the model by Nara *et al.* [7], the line current J_i at section i is determined as the sum of the downstream load current,

$$J_i(\mathbf{x}) = \sum_{j \in C_i^{\text{down}}(\mathbf{x})} I_j + I_i. \quad (3)$$

We give an example of the line current in Fig. 3. The voltage drop at the end of section i is described as

$$D_i(\mathbf{x}) = \sum_{j \in C_i^{\text{up}}(\mathbf{x})} Z_j \left[\frac{I_j}{2} + \sum_{k \in C_j^{\text{down}}(\mathbf{x})} I_k \right]. \quad (4)$$

Finally, the loss minimization problem is formulated as follows,

$$\min_{\mathbf{x}} \sum_{i=1}^m R_i J_i^2(\mathbf{x}), \quad (5)$$

$$\text{s.t. Configuration } \mathbf{x} \text{ provides valid partitions} \quad (6)$$

$$J_i(\mathbf{x}) \leq J_i^{\text{max}}, D_i(\mathbf{x}) \leq D_i^{\text{max}}, i = 1, \dots, m. \quad (7)$$

Constraints (6) and (7) will be referred to as the *topological constraint* and the *electrical constraint* later on. The electrical constraint ensures that the current and voltage are within admissible limits everywhere.

⁶If the load is represented as power, the load current can be estimated by dividing it by the sending line voltage.

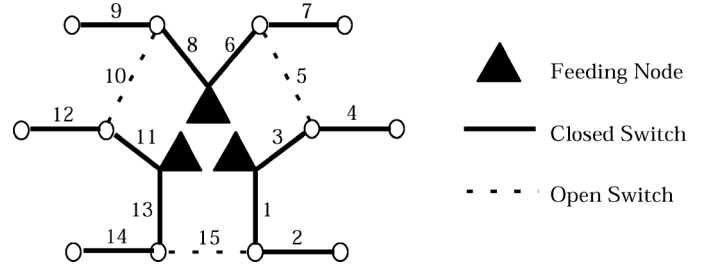


Fig. 4. Dual representation of the distribution network in Fig. 1. Basically, each switch corresponds to an edge and a section is represented as a node. Exceptionally, a set of sections connected with a junction is also represented as a node. A feeding point and adjacent sections are represented as a feeding node.

III. BINARY DECISION DIAGRAMS

A BDD, such as depicted in Fig. 2, is a loopless directed graph with one root node and one terminal node. Each non-terminal node has solid and dotted arrows called one-arc and zero-arc, respectively. A path from the root to the terminal corresponds to a bit vector. An advantage of BDD is that binary operations for two BDDs, such as union and intersection, can be performed without transforming the BDDs into any other data structures. For example, given two BDDs representing $\{\mathbf{x} \mid F(\mathbf{x}) = 1\}$ and $\{\mathbf{x} \mid G(\mathbf{x}) = 1\}$, a new BDD representing $\{\mathbf{x} \mid F(\mathbf{x}) = 1\} \wedge \{\mathbf{x} \mid G(\mathbf{x}) = 1\}$ can be constructed efficiently and directly by the intersection operation [13].

As mentioned in Section I, we reduce the optimization problem to the shortest path-finding problem. As the first step, all bit vectors satisfying the topological constraint are represented as a BDD. All bit vectors satisfying each electrical constraint are also represented as a BDD. The final BDD that contains all bit vectors satisfying all constraints is created via taking an intersection of multiple BDDs using a BDD package.

This section employs a dual representation of the distribution network (Fig. 4). Basically, a switch corresponds to an edge and a section is represented as a node. A set of sections connected by a junction is also represented as a node. A feeding point with all neighboring sections is described as a special node called *feeding node*. If a switch is open, the corresponding edge is removed. Once a configuration of switches is defined, its corresponding subgraph of the original graph is uniquely determined. Since each partition has to be a tree rooted on the feeding node, the topological constraint requires the subgraph to be a rooted spanning forest.

A. Topological Constraint

Let us consider a small graph with two feeding nodes such as depicted upper left in Fig. 5, and assume the order of edges is determined as shown. All rooted spanning forests of this graph can be represented with the *inclusion-exclusion tree* in Fig. 5. One-arcs at level i indicate that the i -th edge is included in the subgraph and zero-arcs indicate exclusion. By looking at the conditions of the edges on a path from the root of a leaf, its corresponding subgraph is determined. The inclusion-exclusion tree is constructed level-by-level. Given the subtree up to level $i - 1$, all candidate subgraphs for level i are created and those with paths between feeding nodes (i.e., shortcuts), loops or unreachable nodes are removed.

Let $h(\mathbf{x})$ denote the Boolean function that returns one if switch configuration \mathbf{x} leads to a rooted spanning forest and zero

