J Glob Optim (2008) 42:423–442
DOI 10.1007/s10898-008-9303-0

A method for approximating pairwise comparison matrices by consistent matrices

János Fülöp

Received: 11 March 2008 / Accepted: 13 March 2008 / Published online: 17 June 2008
© Springer Science+Business Media, LLC. 2008

Abstract In several methods of multiattribute decision making, pairwise comparison matrices are applied to derive implicit weights for a given set of decision alternatives. A class of the approaches is based on the approximation of the pairwise comparison matrix by a consistent matrix. In the paper this approximation problem is considered in the least-squares sense. In general, the problem is nonconvex and difficult to solve, since it may have several local optima. In the paper the classic logarithmic transformation is applied and the problem is transcribed into the form of a separable programming problem based on a univariate function with special properties. We give sufficient conditions of the convexity of the objective function over the feasible set. If such a sufficient condition holds, the global optimum of the original problem can be obtained by local search, as well. For the general case, we propose a branch-and-bound method. Computational experiments are also presented.

Keywords Pairwise comparison matrix · Consistent matrix · Nonconvex programming · Branch-and-bound

1 Introduction

In the paper we consider the following optimization problem:

$$\begin{aligned} \min \quad & \sum_{i=1}^{n}\sum_{j=1}^{n}\left(a_{ij}-\frac{w_i}{w_j}\right)^2 \\ \text{s.t.} \quad & \sum_{i=1}^{n} w_i = 1, \qquad (1) \\ & w_i > 0, \quad i = 1, \ldots, n, \end{aligned}$$

J. Fülöp (✉)
Laboratory of Operations Research and Decision Systems, Computer and Automation Research Institute, Hungarian Academy of Sciences, Kende u. 13-17, 1111 Budapest, Hungary
e-mail: fulop@sztaki.hu

where $A = [a_{ij}]$ is an $n \times n$ pairwise comparison matrix, i.e.

$$a_{ij} > 0 \quad \text{and} \quad a_{ij} = \frac{1}{a_{ji}}, \quad i, j = 1, \ldots, n. \tag{2}$$

Pairwise comparison matrices play an important role in multiattribute decision-making, they are applied to derive priorities or implicit weights for a given set of decision alternatives. Consider, as an example, the prioritization of n alternatives. The priorities represent the relative importance of the alternatives. In the approach based on pairwise comparisons, comparing any two alternatives i and j, the decision-maker assigns the value a_{ij} which represents a judgement concerning the relative importance of the preference of alternative i over alternative j. If alternative i is preferred to alternative j, then $a_{ij} > 1$. The positivity property $a_{ij} > 0$ and the reciprocity property $a_{ij} = 1/a_{ji}$ of (2) are evident assumptions on the pairwise comparisons.

A pairwise comparison matrix A is consistent if

$$a_{ij}a_{jk} = a_{ik}, \quad i, j, k = 1, \ldots, n.$$

It can be shown, see e.g. Saaty (1980), that a pairwise comparison matrix A is consistent if and only if there exists a positive n-vector w such that

$$a_{ij} = w_i/w_j, \quad i, j = 1, \ldots, n.$$

For a consistent pairwise comparison matrix A, the values w_i serve as priorities or implicit weights of the importance of alternatives.

In practice, the decision-maker's evaluations a_{ij} are frequently not consistent. In the case of an inconsistent pairwise comparison matrix A, the evaluations a_{ij} can be considered as perturbations of the appropriate elements of an $n \times n$ consistent pairwise comparison matrix $W = [w_{ij}]$, where

$$w_{ij} = w_i/w_j, \quad i, j = 1, \ldots, n,$$

and $w = (w_1, \ldots, w_n)^T$ is the vector of the priority weights.

Several approaches exist regarding how to derive a suitable vector w from an inconsistent pairwise comparison matrix A. Saaty (1977) proposed the Eigenvector Method in which w is the principal eigenvector of A. Another class of approaches is based on optimization methods and proposes different ways for minimizing the difference between the matrices A and W.

In the Least Squares Method presented by Chu et al. (1979), the matrix A is approximated by W in the least-squares sense. This optimization problem can be written in the form (1) where the constraint $\sum_{i=1}^{n} w_i = 1$ serves for the normalization of the vector w. The objective function of (1) is the Frobenius norm of the difference between the matrices A and W. Problem (1) is a difficult nonconvex optimization problem with several possible local optima, moreover, with possible multiple isolated global optimal solutions (Jensen 1983, 1984). Most of the methods proposed for solving (1) aim at finding local optimal solutions. Chu (1997), and Farkas and Rózsa (2004) apply local search techniques of nonlinear programming. Bozóki (2003, 2006), and Bozóki and Lewis (2005) transcribe (1) into the form of a multivariate polynomial system, and apply resultant and homothopy methods for finding the roots. Farkas (2004), Farkas and Rózsa (2001, 2004), and Farkas et al. (2003) use some techniques of linear algebra, and deal with questions of non-uniqueness and data perturbation, as well. Carrizosa and Messine (2007) propose an interval method for finding global optimal solutions of (1).

Some authors state that problem (1) has no special tractable form and is difficult to solve, see Chu et al. (1979), Golany and Kress (1993), Mikhailov (2000), Choo and Wedley (2004).

In order to elude the difficulties caused by the possible nonconvexity of (1), several other, more easily solvable problem forms are proposed to derive priority weights from an inconsistent pairwise comparison matrix. The Weighted Least Squares Method (Chu et al. 1979; Blankmeyer 1987) applies a convex quadratic optimization problem whose unique optimal solution is obtainable by solving a set of linear equations. The Logarithmic Least Squares Method (De Jong 1984; Crawford and Williams 1985) is based on an optimization problem whose unique optimal solution is the geometric mean of the rows of matrix A. The Goal Programming Method (Bryson 1995), the Chi Square Method (Jensen 1984), the Singular Value Decomposition (Gass and Rapcsák 2004), and the idea of using Support Vector Machines (Carrizosa 2006) are further approaches.

In the paper we focus on problem (1) and its equivalent forms. In Sect. 2 the classic logarithmic transformation is applied and the problem is transcribed into the form of a separable programming problem based on a class of univariate functions. Some special properties of these univariate functions are investigated, too. In Sect. 3 we give sufficient conditions for the global optimality of a local optimal solution. When such a sufficient condition holds, the global optimum of (1) can be obtained by local search. In Sect. 4 we propose a branch-and-bound method for solving (1) in the general case. Some computational experiments are presented in Sect. 5.

2 Transforming the Least Squares problem into a separable programming form

Instead of the normalization constraint $\sum_{i=1}^{n} w_i = 1$ used in (1), we apply the normalization $w_n = 1$, and write (1) into the equivalent form

$$\begin{aligned} \min \ & \sum_{i=1}^{n}\sum_{j=1}^{n}\left(a_{ij}-\frac{w_i}{w_j}\right)^2 \\ \text{s.t.} \ & w_n = 1, \\ & w_i > 0, \quad i=1,\ldots,n-1. \end{aligned} \tag{3}$$

Then, using the classic logarithmic transformation

$$t_i = \log w_i, \quad i=1,\ldots,n, \tag{4}$$

and the reciprocity property (2), problem (3) can be written into the equivalent form of the unconstrained problem

$$\begin{aligned} \min \ & \sum_{i=1}^{n-1}\left(\left(e^{t_i}-a_{in}\right)^2+\left(e^{-t_i}-1/a_{in}\right)^2\right)+ \\ & \sum_{i=1}^{n-2}\sum_{j=i+1}^{n-1}\left(\left(e^{t_i-t_j}-a_{ij}\right)^2+\left(e^{-t_i+t_j}-1/a_{ij}\right)^2\right). \end{aligned} \tag{5}$$

The logarithmic transformation (4) was applied by Chu (1997), as well. In general, nonlinear coordinate transformations like (4) are useful tools of convexification in optimization (Rapcsák 2001). By introducing additional variables $t_{ij},\ i=1,\ldots,n-2,\ j=i+1,\ldots,n-1$, (5) can be transcribed into the form of a separable programming problem:

$$\begin{aligned} \min \quad & \sum_{i=1}^{n-1} \left(\left(e^{t_i} - a_{in}\right)^2 + \left(e^{-t_i} - 1/a_{in}\right)^2 \right) + \\ & \sum_{i=1}^{n-2} \sum_{j=i+1}^{n-1} \left(\left(e^{t_{ij}} - a_{ij}\right)^2 + \left(e^{-t_{ij}} - 1/a_{ij}\right)^2 \right) \\ \text{s.t.} \quad & t_i - t_j - t_{ij} = 0, \ i = 1, \ldots, n-2, \ j = i+1, \ldots, n-1. \end{aligned} \tag{6}$$

Each of the univariate summands appearing in the objective function of (6) is an instance of the class of the univariate functions

$$f_a(t) = \left(e^t - a\right)^2 + \left(e^{-t} - 1/a\right)^2 \tag{7}$$

depending on the real parameter a. Consequently,

$$\begin{aligned} \min \quad & \sum_{i=1}^{n-1} f_{a_{in}}(t_i) + \sum_{i=1}^{n-2} \sum_{j=i+1}^{n-1} f_{a_{ij}}(t_{ij}) \\ \text{s.t.} \quad & t_i - t_j - t_{ij} = 0, \ i = 1, \ldots, n-2, \ j = i+1, \ldots, n-1 \end{aligned} \tag{8}$$

is an equivalent form of (6).

