Journal of the Operations Research
Society of Japan
Vol. 36, No. 1, March 1993

A PRACTICAL APPROACH TO DECOMPOSABLE NONLINEAR PROGRAMMING PROBLEMS

Tatsuro Wakahara — *Toho Gas Co., Ltd.*
Masao Fukushima — *Advanced Institute of Science and Technology, Nara*
Toshihide Ibaraki — *Kyoto University*

(Received January 10, 1992; Final November 24, 1992)

Abstract Nonlinear programming problems often contain two types of variables: one appears linearly, while the other nonlinearly. The purpose of this paper is to propose a practical decomposition approach for solving nonlinear programming problems in which the number of linear variables is presumably much larger than that of nonlinear variables. Using quadratic penalty and quadratic perturbation techniques, we first formulate a parametric approximate problem for the given problem. By reformulating the approximate problem, we then obtain a differentiable nonlinear programming problem containing the nonlinear variables only. The main advantage of this approach is that any available nonlinear programming code may be used to solve the last problem, of which variables are presumably much fewer than the original problem. We may thus avoid solving the given problem directly or developing a specialized algorithm like Benders' algorithm that essentially deals with a nonsmooth optimization problem equivalent to the original problem. We give error bounds for the approximate problem and mention a possibility of parallel decomposition for a class of structured problems. Some computational results indicate that the proposed approach is practically useful.

1. Introduction

Nonlinear programming problems often contain a relatively small number of nonlinear variables. We may formulate such problems in the following form with nonlinear variables $\mathbf{x} \in R^n$ and linear variables $\mathbf{y} \in R^N$:

$$\begin{array}{lll} \text{minimize}_{\mathbf{x},\mathbf{y}} & f(\mathbf{x}) + \mathbf{c}^T\mathbf{y} & \\ \text{subject to} & g_i(\mathbf{x}) + \mathbf{a}_i^T\mathbf{y} \leq b_i, & i = 1,\ldots,\ell, \\ & h_j(\mathbf{x}) \leq 0, & j = 1,\ldots,m, \end{array} \tag{1.1}$$

where f: $R^n \to R$, g_i: $R^n \to R$, $i = 1,\ldots,\ell$, and h_j: $R^n \to R$, $j = 1,\ldots,m$, are twice continuously differentiable functions, $\mathbf{c}$ and $\mathbf{a}_i$, $i = 1,\ldots,\ell$, are N-dimensional constant vectors, and T denotes transposition. In the following, we shall often denote $\mathbf{g}(\mathbf{x}) = (g_1(\mathbf{x}),\ldots,g_\ell(\mathbf{x}))^T$, $\mathbf{h}(\mathbf{x}) = (h_1(\mathbf{x}),\ldots,h_m(\mathbf{x}))^T$, $\mathbf{A} = [\mathbf{a}_1,\ldots,\mathbf{a}_\ell]$ and $\mathbf{b} = (b_1,\ldots,b_\ell)^T$.

To solve problem (1.1), it is often useful, as in Benders' decomposition method [2, 9], to treat the linear part as a linear programming problem by temporarily fixing the nonlinear variables $\mathbf{x}$. As a result, we obtain the following nonlinear programming problem containing the variables $\mathbf{x}$ only:

$$\begin{array}{ll} \text{minimize} & f(\mathbf{x}) + \phi(\mathbf{x}) \\ \text{subject to} & \mathbf{h}(\mathbf{x}) \leq \mathbf{0}, \end{array} \tag{1.2}$$

where the function ϕ: $R^n \to R \cup \{+\infty\}$ is defined by

$$\phi(\mathbf{x}) = \inf_{\mathbf{y}}\{\mathbf{c}^T\mathbf{y} \mid \mathbf{A}^T\mathbf{y} \leq \mathbf{b} - \mathbf{g}(\mathbf{x})\}. \tag{1.3}$$

Note that the problem on the right-hand side of (1.3) may be infeasible for some $\mathbf{x}$, in which case we define $\phi(\mathbf{x}) = +\infty$. We may therefore regard the set $\{\mathbf{x} \mid \phi(\mathbf{x}) < +\infty\}$ as an implicit constraint of problem (1.2). When $\mathbf{g}$ is linear, ϕ is a piecewise linear convex function. In this case, problem (1.2) may be solved, for instance, by the algorithm recently proposed by the authors [16], which utilizes the nonsmooth optimization algorithms developed in [1, 6]. However, if $\mathbf{g}$ is a general nonlinear function, then the algorithm of [16] may not be applied to problem (1.2), because ϕ is usually neither smooth nor convex.