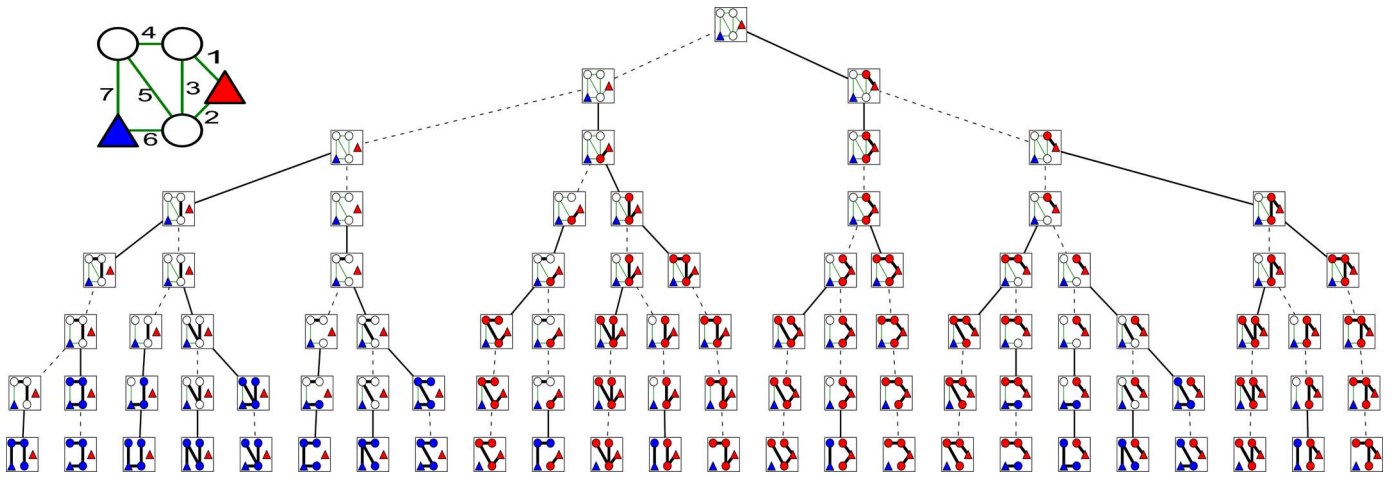


Fig. 5. Inclusion-exclusion tree representing all rooted spanning forests. The solid line (1-arc) at the i -th level indicates the i -th edge is included and the dotted line (0-arc) indicates exclusion. The leaf nodes correspond to all rooted spanning forests of the graph shown above.

otherwise. The inclusion-exclusion tree is an uncompressed representation of all configurations satisfying $h(\mathbf{x}) = 1$. This tree can be compressed to the form of a BDD by merging tree nodes with “equivalent” downstream subtrees into one node. For example, the three nodes highlighted with arrows in Fig. 5 are equivalent and can thus be merged. Internal nodes a and b corresponding to decision paths a_1, \dots, a_s and b_1, \dots, b_s , respectively, are defined to be equivalent iff

$$\begin{aligned} & \{(x_{s+1}, \dots, x_n) \mid h(\mathbf{x}) = 1, x_1 = a_1, \dots, x_s = a_s\} \\ &= \{(x_{s+1}, \dots, x_n) \mid h(\mathbf{x}) = 1, x_1 = b_1, \dots, x_s = b_s\}. \end{aligned}$$

This indicates that a and b has the the same set of downstream decisions after taking paths a_1, \dots, a_s and b_1, \dots, b_s , respectively.

Due to excessive time and memory cost, it is not desirable to construct the whole tree before compression. Thus we need to merge the tree nodes on the fly when new candidates are created at each level. Our approach identifies equivalent subgraphs by looking at the color profile of “frontier nodes.” At the i -th level, edges i, \dots, n are not yet processed. Frontier nodes refer to the nodes adjacent to at least one unprocessed edge. As shown in Fig. 5, the nodes connected to a feeding node via processed edges are distinguished by color. The nodes not yet connected to any feeding node are left uncolored. Interestingly, two subgraphs with the same set of frontier nodes are equivalent if they have the same color profile. When candidate subgraphs for a new level are created, those with shortcuts, loops and unreachable nodes are removed. This removal decision depends only on the color profile of frontier nodes, hence the whole downstream subtree depends only on the profile. By merging equivalent nodes on the fly, a compact BDD is produced in a top-down manner with remarkable efficiency.

Historically, Coudert was the first to construct a BDD representing substructures of a graph [17]. This algorithm was inefficient, because it employed a bottom-up procedure of aggregating small BDDs by binary operations. Recently, Knuth presented a revolutionary top-down path enumeration algorithm, Simpath, based on a similar frontier property [18]. Our algorithm can be seen as an extension of Simpath for rooted spanning forests. To our best knowledge, this extension is novel and plays an indispensable role for the success of loss minimization.

B. Electrical Constraints

The electrical constraint with respect to a feeding point specifies the limit on line current at the feeding point and voltage drop at the leaves of the corresponding partition. These values depend only on the corresponding partition and are irrelevant to the other partitions. Due to this property, a BDD representing each electrical constraint typically includes only a small number of variables.

In constructing the BDD for a feeding point, edges (i.e., switches) are ordered in a breadth-first manner starting from the corresponding feeding node [19]. Then, we start to construct an inclusion-exclusion tree. Given the tree up to the $i - 1$ -th level, candidates for new nodes are created. The candidates violating any electrical constraint are removed at this point. This procedure is valid due to the monotonicity of line current and voltage drop; they only increase when the partition expands. After the whole tree is generated, it is reduced to a BDD using the BDD package.

IV. VARIABLE AGGREGATION

Since the resistive loss is a nonlinear function of the switch configuration \mathbf{x} , we need to transform the final BDD into the search space by aggregating the variables in a *component* of the distribution network. As shown in Fig. 1(b), network components are defined as the connected components of the distribution network after removing the root sections (i.e., sections adjacent to feeding points) and the first junctions (i.e., junctions adjacent to root sections). If the number of components is q , each switch is assigned to one of the subsets M_1, \dots, M_q depending on the component it belongs to. For the example network of Fig. 1(b), $M_1 = \{8, 9, 10, 11, 12\}$, $M_2 = \{3, 4, 5, 6, 7\}$, $M_3 = \{1, 2, 13, 14, 15\}$. Then the configuration vector \mathbf{x} is divided in subvectors $\mathbf{x}_1, \dots, \mathbf{x}_q$ such that \mathbf{x}_i consists of the variables corresponding to the switches in M_i . In addition, let us represent non-root sections of each component as N_1, \dots, N_q , and the set of root sections as U . Using this notation, the objective function (5) is rewritten as

$$f(\mathbf{x}) = \sum_{i \in U} R_i J_i^2(\mathbf{x}) + \sum_{k=1}^q \sum_{i \in N_k} R_i J_i^2(\mathbf{x}_k). \quad (8)$$