Although problem (5) is unconstrained and the feasible region of (8) is unbounded, suitable lower and upper bounds can easily be determined for the variables. These bounds are useful when applying, e.g., branch-and-bound techniques. Let γ be the value of the objective function of (5) at a point $(\bar{t}_1, \ldots, \bar{t}_{n-1})$. Then, from $(e^{t_i} - a_{in})^2 \le \gamma$, we get $t_i \le \log(\sqrt{\gamma} + a_{in})$. Determining the lower and upper bounds

$$\begin{aligned} l_i &= -\log\left(\sqrt{\gamma} + 1/a_{in}\right), \ i = 1, \ldots, n-1, \\ u_i &= \log\left(\sqrt{\gamma} + a_{in}\right), \ i = 1, \ldots, n-1, \\ l_{ij} &= -\log\left(\sqrt{\gamma} + 1/a_{ij}\right), \ i = 1, \ldots, n-2, \ j = i+1, \ldots, n-1, \\ u_{ij} &= \log\left(\sqrt{\gamma} + a_{ij}\right), \ i = 1, \ldots, n-2, \ j = i+1, \ldots, n-1 \end{aligned} \tag{9}$$

in a similar way, and adding the bound constraints on the variables to (8), we get

$$\begin{aligned} \min \quad & \sum_{i=1}^{n-1} f_{a_{in}}(t_i) + \sum_{i=1}^{n-2} \sum_{j=i+1}^{n-1} f_{a_{ij}}(t_{ij}) \\ \text{s.t.} \quad & t_i - t_j = t_{ij}, \ i = 1, \ldots, n-2, \ j = i+1, \ldots, n-1, \\ & l_i \le t_i \le u_i, \ i = 1, \ldots, n-1, \\ & l_{ij} \le t_{ij} \le u_{ij}, \ i = 1, \ldots, n-2, \ j = i+1, \ldots, n-1. \end{aligned} \tag{10}$$

It is clear that the bound constraints added (10) do not exclude any feasible solution of (8) with objective function value less than or equal to γ. An equivalent form of (10), without the additional variables t_{ij}, can be written as

$$\begin{aligned} \min \quad & \sum_{i=1}^{n-1} f_{a_{in}}(t_i) + \sum_{i=1}^{n-2} \sum_{j=i+1}^{n-1} f_{a_{ij}}(t_i - t_j) \\ \text{s.t.} \quad & l_i \le t_i \le u_i, \ i = 1, \ldots, n-1, \\ & l_{ij} \le t_i - t_j \le u_{ij}, \ i = 1, \ldots, n-2, \ j = i+1, \ldots, n-1. \end{aligned} \tag{11}$$

The function $f_a(t)$ of (7) plays an important role also in the equivalent forms above, therefore, its main properties are reviewed below.

It is clear that for any $a \neq 0$, we have

$$f_a(t) = f_{1/a}(-t), \tag{12}$$

i.e., the graph of $f_{1/a}$ can be obtained by reflecting the graph of f_a through the vertical axis. By using simple calculus, it is easy to see that for any $a > 0$, we have

$$\begin{aligned} &f_a(t) \geq 0 \ \text{ for all } t, \\ &\min f_a(t) = 0, \\ &\arg\min f_a(t) = \{\log a\}, \\ &f_a'(\log a) = 0, \\ &f_a'(t) < 0 \ \text{ for all } t < \log a, \\ &f_a'(t) > 0 \ \text{ for all } t > \log a. \end{aligned}$$

Proposition 1 *Let*

$$\bar{a} = \left(\frac{123 + 55\sqrt{5}}{2}\right)^{1/4}. \tag{13}$$

Then for any a with

$$1/\bar{a} \leq a \leq \bar{a}, \tag{14}$$

function f_a is strictly convex. For any

$$0 < a < 1/\bar{a} \quad \text{or} \quad \bar{a} < a, \tag{15}$$

function f_a has two inflexion points $t_a^{(1)} < t_a^{(2)}$, f_a is strictly concave on $[t_a^{(1)}, t_a^{(2)}]$ and strictly convex on $(-\infty, t_a^{(1)}]$ and $[t_a^{(2)}, \infty)$.

Proof From (7), we have

$$f_a''(t) = 2[2e^{2t} - ae^t + 2e^{-2t} - (1/a)e^{-t}].$$

The sign of $f_a''(t)$ determines the convex and concave parts of f_a. Consider the quartic polynomial

$$p_a(x) = 2x^4 - ax^3 - (1/a)x + 2. \tag{16}$$

By substituting

$$x = e^t, \tag{17}$$

we get

$$f_a''(t) = \frac{2}{x^2} p_a(x). \tag{18}$$

Since $x > 0$, the sign of $f_a''(t)$ is the same as the sign of $p_a(x)$.

The number of changes in sign in the sequence of the coefficients of polynomial $p_a(x)$ is 2 (zero coefficients are not counted). According to Descartes' rule of signs (Kurosh 1972, p. 247), the number of positive roots of $p_a(x)$ is 0 or 2 (counting with multiplicities).

It is easy to see that $\lim_{x\to 0} p_a(x) = 2$ and $\lim_{x\to\infty} p_a(x) = \infty$. Moreover, if $1 \leq a \leq x$, then $p_a(x) \geq 2$. To prove it, rearrange (16) as $p_a(x) = (x^4 - ax^3) + (x^4 - (1/a)x) + 2$. Since $x^4 \geq ax^3$ and $x^4 \geq x \geq (1/a)x$, we obtain $p_a(x) \geq 2$.

Assume now that $0 < x \leq a \leq 1$. Then $p_a(x) > 0$. Now, rearrange (16) as $p_a(x) = 2x^4 + (1 - ax^3) + (1 - (1/a)x)$. Since $1 \geq ax^3$ and $1 \geq (1/a)x$, we obtain $p_a(x) > 0$.

As a consequence of the computations above, we obtain that $p_1(x) > 0$ for all $x > 0$. Naturally, for any fixed $x > 0$, there exists an a such that $p_a(x) < 0$; we have to choose an a large enough or an $a > 0$ small enough.

Write (16) as the function of both x and a:

$$P(a, x) = 2x^4 - ax^3 - (1/a)x + 2.$$

Assume that for an $x > 0$ and $a > 1$, we have $P(a, x) \leq 0$. We know that $x < a$ holds in this case, hence

$$0 \geq 2x^4 - ax^3 - (1/a)x + 2 > 2x^4 - ax^3 - (1/a)x + (2/a)x > -ax^3 + (1/a)x.$$

From $0 > -ax^3 + (1/a)x$ and $a > 0$, we have

$$-x^3 + (1/a^2)x = \frac{\partial}{\partial a} P(a, x) < 0.$$

From the computations above, it follows that if for an $a > 1$ the polynomial $p_a(x)$ has a positive root, then $p_{\hat{a}}(x)$ has two positive roots for any $\hat{a} > a$.

From (16), we get $p'_a(x) = 8x^3 - 3ax^2 - (1/a)$. Taking $\lim_{x \to 0} p'_a(x) = -(1/a) < 0$ and $\lim_{x \to \infty} p'_a(x) = \infty$, as well as Decartes's rule of signs into consideration, it follows that for any $a > 0$, $p'_a(x) = 0$ has a unique solution for $x > 0$ and it is also the optimal solution of

$$\min \{p_a(x) \mid x > 0\}. \tag{19}$$

Now, we determine the unique $a > 1$ for which the optimal value of (19) is 0. This means that we have to solve the system

$$p_a(x) = 0, \;\; p'_a(x) = 0,$$

i.e.,

$$\begin{aligned} 2x^4 - ax^3 - (1/a)x + 2 = 0, \\ 8x^3 - 3ax^2 - (1/a) = 0. \end{aligned} \tag{20}$$

A polynomial system equivalent to (20) is

$$\begin{aligned} 2ax^4 - a^2x^3 - x + 2a = 0, \\ 8ax^3 - 3a^2x^2 - 1 = 0. \end{aligned} \tag{21}$$

By using the resultant method and eliminating x from (21) (see Kurosh (1972, p. 331) for more details) (21) can be reduced to solving

$$\begin{vmatrix} 2a & -a^2 & 0 & -1 & 2a & 0 & 0 \\ 0 & 2a & -a^2 & 0 & -1 & 2a & 0 \\ 0 & 0 & 2a & -a^2 & 0 & -1 & 2a \\ 8a & -3a^2 & 0 & -1 & 0 & 0 & 0 \\ 0 & 8a & -3a^2 & 0 & -1 & 0 & 0 \\ 0 & 0 & 8a & -3a^2 & 0 & -1 & 0 \\ 0 & 0 & 0 & 8a & -3a^2 & 0 & -1 \end{vmatrix} = 0. \tag{22}$$

After some computations, (22) can be transformed into the form

$$a^8 - 123a^4 + 1 = 0. \tag{23}$$

From (23), we have

$$a_{1,2}^4 = \frac{123 \pm \sqrt{123^2 - 4}}{2} = \frac{123 \pm \sqrt{15125}}{2} = \frac{123 \pm 55\sqrt{5}}{2}.$$

The positive roots are

$$a_{1,2} = \left(\frac{123 \pm 55\sqrt{5}}{2}\right)^{1/4},$$

but $a_{1,2} \geq 1$ holds only for $\bar{a}$ defined in (13). The approximate value of $\bar{a}$ is 3.330191.

