Review Article
A Review of Piecewise Linearization Methods

Ming-Hua Lin,[1] John Gunnar Carlsson,[2] Dongdong Ge,[3] Jianming Shi,[4] and Jung-Fa Tsai[5]

[1] *Department of Information Technology and Management, Shih Chien University, No. 70 Dazhi Street, Taipei 10462, Taiwan*
[2] *Program in Industrial and Systems Engineering, University of Minnesota, 111 Church Street SE, Minneapolis, MN 55455, USA*
[3] *School of Information Management and Engineering, Shanghai University of Finance and Economics, Shanghai 200433, China*
[4] *School of Management, Tokyo University of Science, 500 Shimokiyoku, Kuki, Saitama 346-8512, Japan*
[5] *Department of Business Management, National Taipei University of Technology, Section 3, No. 1 Chung-Hsiao E. Road,
Taipei 10608, Taiwan*

Correspondence should be addressed to Jung-Fa Tsai; jftsai@ntut.edu.tw

Various optimization problems in engineering and management are formulated as nonlinear programming problems. Because of the nonconvexity nature of this kind of problems, no efficient approach is available to derive the global optimum of the problems. How to locate a global optimal solution of a nonlinear programming problem is an important issue in optimization theory. In the last few decades, piecewise linearization methods have been widely applied to convert a nonlinear programming problem into a linear programming problem or a mixed-integer convex programming problem for obtaining an approximated global optimal solution. In the transformation process, extra binary variables, continuous variables, and constraints are introduced to reformulate the original problem. These extra variables and constraints mainly determine the solution efficiency of the converted problem. This study therefore provides a review of piecewise linearization methods and analyzes the computational efficiency of various piecewise linearization methods.

1. Introduction

Piecewise linear functions are frequently used in various applications to approximate nonlinear programs with nonconvex functions in the objective or constraints by adding extra binary variables, continuous variables, and constraints. They naturally appear as cost functions of supply chain problems to model quantity discount functions for bulk procurement and fixed charges. For example, the transportation cost, inventory cost, and production cost in a supply chain network are often constructed as a sum of nonconvex piecewise linear functions due to economies of scale [1]. Optimization problems with piecewise linear costs arise in many application domains, including transportation, telecommunications, and production planning. Specific applications include variants of the minimum cost network flow problem with nonconvex piecewise linear costs [2–7], the network loading problem [8–11], the facility location problem with staircase costs [12, 13], the merge-in-transit problem [14], and the packing problem [15–17]. Other applications also include production planning [18], optimization of electronic circuits [19], operation planning of gas networks [20], process engineering [21, 22], engineering design [23, 24], appointment scheduling [25], and other network flow problems with nonconvex piecewise linear objective functions [7].

Various methods of piecewisely linearizing a nonlinear function have been proposed in the literature [26–39]. Two well-known mixed-integer formulations for piecewise linear functions are the incremental cost [40] and the convex combination [41] formulations. Padberg [35] compared the linear programming relaxations of the two mixed-integer programming models for piecewise linear functions in the simplest case when no constraint exists. He showed that the feasible set of the linear programming relaxation of the incremental cost formulation is integral; that is, the binary variables are integers at every vertex of the set. He called such formulations locally ideal. On the other hand, the convex combination formulation is not locally ideal, and it

strictly contains the feasible set of the linear programming relaxation of the incremental cost formulation. Then, Sherali [42] proposed a modified convex combination formulation that is locally ideal. Alternatively, Beale and Tomlin [43] suggested a formulation for the piecewise linear function similar to convex combination, except that no binary variable is included in the model and the nonlinearities are enforced algorithmically, directly in the branch-and-bound algorithm, by branching on sets of variables, which they called special ordered sets of type 2 (SOS2). It is also possible to formulate piecewise linear functions similar to incremental cost but without binary variables and enforcing the nonlinearities directly in the branch-and-bound algorithm. Two advantages of eliminating binary variables are the substantial reduction in the size of the model and the use of the polyhedral structure of the problem [44, 45]. Keha et al. [46] studied formulations of linear programs with piecewise linear objective functions with and without additional binary variables and showed that adding binary variables does not improve the bound of the linear programming relaxation. Keha et al. [47] also presented a branch-and-cut algorithm for solving linear programs with continuous separable piecewise-linear cost functions. Instead of introducing auxiliary binary variables and other linear constraints to represent SOS2 constraints used in the traditional approach, they enforced SOS2 constraints by branching on them without auxiliary binary variables.