The purpose of this paper is to propose a practical approach to problem (1.1), in which the number of linear variables is presumably much greater than that of nonlinear variables. Using a penalty function technique [10, 13] and a perturbation technique for linear programs [12, 14], we first formulate a parametric approximate problem for (1.1). By reformulating the approximate problem, we then obtain a differentiable nonlinear programming problem containing the nonlinear variables only. The main advantage of this approach consists in that any available nonlinear programming code can be used to solve the last problem. We may thus avoid dealing with the original problem (1.1) directly or developing a specialized algorithm for solving the nondifferentiable problem (1.2).

The proposed approach is particularly useful when problem (1.1) takes the following structured form:

$$\begin{array}{ll} \text{minimize}_{\mathbf{x},\mathbf{y}} & f(\mathbf{x}) + \sum_{k=1}^{K} \mathbf{c}_k^T \mathbf{y}_k \\ \text{subject to} & \mathbf{g}_k(\mathbf{x}) + \mathbf{A}_k^T \mathbf{y}_k \leq \mathbf{b}_k, \quad k = 1, \ldots, K, \\ & \mathbf{h}(\mathbf{x}) \leq \mathbf{0}, \end{array} \tag{1.4}$$

where $\mathbf{y} = (\mathbf{y}_1, ..., \mathbf{y}_K)$, $\mathbf{g}_k : R^n \to R^{\ell_k}$, $\mathbf{A}_k$ are $n_k \times \ell_k$ matrices, $\mathbf{b}_k$ are ℓ_k-dimensional vectors, $\mathbf{c}_k$ are n_k-dimensional vectors, $\ell = \sum_{k=1}^{K} \ell_k$ and $N = \sum_{k=1}^{K} n_k$. The structure of this problem is illustrated in Figure 1. (Notice the notational difference between problems (1.1) and (1.4), such as g_i, b_i and $\mathbf{g}_k$, $\mathbf{b}_k$.) Problem (1.4) is amenable to parallel decomposition into smaller problems and this feature can be incorporated in the proposed approach to the full extent.

This paper is organized as follows: In Section 2, we introduce the parametric approximate problem for problem (1.1) and derive a differentiable optimization problem, which is equivalent to the approximate problem but contains the nonlinear variables only. In Section 3, we give error bound results for the approximate problem under appropriate assumptions. In Section 4, we focus on the structured problem (1.4) and discuss parallel decomposition into smaller problems. Computational results are reported in Section 5. Finally, we conclude this paper in Section 6.

2. Formulation and Decomposition

In this section, we consider the following problem:

$$\begin{array}{ll} \text{minimize}_{\mathbf{x},\mathbf{y},\mathbf{z}} & f(\mathbf{x}) + \mathbf{c}^T \mathbf{y} + \frac{\varepsilon}{2} \| \mathbf{y} \|^2 + \frac{M}{2} \| \mathbf{z} \|^2 \\ \text{subject to} & \mathbf{g}(\mathbf{x}) + \mathbf{A}^T \mathbf{y} - \mathbf{z} \leq \mathbf{b}, \\ & \mathbf{h}(\mathbf{x}) \leq \mathbf{0}, \end{array} \tag{2.1}$$

where ε and M are positive constants, $\mathbf{z} = (z_1, \ldots, z_\ell)^T$ and $\| \cdot \|$ denotes the Euclidean norm of a vector. The objective function of problem (2.1) is strictly convex in $\mathbf{y}$, though parameter ε is supposed to be sufficiently small. A similar perturbation technique has