Similar investigations can also be performed for $a < 1$ directly, but it is simpler to derive the results from (12) and use the findings for $a > 1$.

For any $0 < a < 1/\bar{a}$ or $\bar{a} < a$, the quartic polynomial $p_a(x)$ of (16) has two positive roots $x_a^{(1)} < x_a^{(2)}$, and they can easily be computed (see e.g. Kurosh, 1972). Furthermore, $p_a(x) < 0$ for all $x \in (x_a^{(1)}, x_a^{(2)})$ and $p_a(x) > 0$ for all $x \in (0, x_a^{(1)}) \cup (x_a^{(2)}, \infty)$. Let $t_a^{(i)} = \log x_a^{(i)}$, $i = 1, 2$. Then, according to (17) and (18), the second statement of the proposition follows immediately. □

The relation of function $f_a(t)$ to some quartic polynomials has already been exploited in the proof of Proposition 1. This relation can also be applied to compute lower and upper bounds better that those in (9). Again, as in (9), let γ be the value of the objective function of (5) at a feasible point. We can assume that $\gamma > 0$ since in case of $\gamma = 0$ we are done: γ is the optimal value of (1), moreover, matrix A is consistent. Consider the equation

$$f_{a_{ij}}(t) = \gamma \tag{24}$$

for each $i = 1, \ldots, n-1$, $j = i+1, \ldots, n$. From the properties of $f_a(t)$, it follows that each equation (24) has exactly two solutions l_{ij} and u_{ij} such that $l_{ij} < \log a_{ij} < u_{ij}$, and $f_{a_{ij}}(t) \leq \gamma$ if and only if $t \in [l_{ij}, u_{ij}]$. The solutions can easily be obtained: after substituting $x = e^t$ and rearranging, we have an equivalent form of (24),

$$x^4 - 2a_{ij}x^3 + (a_{ij}^2 + \frac{1}{a_{ij}^2} - \gamma)x^2 - \frac{2}{a_{ij}}x + 1 = 0 \tag{25}$$

that can be solved as a quartic polinomial equation. Thereafter, the solutions of (24) can be obtained from the positive roots of (25).

It is easy to see that the lower and upper bounds $l_i = l_{in}, u_i = u_{in}, i = 1, \ldots, n-1$ and $l_{ij}, u_{ij}, i = 1, \ldots, n-2, j = i+1, \ldots, n-1$, are better than those determined by (9). Namely, the feasible set of (11) with the bounds from (24) is a proper subset of the feasible set with the bounds from (9).

3 Sufficient conditions for the global optimality of a local optimal solution

Let

$$F(t_1, \ldots, t_{n-1}) = \sum_{i=1}^{n-1} f_{a_{in}}(t_i) + \sum_{i=1}^{n-2} \sum_{j=i+1}^{n-1} f_{a_{ij}}(t_i - t_j). \tag{26}$$

Function F is the objective function of problems (5) and (11), and plays an important role in this section. The value $\bar{a}$ from (13), approximately 3.330191, is also used here.

Proposition 2 *If*

$$0 < a_{ij} \leq \bar{a}, \quad i, j = 1, \ldots, n, \tag{27}$$

where $\bar{a}$ is from (13), then problem (1) has a unique local (thus global) optimal solution, the objective function of (11) is strictly convex and has a unique local (thus global) minimizer point.

Proof If (27) holds, then according to Proposition 1, each univariate function in (26) is strictly convex. The first part of (26), i.e.,

$$\sum_{i=1}^{n-1} f_{a_{in}}(t_i) \tag{28}$$

is strictly convex in $(t_1, \ldots, t_{n-1})$, and $F(t_1, \ldots, t_{n-1})$ remains strictly convex after adding the second part

$$\sum_{i=1}^{n-2} \sum_{j=i+1}^{n-1} f_{a_{ij}}(t_i - t_j)$$

of convex functions to (28). Since (26) is strictly convex and its lower level sets are compact (see (9)), problem (5) has a unique local thus global optimal solution. The convexity of the objective function does not necessarily hold for the original problems in form (1) and (3), but the nonlinear coordinate transformation (4) and its inverse assure the one-to-one correspondance between the local optimal solutions of the problems in the spaces of $(w_1, \ldots, w_n)$ and $(t_1, \ldots, t_{n-1})$, respectively. The statement for problem (11) follows evidently. □

Corollary 1 *If (27) holds, then the equivalent problems (1), (3), (5), (6), (8), (10) and (11) can be solved by local search techniques starting from any feasible point.*

□

Relation (27) is a sufficient but not necessary condition of the convexity of F. It can happen that the strict convexity of several univariate functions in (26) compensates small nonconvexities of some other univariate functions in (26), and hence F is convex. We need the Hessian of F and the quadratic form with it for such investigations. From (26), we obtain that

$$\begin{aligned}(x_1, \ldots, x_{n-1})\nabla^2 F(t_1, \ldots, t_{n-1})(x_1, \ldots, x_{n-1})^T \\ = \sum_{i=1}^{n-1} x_i \nabla^2 f_{a_{in}}(t_i) x_i + \sum_{i=1}^{n-2} \sum_{j=i+1}^{n-1} (x_i, x_j) \nabla^2 \bar{f}_{a_{ij}}(t_i, t_j)(x_i, x_j)^T\end{aligned} \tag{29}$$

for any $(n-1)$-vector $(x_1, \ldots, x_{n-1})$, where

$$\bar{f}_{a_{ij}}(t_i, t_j) = f_{a_{ij}}(t_i - t_j). \tag{30}$$

Equation (29) means that the quadratic form with the Hessian of F can be obtained from the quadratic forms with the 1×1 and 2×2 Hessians of the respective functions on the right-hand-side of (26). Based upon this property, we construct a quadratic matrix such that the quadratic form generated by this matrix underestimates the quadratic form generated by $\nabla^2 F$ at any feasible point of (11).

Clearly, $\nabla^2 f_{a_{in}}(t_i) = f''_{a_{in}}(t_i)$. Let

$$\mu_i = \min\{f''_{a_{in}}(t_i) \mid l_i \leq t_i \leq u_i\}. \tag{31}$$

Then,

$$x_i \nabla^2 f_{a_{in}}(t_i) x_i \geq \mu_i x_i^2 \tag{32}$$

for all $t_i \in [l_i, u_i]$ and for any x_i. Applying a technique of substitution similar to the one used in Proposition 1, the computation of μ_i can be reduced to finding the positive roots of a quartic polynomial. Namely,

$$f_a'''(t) = 2[4e^{2t} - ae^t - 4e^{-2t} + (1/a)e^{-t}].$$

Substituting $x = e^t$ again, we get

$$f_a'''(t) = \frac{2}{x^2}\bar{p}_a(x),$$

where

$$\bar{p}_a(x) = 4x^4 - ax^3 + (1/a)x - 4.$$

By determinig the positive roots of the quartic polynomial $\bar{p}_a(x)$, the real roots of $f_a'''(t)$ can also be obtained. Here again, finite methods can be used to solve the quartic polynomial equations (see e.g. Kurosh 1972).

Now, if the real roots of $f_{a_{in}}'''(t)$ are known, then taking the value of $f_{a_{in}}''(t)$ at the roots lying in $[l_i, u_i]$, and taking the values $f_{a_{in}}''(l_i)$ and $f_{a_{in}}''(u_i)$ also into consideration, μ_i of (31) can be obtained.