Due to the broad applications of piecewise linear functions, many studies have conducted related research on this topic. The main purpose of these studies is to find a better way to represent a piecewise linear function or to tighten the linear programming relaxation. A superior representation of piecewise linear functions can effectively reduce the problem size and enhance the computational efficiency. However, for expressing a piecewise linear function of a single variable x with $m+1$ break points, most of the methods in the textbooks and literature require adding extra m binary variables and $4m$ constraints, which may cause a heavy computational burden when m is large. Recently, Li et al. [48] developed a representation method for piecewise linear functions with fewer binary variables compared to the traditional methods. Although their method needs only $\lceil \log_2 m \rceil$ extra binary variables to piecewisely linearize a nonlinear function with $m + 1$ break points, the approximation process still requires $8 + 8\lceil \log_2 m \rceil$ extra constraints, $2m$ nonnegative continuous variables, and $2\lceil \log_2 m \rceil$ free-signed continuous variables. Vielma et al. [39] presented a note on Li et al.'s paper and showed that two representations for piecewise linear functions introduced by Li et al. [48] are both theoretically and computationally inferior to standard formulations for piecewise linear functions. Tsai and Lin [49] applied the Vielma et al. [39] techniques to express a piecewise linear function for solving a posynomial optimization problem. Croxton et al. [31] indicated that most models of expressing piecewise linear functions are equivalent to each other. Additionally, it is well known that the numbers of extra variables and constraints required in the linearization process for a nonlinear function obviously impact the computational performance of the converted problem. Therefore, this paper focuses on discussing and reviewing the recent advances in piecewise linearization methods. Section 2 reviews the piecewise linearization methods. Section 3 compares the formulations of various methods with the numbers of extra binary/continuous variables and constraints. Section 4 discusses error evaluation in piecewise linear approximation. Conclusions are made in Section 5.

2. Formulations of Piecewise Linearization Functions

Consider a general nonlinear function $f(x)$ of a single variable x; $f(x)$ is a continuous function, and x is within the interval $[a_0, a_m]$. Most commonly used textbooks of nonlinear programming [26–28] approximate the nonlinear function by a piecewise linear function as follows.

Firstly, denote a_k ($k = 0, 2, \ldots, m$) as the break points of $f(x)$, $a_0 < a_1 < \cdots < a_m$, and Figure 1 indicates the piecewise linearization of $f(x)$.

$f(x)$ can then be approximately linearized over the interval $[a_0, a_m]$ as

$$L(f(x)) = \sum_{k=0}^{m} f(a_k) t_k, \tag{1}$$

where $x = \sum_{k=0}^{m} a_k t_k$, $\sum_{k=0}^{m} t_k = 1$, $t_k \geq 0$, in which only two adjacent t_k's are allowed to be nonzero. A nonlinear function is then converted into the following expressions.

Method 1. Consider

$$L(f(x)) = \sum_{k=1}^{m} f(a_k) t_k,$$

$$x = \sum_{k=1}^{m} a_k t_k,$$

$$t_0 \leq y_0, \tag{2}$$

$$t_k \leq y_{k-1} + y_k, \quad \text{for } k = 1, 2, \ldots, m-1, \ t_m \leq y_{m-1},$$

$$\sum_{k=0}^{m-1} y_k = 1, \qquad \sum_{k=0}^{m} t_k = 1,$$

where $y_k \in \{0, 1\}$, $t_k \geq 0$, $k = 0, 1, \ldots, m-1$.

The above expressions involve m new binary variables $y_0, y_1, \ldots, y_{m-1}$. The number of newly added 0-1 variables for piecewisely linearizing a function $f(x)$ equals the number of breaking intervals (i.e., m). If m is large, it may cause a heavy computational burden.

Li and Yu [33] proposed another global optimization method for nonlinear programming problems where the objective function and the constraints might be nonconvex. A univariate function is initially expressed by a piecewise
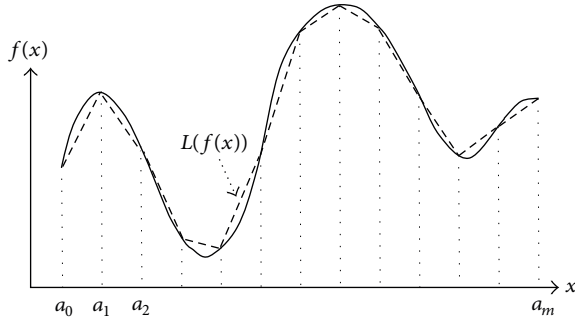
FIGURE 1: Piecewise linearization of $f(x)$.

linear function with a summation of absolute terms. Denote s_k ($k = 0, 1, \ldots, m-1$) as the slopes of line segments between a_k and a_{k+1}, expressed as $s_k = [f(a_{k+1}) - f(a_k)]/[a_{k+1} - a_k]$. $f(x)$ can then be written as follows:

$$
\begin{aligned}
L(f(x)) = {} & f(a_0) + s_0(x - a_0) \\
& + \sum_{k=1}^{m-1} \frac{s_k - s_{k-1}}{2} \left(|x - a_k| + x - a_k \right).
\end{aligned}
\tag{3}
$$

$f(x)$ is convex in the interval $[a_{k-1}, a_k]$ if $s_k - s_{k-1} \geq 0$, and otherwise $f(x)$ is a non-convex function which needs to be linearized by adding extra binary variables. By linearizing the absolute terms, Li and Yu [33] converted the nonlinear function into a piecewise linear function as shown below.