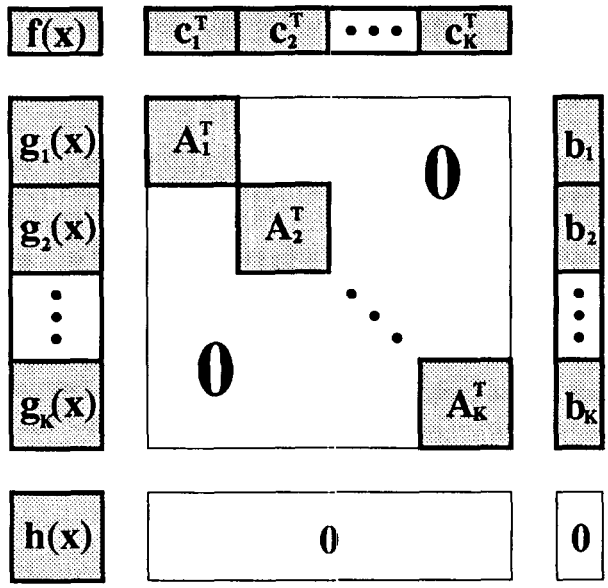

Figure 1: The structure of problem (1.4).

parameter ε is supposed to be sufficiently small. A similar perturbation technique has been used by Mangasarian [12] in the context of linear programming. On the other hand, parameter M is supposed to be large enough, so that the quadratic term of $\mathbf{z}$ in the objective function of problem (2.1) plays the role of a penalty function associated with the constraints $\mathbf{g}(\mathbf{x}) + \mathbf{A}^T\mathbf{y} \leq \mathbf{b}$ in problem (1.1). Recently Lazimy [10] has proposed a similar technique based on a nondifferentiable exact penalty function method. Here we employ a quadratic penalty method in order that the objective function of (2.1) is strictly convex in $\mathbf{z}$. In fact, the strict convexity of the quadratic terms in (2.1) is essential in the subsequent discussion. A combined perturbation penalty technique is also used in [5] to find a solution of the system of linear inequalities

$$\mathbf{A}^T\mathbf{y} \leq \mathbf{b},$$

which may be considered a very special instance of problem (1.1), *i.e.*, $f(\mathbf{x}) \equiv 0$, $\mathbf{g}(\mathbf{x}) \equiv \mathbf{0}$, $\mathbf{h}(\mathbf{x}) \equiv \mathbf{0}$ and $\mathbf{c} = \mathbf{0}$.

Intuitively it is expected that an optimal solution of problem (2.1) approaches an optimal solution of problem (1.1) as $\varepsilon \to 0$ and $M \to +\infty$. In fact, this holds true under suitable assumptions. Rigorous analysis will be given in the next section.

Now, by temporarily fixing $\mathbf{x}$ in problem (2.1), we obtain the following quadratic programming problem in $(\mathbf{y}, \mathbf{z})$:

$$\begin{array}{ll} \text{minimize}_{\mathbf{y},\mathbf{z}} & \mathbf{c}^T\mathbf{y} + \dfrac{\varepsilon}{2} \| \mathbf{y} \|^2 + \dfrac{M}{2} \| \mathbf{z} \|^2 \\ \text{subject to} & \mathbf{A}^T\mathbf{y} - \mathbf{z} \leq \mathbf{b} - \mathbf{g}(\mathbf{x}). \end{array} \tag{2.2}$$

Unlike the problem on the right-hand side of (1.3), problem (2.2) is feasible for any fixed $\mathbf{x}$. Moreover, since the objective function is strictly convex, problem (2.2) has a unique optimal

solution for any $\mathbf{x}$. Therefore if we define the function $\hat{\phi}$ by

$$\hat{\phi}(\mathbf{x};\varepsilon,M) = \min_{\mathbf{y},\mathbf{z}} \left\{ \mathbf{c}^T\mathbf{y} + \frac{\varepsilon}{2} \| \mathbf{y} \|^2 + \frac{M}{2} \| \mathbf{z} \|^2 \,\middle|\, \mathbf{A}^T\mathbf{y} - \mathbf{z} \leq \mathbf{b} - \mathbf{g}(\mathbf{x}) \right\},$$

then $\hat{\phi}(\mathbf{x};\varepsilon,M)$ is finite-valued for all $\mathbf{x}$. By a duality theorem in quadratic programming [11], $\hat{\phi}(\mathbf{x};\varepsilon,M)$ is equal to the optimal value of the following dual problem of (2.2):