The computation of the Hessian of $\bar{f}_{a_{ij}}(t_i, t_j)$ of (30) is also simple. Let $\bar{t} = t_i - t_j$. Then

$$\frac{\partial^2 \bar{f}_{a_{ij}}(t_i, t_j)}{\partial t_i \partial t_i} = \frac{\partial^2 \bar{f}_{a_{ij}}(t_i, t_j)}{\partial t_j \partial t_j} = f_{a_{ij}}''(\bar{t}),$$

$$\frac{\partial^2 \bar{f}_{a_{ij}}(t_i, t_j)}{\partial t_i \partial t_j} = \frac{\partial^2 \bar{f}_{a_{ij}}(t_i, t_j)}{\partial t_j \partial t_i} = -f_{a_{ij}}''(\bar{t})$$

and

$$\nabla^2 \bar{f}_{a_{ij}}(t_i, t_j) = \begin{bmatrix} f_{a_{ij}}''(\bar{t}) & -f_{a_{ij}}''(\bar{t}) \\ -f_{a_{ij}}''(\bar{t}) & f_{a_{ij}}''(\bar{t}) \end{bmatrix}.$$

Let

$$\mu_{ij} = \min\{f_{a_{ij}}''(\bar{t}) \mid l_{ij} \leq \bar{t} \leq u_{ij}\}. \tag{33}$$

Again, μ_{ij} can be obtained by solving a quartic polynomial equation. Then

$$\begin{aligned} (x_i, x_j)\nabla^2 \bar{f}_{a_{ij}}(t_i, t_j)\begin{pmatrix} x_i \\ x_j \end{pmatrix} &= f_{a_{ij}}''(\bar{t})(x_i - x_j)^2 \\ \geq \mu_{ij}(x_i - x_j)^2 = (x_i, x_j)&\begin{bmatrix} \mu_{ij} & -\mu_{ij} \\ -\mu_{ij} & \mu_{ij} \end{bmatrix}\begin{pmatrix} x_i \\ x_j \end{pmatrix} \end{aligned} \tag{34}$$

for all $l_{ij} \leq t_i - t_j \leq u_{ij}$ and for any reals x_i and x_j.

Proposition 3 *Let H_{ij} be the $(n-1) \times (n-1)$ matrix having 1 in position (i, j) and zeros in all other positions. Let*

$$H = \sum_{i=1}^{n-1} \mu_i H_{ii} + \sum_{i=1}^{n-2} \sum_{j=i+1}^{n-1} \mu_{ij}(H_{ii} + H_{jj} - H_{ij} - H_{ji}). \tag{35}$$

If H is positive semidefinite, then the objective function of (11) is convex over the feasible set, (11) is a convex programming problem, thus, any local optimal solution of (11) is global optimal, too. If H is positive definite, then the objective function of (11) is strictly convex over the feasible set, (11) has a single local (thus global) optimal solution.

Proof It follows from (32) and (34) that for any $(n-1)$-vector $(x_1,\dots,x_{n-1})$ and for any feasible solution $(t_1,\dots,t_{n-1})$ of (11), we have

$$(x_1,\dots,x_{n-1})\nabla^2 F(t_1,\dots,t_{n-1})(x_1,\dots,x_{n-1})^T \geq (x_1,\dots,x_{n-1})H(x_1,\dots,x_{n-1})^T.$$

If H is positive semidefinite, then

$$(x_1,\dots,x_{n-1})H(x_1,\dots,x_{n-1})^T \geq 0,$$

hence

$$(x_1,\dots,x_{n-1})\nabla^2 F(t_1,\dots,t_{n-1})(x_1,\dots,x_{n-1})^T \geq 0.$$

This means that $\nabla^2 F(t_1,\dots,t_{n-1})$ is positive semidefinite over the feasible set of (11), and consequently, F is convex over the feasible set of (11). Since the feasible set of (11) is convex, (11) is a convex programming problem. The statement for positive definite H follows similarly. □

Corollary 2 *Consider the case when all l_i and l_{ij} are $-\infty$, and all u_i and u_{ij} are ∞ in (11), i.e. the feasible set of (11) is $\mathbb{R}^{n-1}$. Compute the values μ_i of (31) and μ_{ij} of (33) for this case. If the matrix H in (35) is positive semidefinite, then the equivalent problems (1), (3), (5), (6), (8), (10) and (11) can be solved by local search techniques starting from any feasible solution.*

4 A branch-and-bound method

In this section, we present a branch-and-bound method for solving (1). More precisely, the equivalent form (5) is considered, and its search region is restricted by adding lower and upper bound constraints on the variables. The bounds are computed, using (24), from the objective function value γ of a feasible solution of (5). In this way, problem (11) with a compact feasible set is obtained.

Problem (10) is equivalent to (11) and has a form of separable programming. This suggests, immediately, to apply a rectangular branch-and-bound technique for solving (10). However, (10) has $n(n-1)/2$ variables, while (11) has only $n-1$ variables. This is why we solve (11) but, essentially, we adapt, for (11), the rectangular branch-and-bound method detailed in Tuy (1998, Sects. 5.5–5.6).

The partition sets in the iterations of the branch-and-bound method have the same form as the feasible set of (11), the starting partition set, i.e. a partition set M is of form

$$\begin{aligned}\{(t_1,\dots,t_{n-1})\in\mathbb{R}^{n-1} \mid\ & l_i \leq t_i \leq u_i,\ i=1,\dots,n-1,\\ & l_{ij} \leq t_i - t_j \leq u_{ij},\ i=1,\dots,n-2,\ j=i+1,\dots,n-1\}.\end{aligned} \tag{36}$$

Clearly, the lower bounds l_i, l_{ij} and the upper bounds u_i, u_{ij} give an unambiguous description of the partition set. The partition sets are always assumed being nonempty.

The restriction of (11) to a partition set M means the problem

$$\begin{aligned}&\min\ F(t_1,\dots,t_{n-1})\\ &\text{s.t.}\ \ (t_1,\dots,t_{n-1})\in M,\end{aligned} \tag{37}$$

which, similar to (11), may be a nonconvex problem due to the possible nonconvexity of F in (26). According to the general branch-and-bound scheme, we have to determine a lower bound on the optimal value of (37). This is done by constructing a suitable piecewise linear convex underestimator for F over M, and by minimizing this underestimator over M. The underestimator is obtained from piecewise linear convex underestimators of functions $f_{a_{in}}$ and $f_{a_{ij}}$ appearing in (26).

Let $a > 0$ and consider the univariate function $f_a(t)$ over an interval $[l, u] \subset \mathbb{R}$. First, consider the case when $f_a(t)$ is convex over $[l, u]$. This happens when either (14), or (15) holds and (in the latter case) for the two inflexion points $t_a^{(1)} < t_a^{(2)}$, we have either $u \leq t_a^{(1)}$ or $t_a^{(2)} \leq l$. Let $l = \tau_1 < \tau_2 < \cdots < \tau_m = u$ be a finite partition of $[l, u]$, and let

$$g_{a,l,u}(t) = \max_{p=1,\ldots,m} \{f_a(\tau_p) + f_a^{'}(\tau_p)(t - \tau_p)\}. \tag{38}$$

Function $g_{a,l,u}(t)$ is piecewise linear and convex over $[l, u]$, and

$$g_{a,l,u}(t) \leq f_a(t) \;\text{ for all } t \in [l, u], \tag{39}$$

$$g_{a,l,u}(l) = f_a(l), \;\; g_{a,l,u}(u) = f_a(u), \tag{40}$$

$$|\, g_{a,l,u}(\bar{t}) - g_{a,l,u}(\hat{t})\,| \leq L_{a,l,u}\,|\bar{t} - \hat{t}\,| \;\text{ for all } \bar{t}, \hat{t} \in [l, u], \tag{41}$$

where

$$L_{a,l,u} = \max \{|\, f_a^{'}(t)\,| : t \in [l, u]\}. \tag{42}$$

Functions like (38) are used in separable convex programming, as well, see Burkard et al. (1991).

Now, consider the case when $f_a(t)$ is concave over $[l, u]$. This happens when (15) holds and we have $t_a^{(1)} \leq l < u \leq t_a^{(2)}$ for the two inflexion points. Let

$$g_{a,l,u}(t) = f_a(l) + \frac{f_a(u) - f_a(l)}{u - l}(t - l). \tag{43}$$

In the special case of $u = l$, let

$$g_{a,l,u}(t) = f_a(l).$$

Function $g_{a,l,u}(t)$ is linear, it is the convex envelop of $f_a(t)$ over $[l, u]$, and it fulfills the conditions (39)-(41), too.