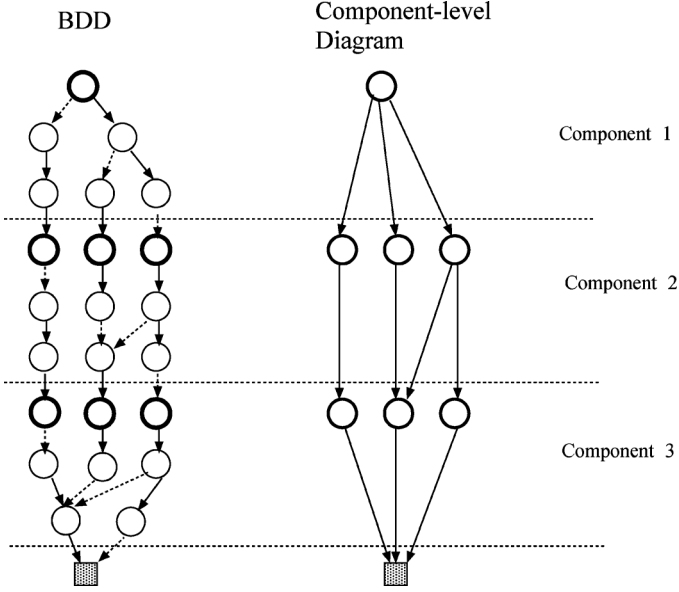


Fig. 6. Variable aggregation. Boundary nodes are shown as bold circles. In the component-level diagram, two nodes are connected if there is a path between them in the BDD.

Importantly, the loss at a non-root section depends only on the configuration of switches in its component, because the line current at a section depends only on downstream sections of that section.

A. Approximation

Minimizing $f(\mathbf{x})$ is extremely difficult due to global dependencies of J_i at root sections. However, when root sections are ignored as

$$f^*(\mathbf{x}) = \sum_{k=1}^q \sum_{i \in N_k} R_i J_i^2(\mathbf{x}_k),$$

the global optimal solution \mathbf{x}^* minimizing $f^*(\mathbf{x})$ under the topological and electric constraints can be obtained by the following procedure of variable aggregation. Let us define the aggregated categorical variable for \mathbf{x}_k as $v_k \in \{0, \dots, 2^{|M_k|} - 1\}$. Although v_k might take $2^{|M_k|}$ different values in the worst scenario, the domain is much smaller in most cases due to topological and electric constraints.

A component-level diagram is created as follows. First, the BDD is rearranged such that the variables in a component are aligned next to each other. The set of nodes located at the top level in each component is called *boundary nodes* (Fig. 6). As the first step, only the boundary nodes are copied to the component-level diagram. Edges between these nodes are then created by enumerating all BDD paths between the boundary nodes. More specifically, if there is a path in BDD, an edge is created in the component-level diagram. A BDD path from component k to $k+1$ specifies a configuration of switches for v_k in the k -th component. We compute the total loss in the component corresponding to the BDD path and assign it as the weight of the new edge in the component-level diagram. If there are multiple BDD paths, the minimum loss is taken as the weight. Finally, the optimal solution is obtained as the shortest path from the root node to the terminal in the component-level diagram.

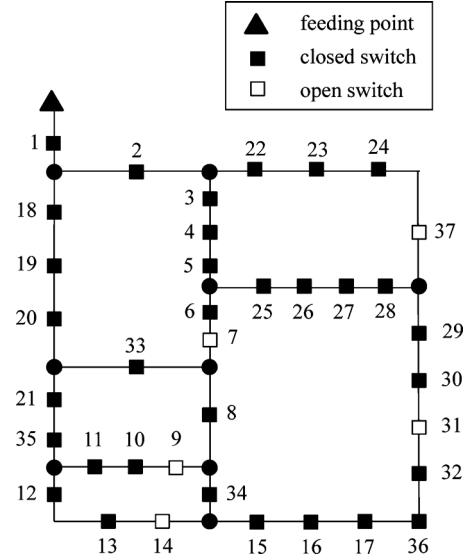


Fig. 7. The 32-bus network by Baran and Wu. Optimal switch configuration determined by our method is shown here.

B. Error Bound

Since the above solution is merely the global optimal solution to an approximated problem, the achieved loss for the original problem is suboptimal, i.e., $f(\mathbf{x}^*) \geq f_{opt}$. Nevertheless, an error bound $f(\mathbf{x}^*) - f_{opt} \leq \epsilon$ can be derived theoretically. Let us revise the original optimization problem as follows: Minimize (5) subject to (6), (7) and a new constraint

$$\sum_{i \in U} J_i(\mathbf{x}) = \sum_{i=1}^n I_i. \quad (9)$$

It indicates that the sum of line current at the root sections is equal to the sum of load of all non-root sections. Since this constraint holds for any \mathbf{x} satisfying the topological constraint, the optimal solution remains unchanged even after introducing the new constraint. Now, assume that $J_i(\mathbf{x})$ is not a function of \mathbf{x} and is regarded as a new free variable J_i ,

$$f_{relax} = \min_{\mathbf{x}, \{J_i \mid i \in U\}} \sum_{i \in U} R_i J_i^2 + \sum_{k=1}^q \sum_{i \in N_k} R_i J_i^2(\mathbf{x}_k), \quad (10)$$

subject to (6), (7) and (9). Here the optimal solution to f_{relax} is \mathbf{x}^* and

$$J_i^* = \frac{\sum_{j=1}^m I_j}{\left(R_i \sum_{j \in U} \frac{1}{R_j} \right)}.$$

It always holds $f_{relax} \leq f_{opt}$ because f_{relax} relaxes the original problem. The error bound is therefore finally obtained as $\epsilon = f(\mathbf{x}^*) - f_{relax}$. Similarly, the relative error of \mathbf{x}^* is bounded as

$$\frac{f(\mathbf{x}^*) - f_{opt}}{f(\mathbf{x}^*)} \leq \frac{f(\mathbf{x}^*) - f_{relax}}{f(\mathbf{x}^*)}. \quad (11)$$

V. EXPERIMENTS

In this section, our method is applied to two model networks. The experiments were conducted using a single core in Intel Xeon CPU E31290 (3.60 GHz).