Method 2. Consider

$$
\begin{aligned}
L(f(x)) = {} & f(a_0) + s_0(x - a_0) \\
& + \sum_{k:s_k > s_{k-1}} (s_k - s_{k-1}) \left(x - a_k + \sum_{l=0}^{k-1} d_l \right) \\
& + \frac{1}{2} \sum_{k:s_k < s_{k-1}} (s_k - s_{k-1}) (x - 2z_k + 2a_k u_k - a_k), \\
& x + \sum_{l=0}^{m-2} d_l \geq a_{m-1}, \qquad 0 \leq d_l \leq a_{l+1} - a_l, \\
& \qquad\qquad\qquad\qquad \text{where } s_k > s_{k-1}, \\
& x + \overline{x}(u_k - 1) \leq z_k, \qquad z_k \geq 0, \quad \text{where } s_k < s_{k-1},
\end{aligned}
\tag{4}
$$

where $x \geq 0$, $d_l \geq 0$, $z_k \geq 0$, $u_k \in \{0, 1\}$, $\overline{x}$ are upper bounds of x and u_k are extra binary variables used to linearize a non-convex function $f(x)$ for the interval $[a_{k-1}, a_k]$.

Comparing Method 2 with Method 1, Method 1 uses binary variables to linearize $f(x)$ for whole x interval. But the binary variables used in Method 2 are only applied to linearize the non-convex parts of $f(x)$. Method 2 therefore uses fewer 0-1 variables than Method 1. However, for $f(x)$ with q intervals of the non-convex parts, Method 2 still requires q binary variables to linearize $f(x)$.

Another general form of representing a piecewise linear function is proposed in the articles of Croxton et al. [31], Li [32], Padberg [35], Topaloglu and Powell [36], and Li and Tsai [38]. The expressions are formulated as shown below.

Method 3. Consider

$$
\begin{aligned}
& a_k - (a_m - a_0)(1 - \lambda_k) \\
& \quad \leq x \leq a_{k+1} - (a_m - a_0)(1 - \lambda_k), \qquad k = 0, 1, \ldots, m-1,
\end{aligned}
\tag{5}
$$

where $\sum_{k=0}^{m-1} \lambda_k = 1$, $\lambda_k \in \{0, 1\}$, and

$$
\begin{aligned}
& f(a_k) + s_k(x - a_k) - M(1 - \lambda_k) \\
& \quad \leq f(x) \\
& \quad \leq f(a_k) + s_k(x - a_k) \\
& \qquad + M(1 - \lambda_k), \quad k = 0, 1, \ldots, m-1,
\end{aligned}
\tag{6}
$$

where M is a large constant and $s_k = (f(a_{k+1}) - f(a_k))/(a_{k+1} - a_k)$.

The above expressions require extra m binary variables and $4m$ constraints, where $m + 1$ break points are used to represent a piecewise linear function.

Form the above discussions, we can know that Methods 1, 2, and 3 require a number of extra binary variables and extra constraints linear in m to express a piecewise linear function. To approximate a nonlinear function by using a piecewise linear function, the numbers of extra binary variable and constraints significantly influence the computational efficiency. If fewer binary variables and constraints are used to represent a piecewise linear function, then less CPU time is needed to solve the transformed problem. For decreasing the extra binary variables involved in the approximation process, Li et al. [48] developed a representation method for piecewise linear functions with the number of binary variables logarithmic in m. Consider the same piecewise linear function $f(x)$ discussed above, where x is within the interval $[a_0, a_m]$ and $m + 1$ break points exist within $[a_0, a_m]$. Let θ be an integer, $0 \leq \theta \leq m - 1$, expressed as

$$
\theta = \sum_{j=1}^{h} 2^{j-1} u_j, \quad h = \lceil \log_2 m \rceil, \quad u_j \in \{0, 1\}.
\tag{7}
$$

Let $G(\theta) \subseteq \{1, 2, \ldots, h\}$ be a set composed of all indices such that $\sum_{j \in G(\theta)} 2^{j-1} = \theta$. For instance, $G(0) = \phi$, $G(3) = \{1, 2\}$.

Denote $\|G(\theta)\|$ to be the number of elements in $G(\theta)$. For instance, $\|G(0)\| = 0$, $\|G(3)\| = 2$.

To approximate a univariate nonlinear function by using a piecewise linear function, the following expressions are deduced by the Li et al. [48] method.

Method 4. Consider

$$\sum_{\theta=0}^{m-1} r_\theta a_\theta \le x \le \sum_{\theta=0}^{m-1} r_\theta a_{\theta+1},$$

$$f(x) = \sum_{\theta=0}^{m-1} \left(f(a_\theta) - s_\theta (a_\theta - a_0) \right) r_\theta$$

$$+ \sum_{\theta=0}^{m-1} s_\theta w_\theta, \quad s_\theta = \frac{f(a_{\theta+1}) - f(a_\theta)}{a_{\theta+1} - a_\theta},$$

$$\sum_{\theta=0}^{m-1} r_\theta = 1, \quad r_\theta \ge 0,$$

$$\sum_{\theta=0}^{m-1} r_\theta \|G(\theta)\| + \sum_{j=1}^{h} z_j = 0,$$

$$-u'_j \le z_j \le u'_j, \quad j = 1, 2, \ldots, h,$$

$$\sum_{\theta=0}^{m-1} r_\theta c_{\theta,j} - \left(1 - u'_j\right) \le z_j \le \sum_{\theta=0}^{m-1} r_\theta c_{\theta,j} + \left(1 - u'_j\right),$$