Assume now that $f_a(t)$ is neither convex nor concave over $[l, u]$. Then, $f_a(t)$ consists of a concave part and one or two convex parts over $[l, u]$. Consider, for the sake of simplicity, the case of two convex parts. We have then $l < t_a^{(1)} < t_a^{(2)} < u$, $f_a(t)$ is convex over $[l, t_a^{(1)}]$ and $[t_a^{(2)}, u]$, and concave over $[t_a^{(1)}, t_a^{(2)}]$. In the same way as shown above, construct the convex envelop $g_{a,t_a^{(1)},t_a^{(2)}}(t)$ of $f_a(t)$ over $[t_a^{(1)}, t_a^{(2)}]$ by (43), and piecewise linear convex underestimators $g_{a,l,t_a^{(1)}}(t)$ and $g_{a,t_a^{(2)},u}(t)$ for $f_a(t)$ over $[l, t_a^{(1)}]$ and $[t_a^{(2)}, u]$ by (38), respectively. If $f_a(t)$ consists of only a convex and a concave part over $[l, u]$, then only the convex envelop of the concave part and a piecewise linear convex underestimator of the convex part are constructed. Thereafter, consider the piecewise linear function over $[l, u]$ obtained by putting together these two or three piecewise linear convex parts. The function constructed in this way is a nonconvex piecewise linear underestimator of $f_a(t)$ over $[l, u]$. Let $g_{a,l,u}(t)$ denote the convex envelop of this nonconvex function over $[l, u]$. Function $g_{a,l,u}(t)$ is piecewise linear and convex over $[l, u]$, and it is easy to be constructed, since its breaking points are a subset of those of the nonconvex function. It is also easy to see that $g_{a,l,u}(t)$ fulfills (39)–(41).

We mention that if $f_a(t)$ is not concave over $[l, u]$, $g_{a,l,u}(t)$ is not unambiguously determined. It depends on the number and the positions of the τ_p's used in (38). Obviously, if the τ_p's constitute an equidistant grid and their number tends to ∞, then the functions $g_{a,l,u}(t)$ converge to the convex envelop of $f_a(t)$ over $[l, u]$.

From the piecewise linear convex underestimators of the univariate functions in (26), we put together a piecewise linear convex underestimator of F over the partition set M. Let

$$G_M(t_1, \ldots, t_{n-1}) = \sum_{i=1}^{n-1} g_{a_{in},l_i,u_i}(t_i) + \sum_{i=1}^{n-2} \sum_{j=i+1}^{n-1} g_{a_{ij},l_{ij},u_{ij}}(t_i - t_j). \tag{44}$$

Then $G_M(t_1, \ldots, t_{n-1}) \le F(t_1, \ldots, t_{n-1})$ for all $(t_1, \ldots, t_{n-1}) \in M$. Consequently, the optimal value of

$$\begin{aligned} &\min\ G_M(t_1, \ldots, t_{n-1}) \\ &\text{s.t.}\ \ (t_1, \ldots, t_{n-1}) \in M \end{aligned} \tag{45}$$

is a lower bound of the optimal value of (37). Problem (45) can be reformulated as a linear programming problem. The univariate piecewise linear convex functions in (44) can be written as

$$\begin{aligned} g_{a_{in},l_i,u_i}(t) &= \max_{p=1,\ldots,m_{in}} \{\alpha_{inp} t + \delta_{inp}\},\ i = 1, \ldots, n-1, \\ g_{a_{ij},l_{ij},u_{ij}}(t) &= \max_{p=1,\ldots,m_{ij}} \{\alpha_{ijp} t + \delta_{ijp}\},\ i = 1, \ldots, n-2,\ j = i+1, \ldots, n-1. \end{aligned} \tag{46}$$

If the univariate function f_{a_{in},l_i,u_i} (or $f_{a_{ij},l_{ij},u_{ij}}$) is convex or concave over $[l_i, u_i]$ (or $[l_{ij}, u_{ij}]$), then (38) and (43) yield the forms (46) directly. In the convex case m_{in} (or m_{ij}) is the number of the points τ_p, in the concave case m_{in} (or m_{ij}) equals to 1. If the univariate function is neither convex nor concave, then m_{in} (or m_{ij}) depends on a_{in}, l_i, u_i (or a_{ij}, l_{ij}, u_{ij}) as well as on the number and positions of the τ_p's used at the convex part(s) written in the form of (38).

Then, (45) is equivalent to the linear programming problem

$$\begin{aligned} \min\ & \sum_{i=1}^{n-1} z_i + \sum_{i=1}^{n-2} \sum_{j=i+1}^{n-1} z_{ij} \\ \text{s.t.}\ & \alpha_{inp} t_i + \delta_{inp} - z_i \le 0,\ p = 1, \ldots, m_{in},\ i = 1, \ldots, n-1, \\ & \alpha_{ijp}(t_i - t_j) + \delta_{ijp} - z_{ij} \le 0,\ p = 1, \ldots, m_{ij},\ i = 1, \ldots, n-2, \\ & \quad j = i+1, \ldots, n-1, \\ & l_i \le t_i \le u_i,\ i = 1, \ldots, n-1, \\ & l_{ij} \le t_i - t_j \le u_{ij},\ i = 1, \ldots, n-2,\ j = i+1, \ldots, n-1, \end{aligned} \tag{47}$$

where $z_i, i = 1, \ldots, n-1$, and $z_{ij}, i = 1, \ldots, n-2,\ j = i+1, \ldots, n-1$, are additional variables.

For a partition set M, let $\beta(M)$ and $\omega(M)$ denote the optimal value and an optimal solution point of (45), respectively. Clearly, $\beta(M)$ is also the optimal value of (47), and the $(n-1)$-vector $\omega(M)$ is the $(t_1, \ldots, t_{n-1})$ part of an optimal solution of (47).

If $G_M(\omega(M)) = F(\omega(M))$, then we have found an optimal solution of (37), consequently, the partition set M needs not to be subdivided in a further step of the branch-and-bound method. Otherwise, we have

$$G_M(\omega(M)) < F(\omega(M)). \tag{48}$$

In this case M may be selected for subdivision (branching) in a further step. The ω-subdivision strategy, which is effective in rectangular algorithms, will be adapted for this purpose.

It follows from (48) that there exists at least one univariate function from the right-hand-side of (26) such that the gap between the function and its piecewise linear convex underestimator is positive at $\omega(M)$. We determine the function with the maximal gap, and use it at the ω-subdivision. Let

$$\begin{aligned}
&h_{in} = f_{a_{in}}(\omega_i(M)) - g_{a_{in},l_i,u_i}(\omega_i(M)),\ i = 1,\ldots,n-1,\\
&h_{ij} = f_{a_{ij}}(\omega_i(M) - \omega_j(M)) - g_{a_{ij},l_{ij},u_{ij}}(\omega_i(M) - \omega_j(M)),\\
&\qquad i = 1,\ldots,n-2,\ j = i+1,\ldots,n-1,\\
&(i_0, j_0) \in \arg\max\,\{h_{ij} \mid i = 1,\ldots,n-1,\ j = i+1,\ldots,n\},\\
&v = \begin{cases} \omega_{i_0}(M), & \text{for } j_0 = n,\\ \omega_{i_0}(M) - \omega_{j_0}(M), & \text{for } j_0 < n. \end{cases}
\end{aligned} \tag{49}$$

From (48), we have

$$h_{i_0 j_0} > 0. \tag{50}$$

Also, it follows from (50) and the property (40) of the univariate piecewise linear convex underestimators that

$$\begin{aligned}
l_{i_0} < v < u_{i_0} \quad & \text{for } j_0 = n,\\
l_{i_0 j_0} < v < u_{i_0 j_0} \quad & \text{for } j_0 < n.
\end{aligned}$$

Then, for $j_0 = n$, let

$$\begin{aligned}
M^{(1)} &= \{(t_1,\ldots,t_{n-1}) \in M \mid t_{i_0} \le v\},\\
M^{(2)} &= \{(t_1,\ldots,t_{n-1}) \in M \mid v \le t_{i_0}\},
\end{aligned} \tag{51}$$

and for $j_0 < n$, let

$$\begin{aligned}
M^{(1)} &= \{(t_1,\ldots,t_{n-1}) \in M \mid t_{i_0} - t_{j_0} \le v\},\\
M^{(2)} &= \{(t_1,\ldots,t_{n-1}) \in M \mid v \le t_{i_0} - t_{j_0}\}.
\end{aligned} \tag{52}$$

We will refer the partition (51)-(52) as *a partition via* (v, i_0, j_0).

The sets $M^{(1)}$ and $M^{(2)}$ are nonempty partition sets of form as that in (36), as well. They are created from M by modifying the lower or upper bound on t_{i_0} or $t_{i_0} - t_{j_0}$. This reduces the range of the possible values of t_{i_0} or $t_{i_0} - t_{j_0}$ over M. The reduction of the range of t_{i_0} or $t_{i_0} - t_{j_0}$ may cause the reduction of the possible range of other $t_{i_0} - t_j$ or $t_i - t_{j_0}$, and this may cause further reductions, and so on. We show an easy way how the tight ranges can be determined.