$$\begin{array}{ll} \text{maximize}_{\mathbf{u}} & -\frac{1}{2}\mathbf{u}^T \left(\frac{1}{\varepsilon}\mathbf{A}^T\mathbf{A} + \frac{1}{M}\mathbf{I} \right) \mathbf{u} + \left(\mathbf{g}(\mathbf{x}) - \mathbf{b} - \frac{1}{\varepsilon}\mathbf{A}^T\mathbf{c} \right)^T \mathbf{u} - \frac{1}{2\varepsilon} \| \mathbf{c} \|^2 \\ \text{subject to} & \mathbf{u} \geq \mathbf{0}, \end{array} \tag{2.3}$$

where $\mathbf{u} = (u_1, \ldots, u_\ell)^T$, and $\mathbf{I}$ denotes the ℓ-dimensional identity matrix. Since this problem also attains its maximum uniquely for any $\mathbf{x}$, a well-known result concerning the differentiability of max functions (see, e.g., [9, Section 8.5]) ensures that the function $\hat{\phi}(\cdot\,;\varepsilon,M)$ is everywhere differentiable and its gradient at $\mathbf{x}$ is given by

$$\nabla\hat{\phi}(\mathbf{x};\varepsilon,M) = \nabla\mathbf{g}(\mathbf{x})\hat{\mathbf{u}}(\mathbf{x};\varepsilon,M), \tag{2.4}$$

where $\hat{\mathbf{u}}(\mathbf{x};\varepsilon,M)$ is the unique optimal solution of (2.3).

As a result, we may rewrite problem (2.1) as follows:

$$\begin{array}{ll} \textit{minimize}_{\mathbf{x}} & f(\mathbf{x}) + \hat{\phi}(\mathbf{x};\varepsilon,M) \\ \text{subject to} & \mathbf{h}(\mathbf{x}) \leq \mathbf{0}. \end{array} \tag{2.5}$$

It follows from the above arguments that problem (2.5) is a differentiable nonlinear programming problem containing the variables $\mathbf{x}$ only. We may therefore apply any conventional nonlinear programming method to solve (2.5), where the values of $\hat{\phi}(\mathbf{x};\varepsilon,M)$ and $\nabla\hat{\phi}(\mathbf{x};\varepsilon,M)$ can be computed by solving the quadratic programming problem (2.2) or its dual (2.3).

3. Error Bounds

In this section, we consider how the solution of problem (2.1) behaves as $\varepsilon \to 0$ and $M \to +\infty$. Sensitivity analysis in nonlinear programming [3] plays an important role here.

We assume that problem (1.1) has an optimal solution $(\bar{\mathbf{x}},\bar{\mathbf{y}})$ satisfying the following three conditions.

(a) Second-order sufficient conditions for optimality:

There exist vectors $\bar{\mathbf{u}} = (\bar{u}_1, \ldots, \bar{u}_\ell)^T$ and $\bar{\mathbf{v}} = (\bar{v}_1, \ldots, \bar{v}_m)^T$ satisfying

$$\left\{ \begin{array}{l} \nabla f(\bar{\mathbf{x}}) + \sum_{i=1}^{\ell} \bar{u}_i \nabla g_i(\bar{\mathbf{x}}) + \sum_{j=1}^{m} \bar{v}_j \nabla h_j(\bar{\mathbf{x}}) = 0, \\ \mathbf{c} + \sum_{i=1}^{\ell} \bar{u}_i \mathbf{a}_i = 0, \\ \bar{u}_i \left(g_i(\bar{\mathbf{x}}) + \mathbf{a}_i^T\bar{\mathbf{y}} - b_i \right) = 0, \\ \quad \bar{u}_i \geq 0, \quad g_i(\bar{\mathbf{x}}) + \mathbf{a}_i^T\bar{\mathbf{y}} \leq b_i, \quad i = 1, \ldots, \ell, \\ \bar{v}_j h_j(\bar{\mathbf{x}}) = 0, \\ \quad \bar{v}_j \geq 0, \quad h_j(\bar{\mathbf{x}}) \leq 0, \qquad\qquad j = 1, \ldots, m, \end{array} \right. \tag{3.1}$$