$$j = 1, 2, \ldots, h,$$

$$\sum_{\theta=0}^{m-1} w_\theta = x - a_0,$$

$$\sum_{\theta=0}^{m-1} w_\theta \|G(\theta)\| + \sum_{j=1}^{h} \delta_j = 0,$$

$$-\left(a_m - a_0\right) u'_j \le \delta_j \le \left(a_m - a_0\right) u'_j, \quad j = 1, 2, \ldots, h,$$

$$\sum_{\theta=0}^{m-1} w_\theta c_{\theta,j} - \left(a_m - a_0\right)\left(1 - u'_j\right)$$

$$\le \delta_j \le \sum_{\theta=0}^{m-1} w_\theta c_{\theta,j} + \left(a_m - a_0\right)\left(1 - u'_j\right), \quad j = 1, 2, \ldots, h,$$

$$\sum_{j=1}^{h} 2^{j-1} u'_j \le m,$$

$$(8)$$

where $u'_j \in \{0, 1\}$, $c_{\theta,j}$, z_j, and δ_j are free continuous variables, r_θ and w_θ are nonnegative continuous, and all the variables are the same as defined before.

The expressions of Method 4 for representing a piecewise linear function $f(x)$ with $m + 1$ break points use $\lceil \log_2 m \rceil$ binary variables, $8 + 8\lceil \log_2 m \rceil$ constraints, $2m$ non-negative variables, and $2\lceil \log_2 m \rceil$ free-signed continuous variables. Comparing with Methods 1, 2, and 3, Method 4 indeed reduces the number of binary variables used such that the computational efficiency is improved. Although Li et al. [48] developed a superior way of expressing a piecewise linear function by using fewer binary variables, Vielma et al. [39]

investigated that this representation for piecewise linear functions is theoretically and computationally inferior to standard formulations for piecewise linear functions. Vielma and Nemhauser [50] recently developed a novel piecewise linear expression requiring fewer variables and constraints than the current piecewise linearization techniques to approximate the univariate nonlinear functions. Their method needs a logarithmic number of binary variables and constraints to express a piecewise linear function. The formulation is described as shown below.

Let $P = \{0, 1, 2, \ldots, m\}$ and $p \in P$. An injective function $B : \{1, 2, \ldots, m\} \rightarrow \{0, 1\}^\theta$, $\theta = \lceil \log_2 m \rceil$, where the vectors $B(p)$ and $B(p + 1)$ differ in at most one component for all $p \in \{1, 2, \ldots, m - 1\}$.

Let $B(p) = (u_1, u_2, \ldots, u_\theta)$, for all $u_k \in \{0, 1\}$, $k = 1, 2, \ldots, \theta$, and $B(0) = B(1)$. Some notations are introduced below.

$S^+(k)$: a set composed of all p, where $u_k = 1$ of $B(p)$ and $B(p + 1)$ for $p = 1, 2, \ldots, m - 1$ or $u_k = 1$ of $B(p)$ for $p \in \{0, m\}$; that is, $S^+(k) = \{p \mid \forall B(p) \text{ and } B(p + 1), u_k = 1, p = 1, 2, \ldots, m - 1\} \cup \{p \mid \forall B(p), u_k = 1, p \in \{0, m\}\}$.

$S^-(k)$: a set composed of all p, where $u_k = 0$ of $B(p)$ and $B(p + 1)$ for $p = 1, 2, \ldots, m - 1$ or $u_k = 0$ of $B(p)$ for $p \in \{0, m\}$; that is, $S^-(k) = \{p \mid \forall B(p) \text{ and } B(p + 1), u_k = 0, p = 1, 2, \ldots, m - 1\} \cup \{p \mid \forall B(p), u_k = 0, p \in \{0, m\}\}$.

The linear approximation of a univariate $f(x)$, $a_0 \le x \le a_m$, by the technique of Vielma and Nemhauser [50] is formulated as follows.

Method 5. Denote $L(f(x))$ as the piecewise linear function of $f(x)$, where $a_0 < a_1 < a_2 < \cdots < a_m$ be the $m+1$ break points of $L(f(x))$. $L(f(x))$ can be expressed as

$$L(f(x)) = \sum_{p=0}^{m} f(a_p) \lambda_p,$$

$$x = \sum_{p=0}^{m} a_p \lambda_p,$$

$$\sum_{p=0}^{m} \lambda_p = 1, \quad (9)$$

$$\sum_{p \in S^+(k)} \lambda_p \le u_k,$$

$$\sum_{p \in S^-(k)} \lambda_p \le 1 - u_k, \quad \forall \lambda_p \in \mathfrak{R}_+, \ \forall u_k \in \{0, 1\}.$$

Method 5 uses $\lceil \log_2 m \rceil$ binary variables, $m + 1$ continuous variables, and $3 + 2\lceil \log_2 m \rceil$ constraints to express a piecewise linearization function with m line segments.