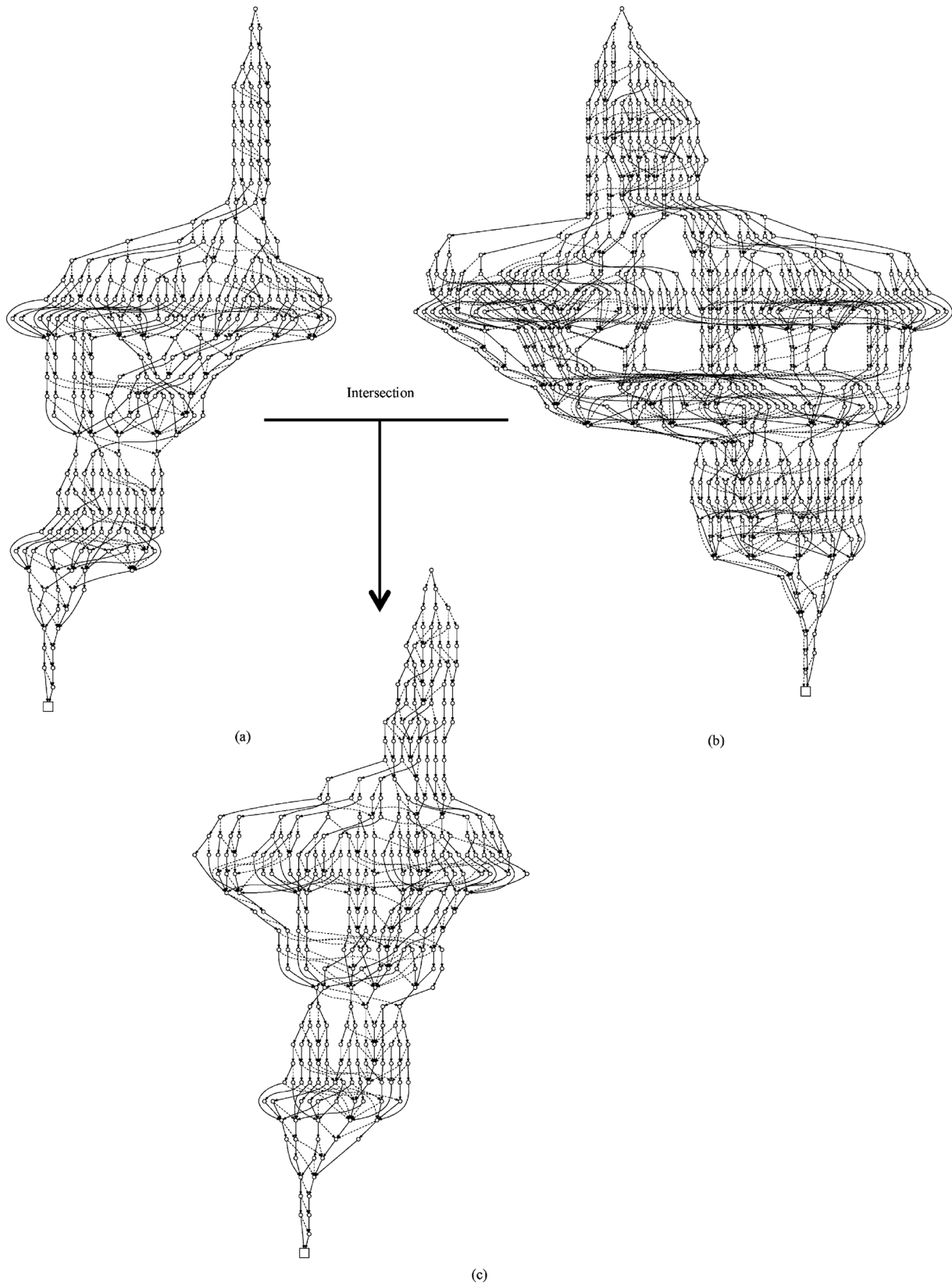


Fig. 8. BDDs for the 32-bus network. Each BDD has 37 levels corresponding to switches in the network. (a) BDD for topology constraint. (b) BDD for electrical constraints. (c) BDD for all constraints.