and

$$\mathbf{p}^T\mathbf{L}(\bar{\mathbf{x}},\bar{\mathbf{u}},\bar{\mathbf{v}})\mathbf{p} > 0, \quad {}^{\forall}\begin{pmatrix} \mathbf{p} \\ \mathbf{q} \end{pmatrix} \in C(\bar{\mathbf{x}}), \begin{pmatrix} \mathbf{p} \\ \mathbf{q} \end{pmatrix} \neq \begin{pmatrix} \mathbf{0} \\ \mathbf{0} \end{pmatrix}, \tag{3.2}$$

where

$$\mathbf{L}(\bar{\mathbf{x}},\bar{\mathbf{u}},\bar{\mathbf{v}}) = \nabla^2 f(\bar{\mathbf{x}}) + \sum_{i=1}^{\ell} \bar{u}_i \nabla^2 g_i(\bar{\mathbf{x}}) + \sum_{j=1}^{m} \bar{v}_j \nabla^2 h_j(\bar{\mathbf{x}}),$$

$$C(\bar{\mathbf{x}}) = \left\{ \begin{pmatrix} \mathbf{p} \\ \mathbf{q} \end{pmatrix} \in R^{n+N} \;\middle|\; \begin{array}{ll} \nabla g_i(\bar{\mathbf{x}})^T \mathbf{p} + \mathbf{a}_i^T \mathbf{q} = 0, & i \in \tilde{I}, \\ \nabla g_i(\bar{\mathbf{x}})^T \mathbf{p} + \mathbf{a}_i^T \mathbf{q} \le 0, & i \in I - \tilde{I}, \\ \nabla h_j(\bar{\mathbf{x}})^T \mathbf{p} = 0, & j \in \tilde{J}, \\ \nabla h_j(\bar{\mathbf{x}})^T \mathbf{p} \le 0, & j \in J - \tilde{J}, \end{array} \right\},$$

and

$$I = \{i \mid g_i(\bar{\mathbf{x}}) + \mathbf{a}_i^T \bar{\mathbf{y}} = b_i,\ i = 1,\dots,\ell\}, \quad \tilde{I} = \{i \mid \bar{u}_i > 0,\ i = 1,\dots,\ell\},$$
$$J = \{j \mid h_j(\bar{\mathbf{x}}) = 0,\ j = 1,\dots,m\}, \qquad \tilde{J} = \{j \mid \bar{v}_j > 0,\ j = 1,\dots,m\}.$$

(Condition (3.2) follows from the fact that problem (1.1) is linear in $\mathbf{y}$. In particular, (3.2) implies that, if $(\mathbf{p}^T, \mathbf{q}^T)^T \in C(\bar{\mathbf{x}})$ and $\mathbf{p} = \mathbf{0}$, then $\mathbf{q} = \mathbf{0}$.)

(b) Linearly independent constraint qualification:

The vectors $\begin{pmatrix} \nabla g_i(\bar{\mathbf{x}}) \\ \mathbf{a}_i \end{pmatrix}$, $i \in I$, and $\begin{pmatrix} \nabla h_j(\bar{\mathbf{x}}) \\ \mathbf{0} \end{pmatrix}$, $j \in J$, are linearly independent.

(c) Strictly complementarity:

$I = \tilde{I}$ and $J = \tilde{J}$, that is,

$$\begin{array}{ll} g_i(\bar{\mathbf{x}}) + \mathbf{a}_i^T \bar{\mathbf{y}} = b_i & \Rightarrow \quad \bar{u}_i > 0, \quad i = 1,\dots,\ell, \\ h_j(\bar{\mathbf{x}}) = 0 & \Rightarrow \quad \bar{v}_j > 0, \quad j = 1,\dots,m. \end{array}$$

Under the above assumptions, the Kuhn-Tucker solution $(\bar{\mathbf{x}}, \bar{\mathbf{y}}, \bar{\mathbf{u}}, \bar{\mathbf{v}})$ is locally unique.