For a partition set M of form as in (36), let

$$\begin{aligned}
\bar{l}_i &= \min\,\{t_i \mid (t_1,\ldots,t_{n-1}) \in M\},\ i = 1,\ldots,n-1,\\
\bar{u}_i &= \max\,\{t_i \mid (t_1,\ldots,t_{n-1}) \in M\},\ i = 1,\ldots,n-1,\\
\bar{l}_{ij} &= \min\,\{t_i - t_j \mid (t_1,\ldots,t_{n-1}) \in M\},\ i = 1,\ldots,n-2,\ j = i+1,\ldots,n-1,\\
\bar{u}_{ij} &= \max\,\{t_i - t_j \mid (t_1,\ldots,t_{n-1}) \in M\},\ i = 1,\ldots,n-2,\ j = i+1,\ldots,n-1.
\end{aligned} \tag{53}$$

The values $\bar{l}_i, \bar{u}_i$ and $\bar{l}_{ij}, \bar{u}_{ij}$ are the tight bounds on t_i and $t_i - t_j$, respectively, over M. Obviously,

$$\begin{aligned}
&l_i \le \bar{l}_i \le \bar{u}_i \le u_i,\ i = 1,\ldots,n-1,\\
&l_{ij} \le \bar{l}_{ij} \le \bar{u}_{ij} \le u_{ij},\ i = 1,\ldots,n-2,\ j = i+1,\ldots,n-1.
\end{aligned}$$

Let

$$\bar{M} = \{(t_1, \ldots, t_{n-1}) \in \mathbb{R}^{n-1} \mid \bar{l}_i \le t_i \le \bar{u}_i,\ i = 1, \ldots, n-1,\\ \bar{l}_{ij} \le t_i - t_j \le \bar{u}_{ij},\ i = 1, \ldots, n-2,\ j = i+1, \ldots, n-1\}.$$

Clearly, $\bar{M} = M$. This means that by changing the bounds of M to the tight bounds, the set itself does not change. Applying tight bounds on the variables is, however, very advantageous when a lower bound is to be computed on the optimal value of problem (37). It is easy to see that the piecewise linear convex underestimators $g_{a,l,u}$ may give better approximations if the bounds l and u are tight.

Range reduction and the computation of bounds as tight as possible are very useful tools in the branch-and-bound methods of global optimization, and several techniques have been developed for this purpose, see Tawarmalani and Sahinidis (2002). In principle, linear programming problems should be solved to obtain the bounds in (53). However, due to the special structure of the partition sets, the values of (53) can be obtained by solving a shortest path problem in a graph.

Let $\mathcal{G} = [\mathcal{N}, \mathcal{A}]$ denote a directed graph, where $\mathcal{N}$ is the set of nodes and $\mathcal{A}$ is the set of arcs. Let $\mathcal{N} = \{1, \ldots, n\}$ and $\mathcal{A} = \{(i, j) \mid i \in \mathcal{N}, j \in \mathcal{N}, i \ne j\}$. A weight c_{ij} is associated with each arc:

$$c_{ij} = \begin{cases} u_{ij}, & \text{for } i < j < n, \\ -l_{ji}, & \text{for } j < i < n, \\ u_i, & \text{for } i < j = n, \\ -l_j, & \text{for } j < i = n. \end{cases} \tag{54}$$

Note that c_{ij} may be nonnegative.

Proposition 4 *Consider the graph $\mathcal{G} = [\mathcal{N}, \mathcal{A}]$ with weights of arcs as defined in (54). Then, $\mathcal{G}$ does not contain negative-weight cycle. Furthermore, for any $i_1, j_1 \in \mathcal{N}$, let $d_{i_1 j_1}$ denote the weight of the shortest path from i_1 to j_1. Then, for the tight bounds in (53), we have*

$$\begin{aligned} &\bar{l}_i = -d_{ni},\ i = 1, \ldots, n-1, \\ &\bar{u}_i = d_{in},\ i = 1, \ldots, n-1, \\ &\bar{l}_{ij} = -d_{ji},\ i = 1, \ldots, n-2,\ j = i+1, \ldots, n-1, \\ &\bar{u}_{ij} = d_{ij},\ i = 1, \ldots, n-2,\ j = i+1, \ldots, n-1. \end{aligned} \tag{55}$$

Proof Consider, as a primal problem, the combinatorial problem of finding the shortest path in $\mathcal{G}$ from $i_1 \in \mathcal{N}$ to $j_1 \in \mathcal{N}$. The dual of the problem can be written as the linear program

$$\begin{aligned} &\max\ y_{i_1} - y_{j_1} \\ &\text{s.t.}\quad y_i - y_j \le c_{ij}\ \ \text{for all } (i, j) \in \mathcal{A}, \end{aligned} \tag{56}$$

see. e.g., Papadimitriou and Steiglitz (1982). Rearranging the constraints of (56), and taking (54) also into consideration, we obtain

$$\begin{aligned} \max\ & y_{i_1} - y_{j_1} \\ \text{s.t.}\ & y_i - y_j \le u_{ij},\ i = 1, \ldots, n-2,\ j = i+1, \ldots, n-1, \\ & y_j - y_i \le -l_{ij},\ i = 1, \ldots, n-2,\ j = i+1, \ldots, n-1, \\ & y_i - y_n \le u_i,\ i = 1, \ldots, n-1, \\ & y_n - y_j \le -l_j,\ j = 1, \ldots, n-1. \end{aligned} \tag{57}$$

It is easy to see that the value of one of the variables can arbitrarily be chosen in (57), so let $y_n = 0$. Then, (57) can be reformulated as

$$\begin{aligned} &\max\ y_{i_1} - y_{j_1} \\ &\text{s.t.}\ \ l_i \le y_i \le u_i,\ i = 1, \ldots, n-1, \\ &\qquad l_{ij} \le y_i - y_j \le u_{ij},\ i = 1, \ldots, n-2,\ j = i+1, \ldots, n-1, \end{aligned} \tag{58}$$

where, in the case of $i_1 = n$ or $j_1 = n$, y_n is left out from the objective function.

Note that the feasible set of (58) is just the partition set M. Since $M \neq \emptyset$, (58), and thus (56) as the dual of the shortest path problem, have feasible solution. Consequently, the optimal value of the primal problem is finite for any $i_1 \in \mathcal{N}$ and $j_1 \in \mathcal{N}$. This also means that $\mathcal{G}$ does not contain negative-weight cycle.

By the duality theorem, the optimal value of problem (58) is $d_{i_1 j_1}$. Let $i_1 < j_1 < n$. Then, from (53) and (58), we get $\bar{u}_{i_1 j_1} = d_{i_1 j_1}$. For $j_1 < i_1 < n$, the maximization of $y_{i_1} - y_{j_1}$ in (58) can be replaced by the minimization of $y_{j_1} - y_{i_1}$. Again, from (53), we get $\bar{l}_{j_1 i_1} = -d_{i_1 j_1}$. The remainder of (55) can be proved in a similar way. □

Corollary 3 *Since $\mathcal{G}$ does not contain negative-weight cycle, the weights of the shortest paths between all pairs of nodes can be determined by the Floyd-Warshall algorithm in $O(n^3)$ steps, see, e.g., Papadimitriou and Steiglitz (1982). Consequently, all tight bounds in (53) can also be determined in $O(n^3)$ steps.*

Now, we present the algorithm proposed for solving problem (5).

Algorithm 1
Select an $\varepsilon \ge 0$.
Initialization. Let $\bar{t}^0$ be the best feasible solution available for (5), and let $\gamma_0 = F(\bar{t}^0)$. With $\gamma = \gamma_0$, determine the lower and upper bounds as the solutions of (24), and construct an initial partition set M_0 of form (36). Let $\mathcal{P}_1 = \mathcal{S}_1 = \{M_0\}$. Set $k = 1$.

Step 1. For each $M \in \mathcal{P}_k$ construct a piecewise linear convex underestimator G_M according to (44) and (46), and solve (45) via solving the equivalent linear programming problem (47). Let $\beta(M)$ and $\omega(M)$ be the optimal value and an optimal solution point of (45), respectively.
Step 2. Update the incumbent by setting $\bar{t}^k$ equal to the best among $\bar{t}^{k-1}$ and all $\omega(M)$, $M \in \mathcal{P}_k$. Let $\gamma_k = F(\bar{t}^k)$.
Step 3. Delete every $M \in \mathcal{S}_k$ such that $\beta(M) \ge \gamma_k - \varepsilon$. Let $\mathcal{R}_k$ be the collection of remaining members of $\mathcal{S}_k$.
Step 4. If $\mathcal{R}_k = \emptyset$, then terminate: $\bar{t}^k$ is a global ε-optimal solution of (5).
Step 5. Choose $M_k \in \arg\min\{\beta(M) \mid M \in \mathcal{R}_k\}$. Subdivide M_k by a partition via (v, i_0, j_0) according to (49)-(52). Let $\mathcal{P}_{k+1}$ be the partition of M_k.
Step 6. For each partition set $M \in \mathcal{P}_{k+1}$, determine the tight bounds according to (55), and replace the bounds by the tight bounds.
Step 7. Set $\mathcal{S}_{k+1} = (\mathcal{R}_k \setminus \{M_k\}) \cup \mathcal{P}_{k+1}$. Set $k \leftarrow k+1$ and go back to Step 1.