A. 32-Bus Network

The first model network is the 32-bus network introduced by Baran and Wu [4] (Fig. 7). It has 37 switches and only one

feeding point. There is one load profile presented for the network. The sending line voltage is 12.66 kV, and the maximum voltage drop must be within 10% of the sending voltage. This model does not have line current constraints, so the minimum

TABLE I
NUMBER OF FEASIBLE CONFIGURATIONS FOR THE 32-BUS NETWORK

Constraints	Number of feasible configurations	BDD size
Topological	50751	316
Electrical	68497224119	596
Both	50728	360

TABLE II
COMPARISON OF POWER LOSS TO THE REPORTED RESULTS FOR THE 32-BUS NETWORK

Configuration	Power loss [kW]	Open switches
Initial	355.34	33, 34, 35, 36, 37
Reported in [20], [21], [22]	231.44	7, 9, 14, 32, 37
Optimal Solution	226.75	7, 9, 14, 31, 37

TABLE III
SPECIFICATION OF THE FUKUI-TEPCO NETWORK

Number of feeders	72
Number of switches	468
Number of sections	648
Total load, $\sum_{i=1}^n I_i$	287 MW at 2 p.m., 113 MW at 4 a.m.
Line capacity, J^{\max}	300 A
Sending line voltage	6.6 kV
Maximum voltage drop, D^{\max}	0.3 kV

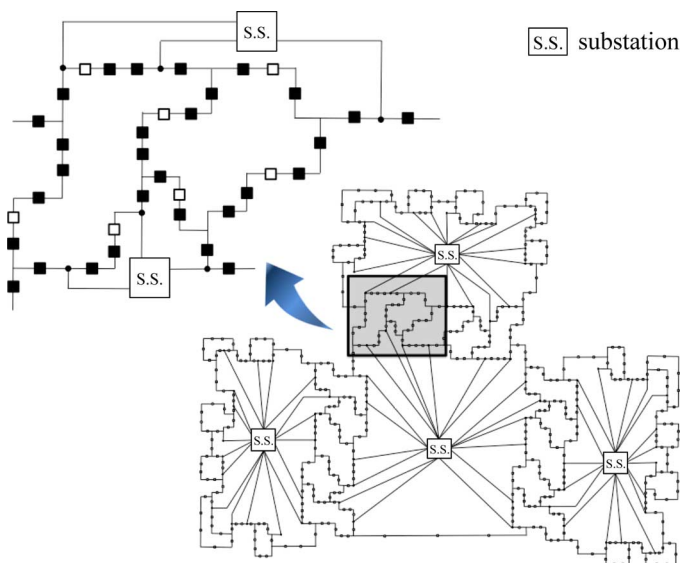


Fig. 9. Switch configuration of the Fukui-TEPCO network determined by our method for the load profile at 4 A.M.

loss configuration is searched for under the voltage constraint only.

The number of all possible configurations is 2^{37} , but the number of feasible configurations satisfying the topological and electrical constraints is much smaller. As discussed in Section III, we created two BDDs representing each constraint and then combined them into one BDD via the intersection operation (Fig. 8). Construction of topological and electrical BDDs took 0.14 second and 15 692 seconds (about 4 hours and 20 minutes), respectively. This huge difference is due to the efficient top-down algorithm for topological BDD. The intersection operation took only 0.018 second, thus almost negligible. As shown in Table I, the number of feasible solutions turned out to be only 50 728. It allowed us to compute the power loss for all feasible solutions in 167 seconds and select the global optimal solution. The variable aggregation

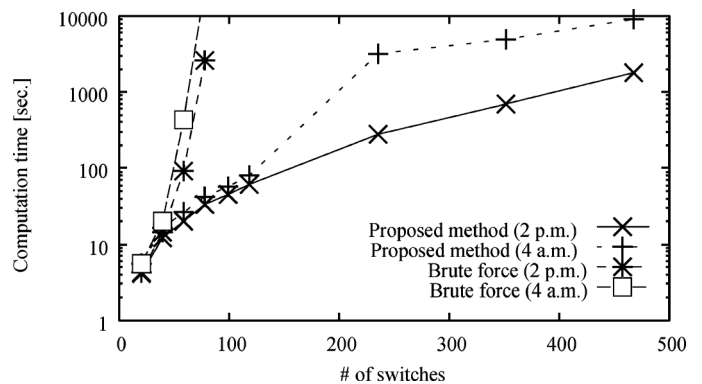


Fig. 10. Computation time of the proposed method and the brute force method.

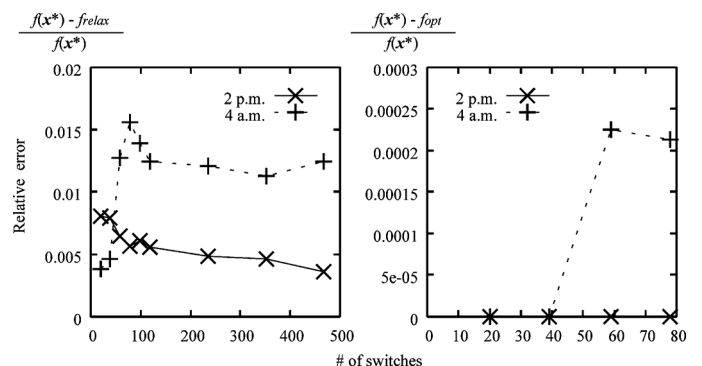


Fig. 11. (Left) Relative error bound of the solutions. (Right) True relative error of the solutions up to 78 switches.

procedure in Section IV was not necessary, because it has only one component.

Table II summarizes the power loss and the solution obtained by our method. It is observed that our solution was slightly better than that of existing methods [20]–[22]. The difference, however, was not significant. It may be due to different flow models they employ and their solution might have been the global optimal solution in terms of their model. The important point is rather that our solution is proven to be global optimal in terms of our flow model. In general, heuristic algorithms can only return solutions that are probably global optimal but have no theoretical guarantee on global optimality.

B. Fukui-TEPCO Network

As mentioned in Section I, we employ the Fukui-TEPCO network including 72 feeders and 468 switches (Table III). The network has 63 components. The number of switches in a component is 7.43 on average, while the minimum and maximum are 3 and 20, respectively. We also generated subsampled versions of the network containing 20, 39, 59, 78, 99, 118, 235, and 352 switches. Among hourly load profiles, the peak load at 2 P.M. and the baseline load at 4 A.M. were used in the experiments.