To analyze the behavior of the solution of problem (2.1), it is convenient to set $\delta = 1/M$. Then Kuhn-Tucker conditions for problem (2.1) may be rewritten as

$$\left\{ \begin{array}{ll} \nabla f(\mathbf{x}) + \sum_{i=1}^{\ell} u_i \nabla g_i(\mathbf{x}) + \sum_{j=1}^{m} v_j \nabla h_j(\mathbf{x}) = 0, & \\ \mathbf{c} + \varepsilon \mathbf{y} + \sum_{i=1}^{\ell} u_i \mathbf{a}_i = 0, & \\ z_i - \delta u_i = 0, & i = 1,\dots,\ell, \\ u_i \left(g_i(\mathbf{x}) + \mathbf{a}_i^T \mathbf{y} - z_i - b_i \right) = 0, & \\ u_i \ge 0, \quad g_i(\mathbf{x}) + \mathbf{a}_i^T \mathbf{y} - z_i \le b_i, & i = 1,\dots,\ell, \\ v_j h_j(\mathbf{x}) = 0, & \\ v_j \ge 0, \quad h_j(\mathbf{x}) \le 0, & j = 1,\dots,m. \end{array} \right. \tag{3.3}$$

Since $\delta \to 0$ when $M \to +\infty$, we are now interested in the behavior of the solution of (3.3) as parameters ε and δ both approach zero. By direct comparison between (3.1) and (3.3), we immediately obtain the following lemma.

Lemma 3.1 *Let* $(\bar{\mathbf{x}}, \bar{\mathbf{y}}, \bar{\mathbf{u}}, \bar{\mathbf{v}})$ *satisfy* (3.1). *Then* $(\mathbf{x}, \mathbf{y}, \mathbf{z}, \mathbf{u}, \mathbf{v}) = (\bar{\mathbf{x}}, \bar{\mathbf{y}}, \mathbf{0}, \bar{\mathbf{u}}, \bar{\mathbf{v}})$ *satisfies* (3.3) *when* $(\varepsilon, \delta) = (0, 0)$.

Proof. Obvious. □

For notational convenience, let $\lambda = (\varepsilon, \delta)^T$, $\mathbf{X} = (\mathbf{x}^T, \mathbf{y}^T, \mathbf{z}^T, \mathbf{u}^T, \mathbf{v}^T)^T$ and $\bar{\mathbf{X}} = (\bar{\mathbf{x}}^T, \bar{\mathbf{y}}^T, \mathbf{0}^T, \bar{\mathbf{u}}^T, \bar{\mathbf{v}}^T)^T$, and define the mapping $\mathbf{\Phi}$: $R^{n+N+2\ell+m+2} \to R^{n+N+2\ell+m}$ by

$$\mathbf{\Phi}(\mathbf{X}, \lambda) = \begin{pmatrix} \nabla f(\mathbf{x}) + \nabla \mathbf{g}(\mathbf{x}) \mathbf{u} + \nabla \mathbf{h}(\mathbf{x}) \mathbf{v} \\ \mathbf{c} + \varepsilon \mathbf{y} + \mathbf{A}\mathbf{u} \\ \mathbf{z} - \delta \mathbf{u} \\ \mathbf{U} \left(\mathbf{g}(\mathbf{x}) + \mathbf{A}^T \mathbf{y} - \mathbf{z} - \mathbf{b} \right) \\ \mathbf{V}\mathbf{h}(\mathbf{x}) \end{pmatrix},$$

where $\mathbf{U}$ and $\mathbf{V}$ are the diagonal matrices such that $\mathbf{U} = \mathrm{diag}(u_i)$ and $\mathbf{V} = \mathrm{diag}(v_j)$, respectively. Then we obtain the following lemma.

Lemma 3.2 *If assumptions* (a), (b) *and* (c) *are satisfied, then the Jacobian matrix* $\nabla_{\mathbf{X}}\mathbf{\Phi}(\bar{\mathbf{X}}, \mathbf{0})$ *is nonsingular.*

Proof. It is sufficient to show that for any $(n + N + 2\ell + m)$-dimensional vector $\mathbf{w}$,

$$\nabla_{\mathbf{X}}\mathbf{\Phi}(\bar{\mathbf{X}}, \mathbf{0})^T\mathbf{w} = \mathbf{0} \implies \mathbf{w} = \mathbf{0}.$$