Proposition 5 *Algorithm 1 can make an infinite number of iterations only if $\varepsilon = 0$ and in this case every accumulation point of the sequence $\{\bar{t}^k\}$ is a global optimal solution of (5).*

Proof If Algorithm 1 is infinite, then there exists at least one infinite sequence of sets M_{k_ν}, $\nu = 1, 2, \ldots$, such that for $\nu > 1$ each M_{k_ν} is a child of its predecessor $M_{k_{\nu-1}}$, i.e. M_{k_ν} is obtained from $M_{k_{\nu-1}}$ directly by a subdivision. Infinite sequences of partition sets with this property are called filters. For a general class of branch-and-bound algorithms, Tuy (1998, Sections 5.5-5.6) proved that if every filter $\{M_k, k \in K\}$ contains an infinite nested sequence $\{M_k, k \in K_1\}$ such that

$$\gamma_k - \beta(M_k) \to 0 \quad (k \to \infty, k \in K_1), \tag{59}$$

then the algorithm is convergent. Since Algorithm 1 is a special case of the general class of branch-and-bound algorithms being in the focus of Tuy (1998, Sections 5.5-5.6), we shall prove (59) and refer to Tuy (1998).

Consider a filter $\{M_k, k \in K\}$. It is clear that there exist $1 \leq i_1 < j_1 \leq n$, $s \in \{1, 2\}$ and an infinite subset $K_1 \subset K$ such that every $M_k, k \in K_1$ is subdivided by a partition via (v^k, i_1, j_1) and its child in the filter $\{M_k, k \in K\}$ has the form $M^{(s)}$ of (51) or (52). Assume, without loss of generality, that $i_1 = 1$, $j_1 = n$ and $s = 1$. Then, for any $k_1, k_2 \in K_1$ for which $k_1 < k_2$, we get

$$l_1^{k_1} \leq l_1^{k_2} \leq u_1^{k_2} \leq \omega_1(M_{k_1}) < u_1^{k_1}, \tag{60}$$

where l_1^k and u_1^k denote the tight bounds of t_1 in the partition set M_k. From (60), it follows that there exist $\hat{l}_1$ and $\hat{u}_1$ such that

$$l_1^k \to \hat{l}_1, \;\; u_1^k \to \hat{u}_1, \quad \omega_1(M_k) \to \hat{u}_1 \;\; (k \to \infty, \; k \in K_1). \tag{61}$$

To prove (59), we show first that

$$f_{a_{1n}}(\omega_1(M_k)) - g_{a_{1n},l_1^k,u_1^k}(\omega_1(M_k)) \to 0 \quad (k \to \infty, \; k \in K_1). \tag{62}$$

Clearly,

$$\begin{aligned} & f_{a_{1n}}(\omega_1(M_k)) - g_{a_{1n},l_1^k,u_1^k}(\omega_1(M_k)) \\ & \quad = (f_{a_{1n}}(\omega_1(M_k)) - f_{a_{1n}}(u_1^k)) + (f_{a_{1n}}(u_1^k) - g_{a_{1n},l_1^k,u_1^k}(\omega_1(M_k)). \end{aligned} \tag{63}$$

From (61) and the continuity of f_{a_1}, we get

$$f_{a_{1n}}(\omega_1(M_k)) - f_{a_{1n}}(u_1^k) \to 0 \quad (k \to \infty, \; k \in K_1). \tag{64}$$

From (40), it follows that

$$f_{a_{1n}}(u_1^k) = g_{a_{1n},l_1^k,u_1^k}(u_1^k), \tag{65}$$

and from (41) and (65),

$$| f_{a_{1n}}(u_1^k) - g_{a_{1n},l_1^k,u_1^k}(\omega_1(M_k)) | \leq L_{a_{1n},l_1^k,u_1^k} \, | u_1^k - \omega_1(M_k) |,$$

where the Lipschitz constant L is defined in (42). Since for any $k_1, k_2 \in K_1$ and $k_1 < k_2 \in K_1$, we have

$$L_{a_{1n},l_1^{k_1},u_1^{k_1}} \geq L_{a_{1n},l_1^{k_2},u_1^{k_2}} \geq 0,$$

as well as $u_1^k - \omega_1(M_k) \to 0$ $(k \to \infty, \; k \in K_1)$, it follows that

$$f_{a_{1n}}(u_1^k) - g_{a_{1n},l_1^k,u_1^k}(\omega_1(M_k)) \to 0 \quad (k \to \infty, \; k \in K_1). \tag{66}$$

From (63), (64), and (66), we obtain (62). Then, (62) entails

$$F(\omega(M_k)) - G_{M_k}(\omega(M_k)) \to 0 \quad (k \to \infty, \; k \in K_1),$$

and taking $F(\omega(M_k)) \geq \gamma_k \geq \beta(M_k) = G_{M_k}(\omega(M_k))$ also into account, (59) follows immediately. This completes the proof. □

Corollary 4 *Let*

$$\bar{w}_i^k = \begin{cases} e^{\bar{t}_i^k}/(1+\sum_{j=1}^{n-1} e^{\bar{t}_j^k}), & \text{for } i = 1,\ldots,n-1, \\ 1/(1+\sum_{j=1}^{n-1} e^{\bar{t}_j^k}), & \text{for } i = n. \end{cases}$$

Then, if Algorithm 1 is infinite, every accumulation point of the sequence $\{\bar{w}^k\}$ is a global optimal solution of (1). Similarly, if $\bar{t}^k$ is a global ε-optimal solution of (5), so is $\bar{w}^k$ for (1).

Algorithm 1 differs from the methods published earlier for finding the global optimum of problem (1). Bozóki (2003, 2006), and Bozóki and Lewis (2005) wrote the first-order necessary condition for (1) in the form of a multivariate polynomial system, and applied resultant and homothopy methods for finding the roots. This approach is capable for finding all local optima of (1). This needs, however, more computational efforts in comparison to Algorithm 1 that finds only the global optimal solutions of problem (1).

Carrizosa and Messine (2007) showed that standard interval branch-and-bound algorithms can be directly used for solving problem (1) and even more general problems, as well. They consider the general problem

$$\begin{aligned} \min \quad & g\left((|a_{ij} - \tfrac{w_i}{w_j}|)_{i,j=1}^n\right) \\ \text{s.t.} \quad & w_i > 0, \quad i = 1,\ldots,n, \end{aligned} \tag{67}$$

where g is a monotonic norm in the nonnegative orthant $\mathbb{R}_+^{n\times n}$. Clearly, if g is the l_2 norm, then (67) is equivalent to problem (1). Carrizosa and Messine (2007) extended (67) also for the case when the scalar values a_{ij} are replaced by intervals. In the case of problem (1), the standard interval branch-and-bound algorithms do not exploit the computational advantages derived from the special structure of problem (1) as detailed in the present paper. This methodological advantage of Algorithm 1 over the standard interval techniques is valid only for problem (1). For monotonic norms g different from l_2 as well as for the case when intervals are given instead of scalar values a_{ij}, a branch-and-bound method similar to Algorithm 1 may be competitive with the standard interval techniques only if special properties similar to those in this paper can be found and exploited.

5 Computational experiments

Algorithm 1 has been tested on randomly generated problems. The algorithm was coded in C and run on an Intel Pentium 4 PC (3.2 GHz, 2 GB of RAM). The linear programming problems (47) were solved by the package BPMPD developed by Mészáros (1999). The unique optimal solution of the Logarithmic Least Squares Method was used as a starting feasible solution $\bar{t}^0$ since it can directly be obtained as the geometric mean of the rows of A. The tolerance $\varepsilon = 10^{-3}$ was used in the course of the computational experiments.