3. Formulation Comparisons

The comparison results of the above five methods in terms of the numbers of binary variables, continuous variables, and constraints are listed in Table 1. The number of extra binary variables of Methods 1 and 3 is linear in the number of line segments. Methods 4 and 5 have the logarithmic number of

extra binary variables with m line segments, and the number of extra binary variables of Method 2 is equal to the number of concave piecewise line segments. In the deterministic global optimization for a minimization problem, inverse, power, and exponential transformations generate nonconvex expressions that require to be linearly approximated in the reformulated problem. That means Methods 4 and 5 are superior to Methods 1, 2, and 3 in terms of the numbers of extra binary variables and constraints as shown in Table 1. Moreover, Method 5 has fewer extra continuous variables and constraints than Method 4 in linearizing a nonlinear function.

Till et al. [51] reviewed the literature on the complexity of mixed-integer linear programming (MILP) problems and summarized that the computational complexity varies from $O(d \cdot n^2)$ to $O(2^d \cdot n^3)$, where n is the number of constraints and d is the number of binaries. Therefore, reducing constraints and binary variables makes a greater impact than reducing continuous variables on computational efficiency of solving MILP problems. For finding a global solution of a nonlinear programming problem by a piecewise linearization method, if the linearization method generates a large number of additional constraints and binaries, the computational efficiency will decrease and cause heavy computational burdens. According to the above discussions, Method 5 is more computationally efficient than the other four methods. Experiment results from the literature [39, 48, 49] also support the statement.

Beale and Tomlin [43] suggested a formulation for piecewise linear functions by using continuous variables in special ordered sets of type 2 (SOS2). Although no binary variables are included in the SOS2 formulation, the nonlinearities are enforced algorithmically and directly in the branch-and-bound algorithm by branching on sets of variables. Since the traditional SOS2 branching schemes have too many dichotomies, the piecewise linearization technique in Method 5 induces an independent branching scheme of logarithm depth and provides a significant computational advantage [50]. The computational results in Vielma and Nemhauser [50] show that Method 5 outperforms the SOS2 model without binary variables.

The factors affecting the computational efficiency in solving nonlinear programming problems include the tightness of the constructed convex underestimator, the efficiency of the piecewise linearization technique, and the number of the transformed variables. An appropriate variable transformation constructs a tighter convex underestimator and makes fewer break points required in the linearization process to satisfy the same optimality tolerance and feasibility tolerance. Vielma and Nemhauser [50] indicated that the formulation of Method 5 is sharp and locally ideal and has favorable tightness properties. They presented experimental results showing that Method 5 significantly outperforms other methods, especially when the number of break points becomes large. Vielma et al. [39] explained that the formulation of Method 4 is not sharp and is theoretically and computationally inferior to standard MILP formulations (convex combination model, logarithmic convex combination model) for piecewise linear functions.

4. Error Evaluation

For evaluating the error of piecewise linear approximation, Tsai and Lin [49, 52] and Lin and Tsai [53] utilized the expression $|f(x) - L(f(x))|$ to estimate the error indicated in Figure 2. If $f(x)$ is the objective function, $g_i(x) < 0$ is the ith constraint, and x^* is the solution derived from the transformed program, then the linearization does not require to be refined until $|f(x^*) - L(f(x^*))| \leq \varepsilon_1$ and $\text{Max}_i(g_i(x^*)) \leq \varepsilon_2$, where $|f(x^*) - L(f(x^*))|$ is the evaluated error in objective, ε_1 is the optimality tolerance, $g_i(x^*)$ is the error in the ith constraint, and ε_2 is the feasibility tolerance.

The accuracy of the linear approximation significantly depends on the selection of break points and more break points can increase the accuracy of the linear approximation. Since adding numerous break points leads to a significant increase in the computational burden, the break point selection strategies can be applied to improve the computational efficiency in solving optimization problems by the deterministic approaches. Existing break point selection strategies are classified into three categories as follows [54]:

(i) add a new break point at the midpoint of each interval of existing break points;

(ii) add a new break point at the point with largest approximation error of each interval;

(iii) add a new break point at the previously obtained solution point.

According to the deterministic optimization methods for solving nonconvex nonlinear problems [29, 33, 38, 39, 48, 49, 53–56], the inverse or logarithmic transformation is required to be approximated by the piecewise linearization function. For example, the function $y = \ln x$ or $y = x^{-1}$ is required to be piecewisely linearized by using an appropriate breakpoint selection strategy, if a new break point is added at the midpoint of each interval of existing break points or at the point with largest approximation error, the number of line segments becomes double in each iteration. If a new breakpoint is added at the previously obtained solution point, only one breakpoint is added in each iteration. How to improve the computational efficiency by a better break point selection strategy still needs more investigations or experiments to get concrete results.

5. Conclusions

This study provides an overview on some of the most commonly used piecewise linearization methods in deterministic optimization. From the formulation point of view, the numbers of extra binaries, continuous variables, and constraints are decreasing in the latest development methods especially for the number of extra binaries which may cause heavy computational burdens. Additionally, a good piecewise linearization method must consider the tightness properties such as sharp and locally ideal. Since effective break points selection strategy is important to enhance the computational efficiency in linear approximation, more work should be done to study the optimal positioning of the break points.