Let us denote $\mathbf{w} = (\mathbf{p}^T, \mathbf{q}^T, \mathbf{r}^T, \mathbf{s}^T, \mathbf{t}^T)^T$, where $\mathbf{p} = (p_1, \cdots, p_n)^T$, $\mathbf{q} = (q_1, \cdots, q_N)^T$, $\mathbf{r} = (r_1, \cdots, r_\ell)^T$, $\mathbf{s} = (s_1, \cdots, s_\ell)^T$ and $\mathbf{t} = (t_1, \cdots, t_m)^T$. Then $\nabla_{\mathbf{X}}\mathbf{\Phi}(\bar{\mathbf{X}}, \mathbf{0})^T\mathbf{w} = \mathbf{0}$ yields the following system of equations:

$$\mathbf{L}(\bar{\mathbf{x}}, \bar{\mathbf{u}}, \bar{\mathbf{v}})\mathbf{p} + \nabla\mathbf{g}(\bar{\mathbf{x}})\mathbf{s} + \nabla\mathbf{h}(\bar{\mathbf{x}})\mathbf{t} = \mathbf{0}, \tag{3.4}$$

$$\mathbf{A}\mathbf{s} = \mathbf{0}, \tag{3.5}$$

$$\mathbf{r} = \mathbf{0}, \tag{3.6}$$

$$\bar{\mathbf{U}}\nabla\mathbf{g}(\bar{\mathbf{x}})^T\mathbf{p} + \bar{\mathbf{U}}\mathbf{A}\bar{\mathbf{y}} - \bar{\mathbf{U}}\mathbf{r} + \mathbf{S}(\mathbf{g}(\bar{\mathbf{x}}) + \mathbf{A}^T\bar{\mathbf{y}} - \mathbf{b}) = \mathbf{0}, \tag{3.7}$$

$$\bar{\mathbf{V}}\nabla\mathbf{h}(\bar{\mathbf{x}})^T\mathbf{p} + \mathbf{T}\mathbf{h}(\bar{\mathbf{x}}) = \mathbf{0}, \tag{3.8}$$

where $\bar{\mathbf{U}} = \mathrm{diag}(\bar{u}_i)$, $\bar{\mathbf{V}} = \mathrm{diag}(\bar{v}_j)$, $\mathbf{S} = \mathrm{diag}(s_i)$ and $\mathbf{T} = \mathrm{diag}(t_j)$.

Since the strict complementarity condition (c) is satisfied, it follows from (3.6), (3.7) and (3.8) that

$$s_i = 0, \quad i \notin I, \tag{3.9}$$

$$\nabla g_i(\bar{\mathbf{x}})^T\mathbf{p} + \mathbf{a}_i^T\mathbf{q} = 0, \quad i \in I, \tag{3.10}$$

$$t_j = 0, \quad j \notin J, \tag{3.11}$$

$$\nabla h_j(\bar{\mathbf{x}})^T\mathbf{p} = 0, \quad j \in J. \tag{3.12}$$

By taking the inner products of $\mathbf{p}$ and the left-hand side of (3.4) and of $\mathbf{q}$ and the left-hand side of (3.5), and then adding them, we obtain

$$\begin{aligned} 0 &= \mathbf{p}^T\mathbf{L}(\bar{\mathbf{x}}, \bar{\mathbf{u}}, \bar{\mathbf{v}})\mathbf{p} + \mathbf{p}^T\nabla\mathbf{g}(\bar{\mathbf{x}})\mathbf{s} + \mathbf{p}^T\nabla\mathbf{h}(\bar{\mathbf{x}})\mathbf{t} + \mathbf{q}^T\mathbf{A}\mathbf{s} \\ &= \mathbf{p}^T\mathbf{L}(\bar{\mathbf{x}}, \bar{\mathbf{u}}, \bar{\mathbf{v}})\mathbf{p} + \mathbf{s}^T(\nabla\mathbf{g}(\bar{\mathbf{x}})^T\mathbf{p} + \mathbf{A}^T\mathbf{q}) + \mathbf{t}^T\nabla\mathbf{h}(\mathbf{x})^T\mathbf{p} \\ &= \mathbf{p}^T\mathbf{L}(\bar{\mathbf{x}}, \bar{\mathbf{u}}, \bar{\mathbf{v}})\mathbf{p}, \end{aligned} \tag{3.13}$$

where the last equality follows from (3.9)-(3.12). Since (3.9)-(3.12) also imply

$$\begin{pmatrix} \mathbf{p} \\ \mathbf{q} \end{pmatrix} \in C(\bar{\mathbf{x}}),$$

it follows from (3.13) and assumption (a) that $\mathbf{p} = \mathbf{0}$ and $\mathbf{q} = \mathbf{0}$.