Fig. 10 shows the computation time of our method and that of the brute force method for networks of different sizes and two time points (2 P.M. and 4 A.M.). Fig. 11, left, plots the relative error bound (11) of our solutions. Our solution for 4 A.M. is partly visualized in Fig. 9. Our method finished the whole optimization procedure in less than three hours for the full network with 468 switches and the relative error bound was 1.56% at most. The power loss of our solutions at 2 P.M. and 4 A.M. was 2 507 336 W and 557 660 W, which amounts to 0.874% and

TABLE IV
NUMBERS OF FEASIBLE CONFIGURATIONS IN A NETWORK OF 468 SWITCHES

Condition	Number of feasible configurations	BDD size
All constraints, loads at 2 p.m.	56549012847446003723757714431732193815091620755492933270200	3223985
All constraints, loads at 4 a.m.	15052768726188994695810341375588783632354554002783638970270450179200000	3171
Topological constraint	218646889093444243387855355581579747968214496454992053728787429330078125	1554

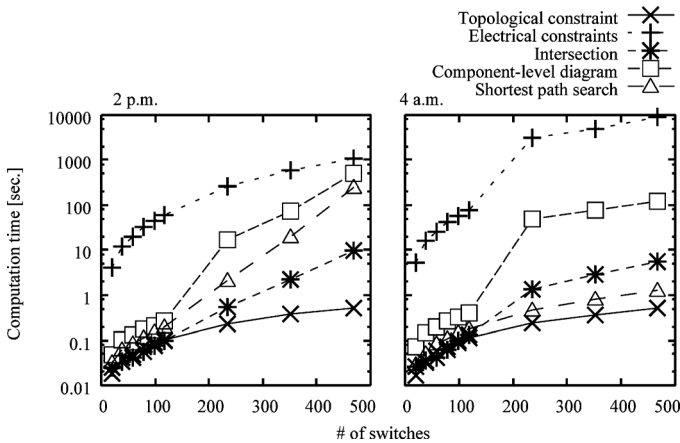


Fig. 12. Computation time of each process in the proposed method.

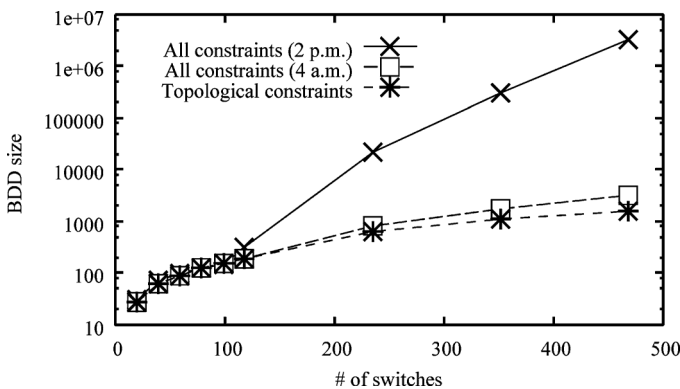


Fig. 13. BDD size for the constraints.

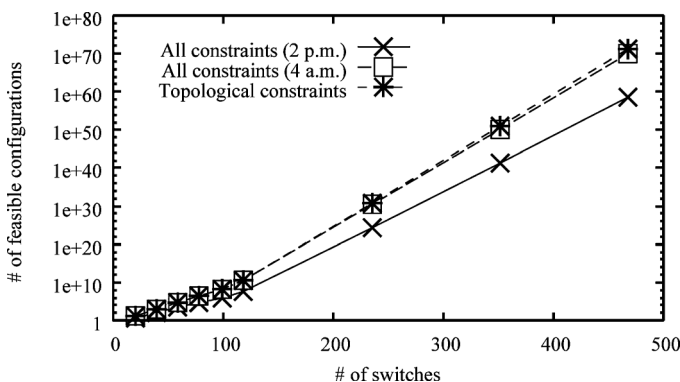


Fig. 14. Number of feasible configurations.

0.494% of the total load, respectively. As expected, the computational time of the brute-force approach exploded quickly. With a time limit of 10 000 seconds, the global optimal solution was obtained only up to 78 switches. Fig. 11, right, shows the true relative error of our solutions up to 78 switches. In many cases, our solution was indeed optimal (i.e., the relative error was zero). The maximum relative error was about 0.0225%; much smaller than the theoretical bound.

Fig. 12 shows the computation time required by each process. The construction of BDDs for electrical constraints takes the largest fraction of the computation time. This is because large inclusion-exclusion trees are produced without on-the-fly merging, and computing voltage drop for each subgraph is time-consuming. In contrast, the construction for the topological constraint finished in less than one second even for up to 468 switches, clearly indicating the significant importance of on-the-fly merging. The BDD construction for electrical constraints took significantly more time for the 4 A.M. time point; Since the section loads were smaller at night, a larger inclusion-exclusion tree had to be explored to reach the line current capacity.

The number of nodes in the BDDs for the constraints are shown in Fig. 13. Since a single BDD node requires about 32 bytes, this figure indicates that the amount of memory required to store the BDD was about 100 MB and 100 KB for 2 P.M. and 4 A.M., respectively. Given a BDD, it is easy to count the number of represented bit vectors by a recursive algorithm [23]. We therefore computed the number of feasible configurations for the topological constraint and all constraints combined. The results are shown in Fig. 14 and Table IV. Interestingly, such a large number of feasible configurations is compressed to a small BDD. Larger BDDs do not necessarily represent larger numbers of bit vectors. Intuitively, the BDD size gets smaller if the set of bit vectors has some kind of regularity. In general, however, it is hard to predict the size of BDDs in advance for a given problem.