We applied the following strategy to construct piecewise linear convex underestimators. If at least one of the univariate functions of (26) was nonconvex over the partition set, then the convex univariate functions $f_a(t)$ were approximated by (38) using $m = 2$ or $m = 3$. The endpoints of the interval were chosen as τ's, in addition, if $\log a$ was in the interval, then in order to assure the nonnegativity of the underestimation, $\log a$ was used as a τ, too. If all of the univariate functions were convex over the partition set, an equidistant grid of τ's with $m = 20$ was used for each function. By the latter approximation, an iteration of the algorithm

Table 1 Randomly generated test problems

n	p	Average		Minimal		Maximal	
		Time (s)	Subdivisions	Time (s)	Subdivisions	Time (s)	Subdivisions
5	20	0.05	3.50	0.01	0	0.25	19
5	40	0.27	19.90	0.06	4	0.66	48
5	60	0.29	21.40	0.05	3	0.67	47
5	80	0.38	29.75	0.17	13	0.61	51
6	20	0.43	24.25	0.03	1	0.94	52
6	40	0.86	46.70	0.42	21	2.39	124
6	60	1.17	64.70	0.56	32	2.36	131
6	80	1.35	79.65	0.67	38	2.45	148
7	20	2.86	118.45	0.75	33	5.86	234
7	40	3.16	126.15	1.22	49	6.44	276
7	60	3.47	147.85	1.52	62	6.63	286
7	80	5.34	241.45	2.42	108	10.17	491
8	20	8.60	268.45	3.16	97	16.75	527
8	40	9.72	302.70	2.75	88	25.09	764
8	60	11.38	374.80	5.70	187	20.16	666
8	80	24.39	880.90	3.94	127	66.38	2761
9	20	32.02	772.70	7.13	179	72.31	1782
9	40	23.65	586.00	10.36	257	93.30	2224
9	60	32.78	849.85	11.19	274	82.17	2264
9	80	58.65	1650.65	13.23	350	218.95	6537
10	20	99.28	1893.45	29.92	568	249.02	4733
10	40	78.92	1567.55	23.25	466	341.34	6971
10	60	91.51	1879.35	21.25	445	197.78	4043
10	80	220.95	4856.45	98.02	2069	539.13	12078

proposed by Burkard et al. (1991) for solving separable convex programming problems was performed. Another approach can be to find the minimum of the convex function over the partition set directly by using an iterative local optimization technique. It may however turn out that the minimal value of the convex function over the partition set is not better than the objective function value of the incumbent, thus the partition set is deleted, making the additional computational efforts wasted.

The randomly generated test problems were created in a way similar as Golany and Kress (1993), and Carrizosa and Messine (2007) did. For a fixed n, a weight vector $\bar{w} = (\bar{w}_1, \ldots, \bar{w}_n)^T$ was generated randomly, where each $\bar{w}_i$ was selected independently and uniformly from the set $\{1, \ldots, 9\}$. The entries of the consistent matrix of ratios $\bar{w}_i/\bar{w}_j$ were then perturbated by using a parameter p (given in %) and a perturbation factor ξ_{ij} randomly generated from a uniform distribution in the interval $[-1, 1]$. Then, the entries of matrix A were obtained as

$$a_{ij} = \begin{cases} \frac{\bar{w}_i}{\bar{w}_j}(1 + \frac{p}{100}\xi_{ij}), & \text{for } i < j, \\ 1, & \text{for } i = j, \\ 1/a_{ji}, & \text{for } i > j. \end{cases}$$

The parameter p is referred as 'degree of inconsistency' in Golany and Kress (1993). Clearly, a greater value of p allows a greater possible deviation from the value of the given entry of the consistent matrix.

Table 1 summarizes the computational experiments on randomly generated test problems. The test problems were generated for $n = 5, \ldots, 10$ and $p = 20, 40, 60, 80$. For each category, 20 problems were generated and solved. In Table 1, the average, the minimal and the maximal values of the running times and the numbers of subdivisions are listed in each category.

We can observe the usual phenomenon of the branch-and-bound methods, namely, if n increases, then the computational efforts needed to solve the problems increase more or less exponentially. Also, except for some categories, considering a fixed n, a greater value of p entails greater computational efforts. A more precise and detailed analysis of the relation of the consistency ratio (Saaty 1980) and other measures of inconsistency to the convexity and nonconvexity properties of the problems considered in the paper will be the topic of further research.

Acknowledgements The author would like to thank the anonymous referees for their helpful remarks and suggestions. This research was supported, in part, by the Hungarian Scientific Research Fund, Grant Nos. OTKA-T043276, T043241 and K60480.

References

Blankmeyer, E.: Approaches to consistency adjustments. J. Optim. Theory Appl. **54**, 479–488 (1987)

Bozóki, S.: A method for solving the LSM problems of small size in the AHP. Cent. Eur. J. Oper. Res. **11**, 17–33 (2003)

Bozóki, S.: Weights from the least squares approximation of pairwise comparison matrices (in Hungarian). Alkalmazott Matematikai Lapok. **23**, 121–137 (2006)

Bozóki, S.: Weights from pairwise comparisons and evaluation by using utility functions in multi-attribute decision problems. Ph.D. Dissertation, Corvinus University of Budapest, Budapest (2006)

Bozóki, S., Lewis, R.H.: Solving the Least Squares Method problem in the AHP for 3 × 3 and 4 × 4 matrices. Cent. Eur. J. Oper. Res. **13**, 255–270 (2005)

Bryson, N.: A goal programming method for generating priority vectors. J. Oper. Res. Soc. **46**, 641–648 (1995)

Burkard, R.E., Hamacher, H.W., Rote, G.: Sandwich approximation of univariate convex functions with an application to separable convex programming. Nav. Res. Log. **38**, 911–924 (1991)

Carrizosa, E.: Deriving weights in multiple-criteria decision making with support vector machines. TOP **14**, 399–424 (2006)

Carrizosa, E., Messine, F.: An exact global optimization method for deriving weights from pairwise comparison matrices. J. Glob. Optimi. **38**, 237–247 (2007)

Choo, E.U., Wedley, W.C.: A common framework for deriving preference values from pairwise comparison matrices. Comp. Oper. Res. **31**, 893–908 (2004)

Chu, M.T.: On the optimal consistent approximation to pairwise comparison matrices. Linear Algebra Appl. **272**, 155–168 (1997)

Chu, A.T.W., Kalaba, R.E., Spingarn, K.: A comparison of two methods for determining the weight belonging to fuzzy sets. J. Optim. Theory Appl. **4**, 531–538 (1979)

Crawford, G., Williams, C.: A note on the analysis of subjective judgment matrices. J. Math. Psychol. **29**, 387–405 (1985)

De Jong, P.: A statistical approach to Saaty's scaling method for priorities. J. Math. Psychol. **28**, 467–478 (1984)

Farkas, A.: Cardinal Measurement of Consumer's Preferences. Ph.D. Dissertation, Budapest University of Technology and Economics, Budapest (2004)

Farkas, A., Rózsa, P.: Data perturbations of matrices of pairwise comparisons. Ann. Oper. Res. **101**, 401–425 (2001)

Farkas, A., Rózsa, P.: On the non-uniqueness of the solution to the least-squares optimization of pairwise comparison matrices, Acta Polytech. Hung. J. Appl. Sci. at Budapest Tech Hungary **1**, 1–20 (2004)

Farkas, A., Lancaster, P., Rózsa, P.: Consistency adjustment for pairwise comparison matrices. Numerical Linear Algebra Appl. **10**, 689–700 (2003)

Gass, S.I., Rapcsák, T.: Singular value decomposition in AHP. Eur. J. Oper. Res. **154**, 573–584 (2004)

Golany, B., Kress, M.: A multicriteria evaluation method for obtaining weights from ratio-scale matrices. Eur. J. Oper. Res. **69**, 210–220 (1993)

Jensen, R.E.: Comparison of eigenvector, least squares, chi squares and logarithmic least squares methods of scaling a reciprocal matrix. Working paper 153, Trinity University (1983)

Jensen, R.E.: Alternative scaling method for priorities in hierarchical structures. J. Math. Psychol. **28**, 317–332 (1984)

Kurosh, A.G.: Higher Algebra. Mir Publishers, Moscow (1972)

Mészáros, Cs.: BPMPD version 2.21. Optimization Methods and Software. **11-12**, CD Supplement, Software section (1999) (http://www.sztaki.hu/~meszaros/bpmpd/)

Mikhailov, L.: A fuzzy programming method for deriving priorities in the analytic hiarerchy process. J. Oper. Res. Soc. **51**, 341–349 (2000)

Papadimitriou, C.H., Steiglitz, K.: Combinatorial Optimization: Algorithms and Complexity. Pren-tice-Hall, Englewood Cliffs (1982)

Rapcsák, T.: Convexification of functions by nonlinear coordinate transformations. In: Giannessi, F., Pardalos, P., Rapcsák, T. (eds.) Optimization Theory, pp. 179–189. Kluwer (2001)

Saaty, T.L.: A scaling method for priorities in hierarchical structures. J. Math. Psychol. **15**, 234–281 (1977)

Saaty, T.L.: The Analytic Hierarchy Process. McGraw-Hill, New York (1980)

Tawarmalani, M, Sahinidis, N.V.: Convexification and Global Optimization in Continuous and Mixed-integer Nonlinear Programming. Kluwer, Dordrecht (2002)

Tuy, H.: Convex Analysis and Global Optimization. Kluwer, Dordrecht (1998)