TABLE 1: Comparison results of five methods in expressing a piecewise linearization function with m line segments (i.e., $m + 1$ break points).

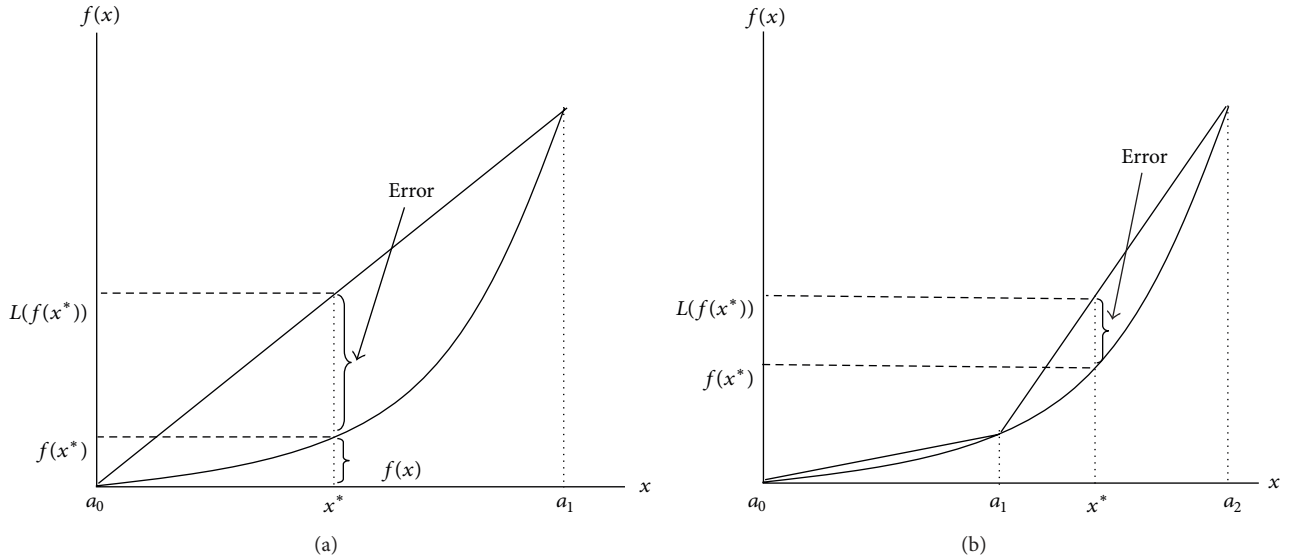| Items	Method 1	Method 2	Method 3	Method 4	Method 5
No. of binary variables	m	q (no. of concave piecewise segments)	m	$\lceil \log_2 m \rceil$	$\lceil \log_2 m \rceil$
No. of continuous variables	$m + 1$	$m + 1$	0	$2m + 2\lceil \log_2 m \rceil$	$m + 1$
No. of constraints	$m + 5$	$m + 1$	$4m$	$8 + 8\lceil \log_2 m \rceil$	$3 + 2\lceil \log_2 m \rceil$

(a)

(b)

FIGURE 2: Error evaluation of the linear approximation.

Although a logarithmic piecewise linearization method with good tightness properties has been proposed, it is still too time consuming for finding an approximately global optimum of a large scale nonconvex problem. Developing an efficient polynomial time algorithm for solving nonconvex problems by piecewise linearization techniques is still a challenging question. Obviously, this contribution gives only a few preliminary insights and might point toward issues deserving additional research.

Acknowledgment

References

[1] J. F. Tsai, "An optimization approach for supply chain management models with quantity discount policy," *European Journal of Operational Research*, vol. 177, no. 2, pp. 982–994, 2007.

[2] E. H. Aghezzaf and L. A. Wolsey, "Modelling piecewise linear concave costs in a tree partitioning problem," *Discrete Applied Mathematics*, vol. 50, no. 2, pp. 101–109, 1994.

[3] A. Balakrishnan and S. Graves, "A composite algorithm for a concave-cost network flow problem," *Networks*, vol. 19, no. 2, pp. 175–202, 1989.

[4] K. L. Croxton, *Modeling and solving network flow problems with piecewise linear costs, with applications in supply chain management [Ph.D. thesis]*, Operations Research Center, Massachusetts Institute of Technology, Cambridge, Mass, USA, 1999.

[5] L. M. A. Chan, A. Muriel, Z. J. Shen, and D. Simchi-Levi, "On the effectiveness of zero-inventory-ordering policies for the economic lot-sizing model with a class of piecewise linear cost structures," *Operations Research*, vol. 50, no. 6, pp. 1058–1067, 2002.

[6] L. M. A. Chan, A. Muriel, Z. J. M. Shen et al., "Effective zero-inventory-ordering policies for the single-warehouse multiretailer problem with piecewise linear cost structures," *Management Science*, vol. 48, no. 11, pp. 1446–1460, 2002.