VI. CONCLUSION

In this paper, we have developed an efficient network reconfiguration method that yields an error bound. In the real-world settings, heuristic algorithms without solution quality guarantee are unlikely to be used because of the danger of returning solutions incurring an excessive running cost for managing the network. It may be possible to reduce the probability of such a catastrophic failure caused by local-update-based methods, e.g., by using multiple starting points. However, since distribution networks are fundamental infrastructure, failures cannot be allowed even with the smallest probability. Although global optimal solutions cannot always be attained in large-scale networks, users are at least notified of a bound of power loss due to the configuration our algorithm returns. In contrast, such bounds cannot be estimated for the configurations generated by heuristic methods. In this respect, we believe that this work is an important step towards automated network reconfiguration.

ACKNOWLEDGMENT

The authors would like to thank T. Horiyama, T. Uno, H. Takano, M. Ishihata, and D. duVerle for their valuable comments. A. Kishimoto's work was performed while he was affiliated with Graduate School of Information Science and Engineering, Tokyo Institute of Technology, Japan.

REFERENCES

- [1] D. Shirmohammadi and H. Hong, "Reconfiguration of electric distribution networks for resistive line losses reduction," *IEEE Trans. Power Del.*, vol. 4, no. 2, pp. 1492–1498, 1989.
- [2] E. Carreno, R. Romero, and A. Padilha-Feltrin, "An efficient codification to solve distribution network reconfiguration for loss reduction problem," *IEEE Trans. Power Syst.*, vol. 23, no. 4, pp. 1542–1551, 2008.
- [3] S. Civanlar, J. Grainger, H. Yin, and S. Lee, "Distribution feeder reconfiguration for loss reduction," *IEEE Trans. Power Del.*, vol. 3, no. 3, pp. 1217–1223, 1988.
- [4] M. Baran and F. Wu, "Network reconfiguration in distribution systems for loss reduction and load balancing," *IEEE Trans. Power Del.*, vol. 4, no. 2, pp. 1401–1407, 1989.
- [5] W.-M. Lin and H.-C. Chin, "A new approach for distribution feeder reconfiguration for loss reduction and service restoration," *IEEE Trans. Power Del.*, vol. 13, no. 3, pp. 870–875, 1998.
- [6] H.-D. Chiang and R. Jean-Jumeau, "Optimal network reconfigurations in distribution systems. I. A new formulation and a solution methodology," *IEEE Trans. Power Del.*, vol. 5, no. 4, pp. 1902–1909, 1990.
- [7] K. Nara, A. Shiose, M. Kitagawa, and T. Ishihara, "Implementation of genetic algorithm for distribution systems loss minimum re-configuration," *IEEE Trans. Power Syst.*, vol. 7, no. 3, pp. 1044–1051, 1992.
- [8] Y. Hayashi and J. Matsuki, "Loss minimum configuration of distribution system considering $n - 1$ security of dispersed generators," *IEEE Trans. Power Syst.*, vol. 19, no. 1, pp. 636–642, 2004.
- [9] B. Enacheanu, B. Raison, R. Caire, O. Devaux, W. Bienia, and N. Hadjsaid, "Radial network reconfiguration using genetic algorithm based on the matroid theory," *IEEE Trans. Power Syst.*, vol. 23, no. 1, pp. 186–195, 2008.
- [10] A. Morton and I. Mareels, "An efficient brute-force solution to the network reconfiguration problem," *IEEE Trans. Power Del.*, vol. 15, no. 3, pp. 996–1000, 2000.
- [11] T. McDermott, I. Drezga, and R. Broadwater, "A heuristic nonlinear constructive method for distribution system reconfiguration," *IEEE Trans. Power Syst.*, vol. 14, no. 2, pp. 478–483, 1999.
- [12] Y.-J. Jeon, J.-C. Kim, J.-O. Kim, J.-R. Shin, and K. Lee, "An efficient simulated annealing algorithm for network reconfiguration in large-scale distribution systems," *IEEE Trans. Power Del.*, vol. 17, no. 4, pp. 1070–1078, 2002.
- [13] S. Minato, "Zero-suppressed BDDs for set manipulation in combinatorial problems," in *Proc. Conf. Design Autom.*, 1993, pp. 272–277.
- [14] Y. Hayashi, S. Kawasaki, J. Matsuki, H. Matsuda, S. Sakai, T. Miyazaki, and N. Kobayashi, "Establishment of a standard analytical model of distribution network with distributed generators and development of multi evaluation method for network configuration candidates," (in Japanese) *IEEJ Trans. Power Energy*, vol. 126, no. 10, pp. 1013–1022, 2006.
- [15] J. Grainger and W. Stevenson, *Power System Analysis*. New York: McGraw-Hill, 1994.
- [16] D. Shirmohammadi, H. W. Hong, and G. X. Luo, "A compensation-based power flow method for weakly meshed distribution and transmission networks," *IEEE Trans. Power Syst.*, vol. 3, no. 2, pp. 753–762, 1988.
- [17] O. Coudert, "Solving graph optimization problems with ZBDDs," in *Proc. Eur. Design Test Conf.*, 1997, pp. 224–228.
- [18] D. Knuth, *The Art of Computer Programming, volume 4, Fascicle 1: Bitwise Tricks and Techniques; Binary Decision Diagrams*. Reading, MA, USA: Addison-Wesley, 2009.
- [19] J. Kleinberg and E. Tardos, *Algorithm Design*. Reading, MA, USA: Pearson Education, 2006.
- [20] G. Raju and P. Bijwe, "An efficient algorithm for minimum loss reconfiguration of distribution system based on sensitivity and heuristics," *IEEE Trans. Power Syst.*, vol. 23, no. 3, pp. 1280–1287, 2008.
- [21] F. Gomes, J. Carneiro, S. J. L. R. Pereira, M. Vinagre, P. Garcia, and L. R. de Araujo, "A new distribution system reconfiguration approach using optimum power flow and sensitivity analysis for loss reduction," *IEEE Trans. Power Syst.*, vol. 21, no. 4, pp. 1616–1623, Nov. .
- [22] E. Ramos, A. Exposito, J. Santos, and F. Iborra, "Path-based distribution network modeling: Application to reconfiguration for loss reduction," *IEEE Trans. Power Syst.*, vol. 20, no. 2, pp. 556–564, May 2005.
- [23] R. Bryant, "Graph-based algorithms for Boolean function manipulation," *IEEE Trans. Comput.*, vol. C-35, no. 8, pp. 677–691, 1986.



Takeru Inoue received the B.E., M.E., and Ph.D. degrees from Kyoto University, Kyoto, Japan, in 1998, 2000, and 2006, respectively. He joined NTT Laboratories, Kanagawa, Japan, in 2000. He is also a Researcher at Japan Science and Technology Agency, Sapporo, Japan. His research fields of interest are design and control of network systems including power grids.



Keiji Takano was born in Tokyo, Japan, on September 8, 1987. He received his B.E and M.E degrees from Tokyo Institute of Technology, Japan in 2011 and 2013, respectively. He works at the Daiwa Institute of Research, Tokyo, Japan, now. His research fields of interest include artificial intelligence and computational complexity.



Takayuki Watanabe was born in Ibaraki, Japan, on April 8, 1988. He received his B.E and M.E degrees from Waseda University, Japan in 2011 and 2013 respectively. He joined the Innovative IP Architecture Center at NTT Communications, Tokyo, Japan, in April 2013. His research fields are related to malware analysis and control of network systems including power systems.



algorithms.

Jun Kawahara was born in Osaka, Japan, in 1981. He received the B.S. degree in science and the M.E. and Ph.D. degrees in informatics from Kyoto University, Japan, in 2004, 2006, and 2009, respectively. In 2009–2010, he worked at Kyoto University as a Researcher. He joined the JST ERATO Minato Discrete Structure Manipulation System Project as a Researcher in 2010–2012. Currently, he is an Assistant Professor at Nara Institute of Science and Technology, Nara, Japan. His research interests include theoretical computer science and discrete



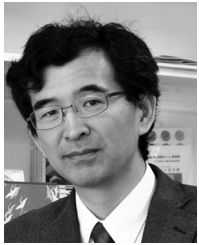
Ryo Yoshinaka received LL.B., Master of Arts and Sciences, and Ph.D. degrees from the University of Tokyo, Japan, in 2000, 2003 and 2006, respectively. Subsequently, he worked as a Postdoctor in INRIA-Lorraine, Hokkaido University, and Japan Science and Technology Agency. He is currently working as an assistant professor in Kyoto University, Kyoto, Japan. His academic interest includes algorithmic learning theory, formal language theory, and algorithms and data structures.



Akihiro Kishimoto received the B.Sc. degree from the University of Tokyo, Japan, in 1997 and the M.Sc. and Ph.D. degrees from the University of Alberta, Canada, in 2001 and 2005, respectively. He is a Research Staff Member at IBM Research, Ireland. His research interests include artificial intelligence and parallel computing. Particularly he is interested in developing high-performance game-playing programs, planning, and risk management systems.



Koji Tsuda received B.E., M.E., and Ph.D degrees from Kyoto University, Japan, in 1994, 1995, and 1998, respectively. Subsequently, he joined former Electrotechnical Laboratory (ETL), Tsukuba, Japan, as Research Scientist. When ETL was reorganized as AIST in 2001, he joined newly established Computational Biology Research Center, Tokyo, Japan. In 2000–2001, he worked at GMD FIRST (currently Fraunhofer FIRST) in Berlin, Germany, as Visiting Scientist. In 2003–2004 and 2006–2008, he worked at Max Planck Institute for Biological Cybernetics, Tübingen, Germany, first as Research Scientist and later as Project Leader. Currently, he is Senior Research Scientist at AIST Computational Biology Research Center and Group Leader of JST ERATO Minato Project. His interests include machine learning, data mining, discrete optimization, and their applications to various disciplines including computational biology and smart grid technologies.



Shin-ichi Minato (M'93) received the B.E., M.E., and D.E. degrees in information science from Kyoto University, Japan, in 1988, 1990, and 1995, respectively. He is a Professor at the Graduate School of Information Science and Technology, Hokkaido University, Japan. He also serves as the Research Director of the ERATO Minato Discrete Structure Manipulation System Project, executed by Japan Science and Technology Agency. He worked for NTT Laboratories from 1990 until 2004. He was a Visiting Scholar at the Computer Science

Department of Stanford University in 1997. He joined Hokkaido University as an Associate Professor in 2004, and has been a Professor since October 2010. His interests include efficient representations and manipulation algorithms for large-scale discrete structures, such as Boolean functions, sets of combinations, sequences, permutations, etc. He has published *Binary Decision Diagrams and Applications for VLSI CAD* (Kluwer, 1995). He is a senior member of IEICE, and is a member of IPSJ and JSAI.



Yasuhiro Hayashi (M'91) received his B.E., M.E., and D.Eng. degrees from Waseda University, Japan, in 1989, 1991 and 1994, respectively. In April 1994, he became a Research Associate at Ibaraki University, Japan. In April 2000, he became an Associate Professor at the Department of Electrical and Electronics Engineering at Fukui University, Japan. He became a Professor at the Department of Electrical Engineering and Bioscience at Waseda University in April 2009. He also became a Director of Research Institute of Advanced Network Technology (RIANT) at Waseda University in 2010. His research fields of interest are optimization of distribution system and forecasting, operation, planning, and control concerned with renewable energy sources and demand response.