[7] K. L. Croxton, B. Gendron, and T. L. Magnanti, "Variable disaggregation in network flow problems with piecewise linear costs," *Operations Research*, vol. 55, no. 1, pp. 146–157, 2007.

[8] D. Bienstock and O. Günlük, "Capacitated network design polyhedral structure and computation," *INFORMS Journal on Computing*, vol. 8, no. 3, pp. 243–259, 1996.

[9] V. Gabrel, A. Knippel, and M. Minoux, "Exact solution of multicommodity network optimization problems with general step cost functions," *Operations Research Letters*, vol. 25, no. 1, pp. 15–23, 1999.

[10] O. Günlük, "A branch-and-cut algorithm for capacitated network design problems," *Mathematical Programming A*, vol. 86, no. 1, pp. 17–39, 1999.

[11] T. L. Magnanti, P. Mirchandani, and R. Vachani, "Modeling and solving the two-facility capacitated network loading problem," *Operations Research*, vol. 43, no. 1, pp. 142–157, 1995.

[12] K. Holmberg, "Solving the staircase cost facility location problem with decomposition and piecewise linearization," *European Journal of Operational Research*, vol. 75, no. 1, pp. 41–61, 1994.

[13] K. Holmberg and J. Ling, "A Lagrangean heuristic for the facility location problem with staircase costs," *European Journal of Operational Research*, vol. 97, no. 1, pp. 63–74, 1997.

[14] K. L. Croxton, B. Gendron, and T. L. Magnanti, "Models and methods for merge-in-transit operations," *Transportation Science*, vol. 37, no. 1, pp. 1–22, 2003.

[15] H. L. Li, C. T. Chang, and J. F. Tsai, "Approximately global optimization for assortment problems using piecewise linearization techniques," *European Journal of Operational Research*, vol. 140, no. 3, pp. 584–589, 2002.

[16] J. F. Tsai and H. L. Li, "A global optimization method for packing problems," *Engineering Optimization*, vol. 38, no. 6, pp. 687–700, 2006.

[17] J. F. Tsai, P. C. Wang, and M. H. Lin, "An efficient deterministic optimization approach for rectangular packing problems," *Optimization*, 2013.

[18] R. Fourer, D. M. Gay, and B. W. Kernighan, *AMPL—A Modeling Language for Mathematical Programming*, The Scientific Press, San Francisco, Calif, USA, 1993.

[19] T. Graf, P. van Hentenryck, C. Pradelles-Lasserre, and L. Zimmer, "Simulation of hybrid circuits in constraint logic programming," *Computers & Mathematics with Applications*, vol. 20, no. 9-10, pp. 45–56, 1990.

[20] A. Martin, M. Möller, and S. Moritz, "Mixed integer models for the stationary case of gas network optimization," *Mathematical Programming*, vol. 105, no. 2-3, pp. 563–582, 2006.

[21] M. L. Bergamini, P. Aguirre, and I. Grossmann, "Logic-based outer approximation for globally optimal synthesis of process networks," *Computers and Chemical Engineering*, vol. 29, no. 9, pp. 1914–1933, 2005.

[22] M. L. Bergamini, I. Grossmann, N. Scenna, and P. Aguirre, "An improved piecewise outer-approximation algorithm for the global optimization of MINLP models involving concave and bilinear terms," *Computers and Chemical Engineering*, vol. 32, no. 3, pp. 477–493, 2008.

[23] J. F. Tsai, "Global optimization of nonlinear fractional programming problems in engineering design," *Engineering Optimization*, vol. 37, no. 4, pp. 399–409, 2005.

[24] M. H. Lin, J. F. Tsai, and P. C. Wang, "Solving engineering optimization problems by a deterministic global approach," *Applied Mathematics and Information Sciences*, vol. 6, no. 7, pp. 21S–27S, 2012.

[25] D. Ge, G. Wan, Z. Wang, and J. Zhang, "A note on appointment scheduling with piece-wise linear cost functions," Working paper, 2012.

[26] M. S. Bazaraa, H. D. Sherali, and C. M. Shetty, *Nonlinear Programming—Theory and Algorithms*, John Wiley & Sons, New York, NY, USA, 2nd edition, 1993.

[27] F. S. Hillier and G. J. Lieberman, *Introduction to Operations Research*, McGraw-Hill, New York, NY, USA, 6th edition, 1995.

[28] H. A. Taha, *Operations Research*, Macmillan, New York, NY, USA, 7th edition, 2003.

[29] C. A. Floudas, *Deterministic Global Optimization—Theory, Methods, and Applications*, Kluwer Academic Publishers, Dordrecht, The Netherlands, 1999.

[30] S. Vajda, *Mathematical Programming*, Addison-Wesley, New York, NY, USA, 1964.

[31] K. L. Croxton, B. Gendron, and T. L. Magnanti, "A comparison of mixed-integer programming models for nonconvex piecewise linear cost minimization problems," *Management Science*, vol. 49, no. 9, pp. 1268–1273, 2003.

[32] H. L. Li, "An efficient method for solving linear goal programming problems," *Journal of Optimization Theory and Applications*, vol. 90, no. 2, pp. 465–469, 1996.

[33] H. L. Li and C. S. Yu, "Global optimization method for nonconvex separable programming problems," *European Journal of Operational Research*, vol. 117, no. 2, pp. 275–292, 1999.

[34] H. L. Li and H. C. Lu, "Global optimization for generalized geometric programs with mixed free-sign variables," *Operations Research*, vol. 57, no. 3, pp. 701–713, 2009.

[35] M. Padberg, "Approximating separable nonlinear functions via mixed zero-one programs," *Operations Research Letters*, vol. 27, no. 1, pp. 1–5, 2000.

[36] H. Topaloglu and W. B. Powell, "An algorithm for approximating piecewise linear concave functions from sample gradients," *Operations Research Letters*, vol. 31, no. 1, pp. 66–76, 2003.

[37] S. Kontogiorgis, "Practical piecewise-linear approximation for monotropic optimization," *INFORMS Journal on Computing*, vol. 12, no. 4, pp. 324–340, 2000.

[38] H. L. Li and J. F. Tsai, "Treating free variables in generalized geometric global optimization programs," *Journal of Global Optimization*, vol. 33, no. 1, pp. 1–13, 2005.

[39] J. P. Vielma, S. Ahmed, and G. Nemhauser, "A note on 'a superior representation method for piecewise linear functions'," *INFORMS Journal on Computing*, vol. 22, no. 3, pp. 493–497, 2010.

[40] H. M. Markowitz and A. S. Manne, "On the solution of discrete programming problems," *Econometrica*, vol. 25, no. 1, pp. 84–110, 1957.

[41] G. B. Dantzig, "On the significance of solving linear programming problems with some integer variables," *Econometrica*, vol. 28, no. 1, pp. 30–44, 1960.

[42] H. D. Sherali, "On mixed-integer zero-one representations for separable lower-semicontinuous piecewise linear functions," *Operations Research Letters*, vol. 28, no. 4, pp. 155–160, 2001.

[43] E. L. M. Beale and J. A. Tomlin, "Special facilities in a general mathematical programming system for nonconvex problems using ordered sets of variables," in *Proceedings of the 5th International Conference on Operations Research*, J. Lawrence, Ed., pp. 447–454, Tavistock Publications, London, UK, 1970.

[44] I. R. de Farias Jr., E. L. Johnson, and G. L. Nemhauser, "Branch-and-cut for combinatorial optimization problems without auxiliary binary variables," *The Knowledge Engineering Review*, vol. 16, no. 1, pp. 25–39, 2001.

[45] T. Nowatzki, M. Ferris, K. Sankaralingam, and C. Estan, *Optimization and Mathematical Modeling in Computer Architecture*, Morgan & Claypool Publishers, San Rafael, Calif, USA, 2013.

[46] A. B. Keha, I. R. de Farias, and G. L. Nemhauser, "Models for representing piecewise linear cost functions," *Operations Research Letters*, vol. 32, no. 1, pp. 44–48, 2004.

[47] A. B. Keha, I. R. de Farias, and G. L. Nemhauser, "A branch-and-cut algorithm without binary variables for nonconvex piecewise linear optimization," *Operations Research*, vol. 54, no. 5, pp. 847–858, 2006.

[48] H. L. Li, H. C. Lu, C. H. Huang, and N. Z. Hu, "A superior representation method for piecewise linear functions," *INFORMS Journal on Computing*, vol. 21, no. 2, pp. 314–321, 2009.

[49] J. F. Tsai and M. H. Lin, "An efficient global approach for posynomial geometric programming problems," *INFORMS Journal on Computing*, vol. 23, no. 3, pp. 483–492, 2011.

[50] J. P. Vielma and G. L. Nemhauser, "Modeling disjunctive constraints with a logarithmic number of binary variables and constraints," *Mathematical Programming*, vol. 128, no. 1-2, pp. 49–72, 2011.

[51] J. Till, S. Engell, S. Panek, and O. Stursberg, "Applied hybrid system optimization: an empirical investigation of complexity," *Control Engineering Practice*, vol. 12, no. 10, pp. 1291–1303, 2004.

[52] J. F. Tsai and M. H. Lin, "Global optimization of signomial mixed-integer nonlinear programming problems with free variables," *Journal of Global Optimization*, vol. 42, no. 1, pp. 39–49, 2008.

[53] M. H. Lin and J. F. Tsai, "Range reduction techniques for improving computational efficiency in global optimization of signomial geometric programming problems," *European Journal of Operational Research*, vol. 216, no. 1, pp. 17–25, 2012.

[54] A. Lundell, *Transformation techniques for signomial functions in global optimization [Ph.D. thesis]*, Åbo Akademi University, 2009.

[55] A. Lundell and T. Westerlund, "On the relationship between power and exponential transformations for positive signomial functions," *Chemical Engineering Transactions*, vol. 17, pp. 1287–1292, 2009.

[56] A. Lundell, J. Westerlund, and T. Westerlund, "Some transformation techniques with applications in global optimization," *Journal of Global Optimization*, vol. 43, no. 2-3, pp. 391–405, 2009.