Finally substituting $\mathbf{p} = \mathbf{0}$, $\mathbf{q} = \mathbf{0}$, $s_i = 0$, $i \notin I$, and $t_j = 0$, $j \notin J$, into (3.4) and (3.5) yields

$$\sum_{i \in I} s_i \begin{pmatrix} \nabla g_i(\bar{\mathbf{x}}) \\ \mathbf{a}_i \end{pmatrix} + \sum_{j \in J} t_j \begin{pmatrix} \nabla h_j(\bar{\mathbf{x}}) \\ \mathbf{0} \end{pmatrix} = \begin{pmatrix} \mathbf{0} \\ \mathbf{0} \end{pmatrix},$$

which along with assumption (b) implies $s_i = 0$, $i \in I$, and $t_j = 0$, $j \in J$.

Consequently, $\nabla_{\mathbf{X}}\mathbf{\Phi}(\bar{\mathbf{X}}, \mathbf{0})^T\mathbf{w} = \mathbf{0}$ implies $\mathbf{w} = \mathbf{0}$, i.e., the Jacobian matrix $\nabla_{\mathbf{X}}\mathbf{\Phi}(\bar{\mathbf{X}}, \mathbf{0})$ is nonsingular. □

The next theorem is a direct consequence of the above lemmas.

Theorem 3.1 *Let assumptions* (a), (b) *and* (c) *be satisfied. Then there exists a continuously differentiable function* $\mathbf{X}(\cdot)$ *on a neighborhood* Ω *of* $\lambda = \mathbf{0}$ *such that* $\mathbf{X}(\mathbf{0}) = \bar{\mathbf{X}}$. *Moreover, for each* $\lambda \in \Omega$, $\mathbf{X}(\lambda)$ *satisfies the second-order sufficient optimality conditions, linearly independent constraint qualification and strictly complementarity for problem* (2.1).

Proof. From Lemma 3.1, $\mathbf{\Phi}(\bar{\mathbf{X}}, \mathbf{0}) = \mathbf{0}$. Since $\nabla_{\mathbf{X}}\mathbf{\Phi}(\bar{\mathbf{X}}, \mathbf{0})$ is nonsingular by Lemma 3.2, it follows from the implicit function theorem applied to the equation $\mathbf{\Phi}(\mathbf{X}, \lambda) = \mathbf{0}$ that there exists a continuously differentiable function $\mathbf{X}(\cdot)$ on a neighborhood Ω of $\lambda = \mathbf{0}$ such that $\mathbf{\Phi}(\mathbf{X}(\lambda), \lambda) = \mathbf{0}$ and $\mathbf{X}(\mathbf{0}) = \bar{\mathbf{X}}$. Moreover, since $\mathbf{X}(\cdot)$ is continuous and $I = \tilde{I}$, $J = \tilde{J}$, we can prove the last half of the theorem by supposing the neighborhood Ω to be sufficiently small. □

Finally, an error analysis for problem (2.1) is given by the following theorem.

Theorem 3.2 *Let assumptions* (a), (b) *and* (c) *be satisfied. Then, for any* $\kappa > 0$, *there exists a neighborhood* Ω *of* $\lambda = \mathbf{0}$ *such that*

$$\| \mathbf{X}(\lambda) - \bar{\mathbf{X}} \| \leq \left(\left\| [\nabla_{\mathbf{X}}\mathbf{\Phi}(\bar{\mathbf{X}}, \mathbf{0})^T]^{-1} \nabla_{\lambda}\mathbf{\Phi}(\bar{\mathbf{X}}, \mathbf{0})^T \right\| + \kappa \right) \| \lambda \|, \quad \forall \lambda \in \Omega. \tag{3.14}$$

Proof. Differentiating the both sides of $\mathbf{\Phi}(\mathbf{X}(\lambda), \lambda) = \mathbf{0}$ with respect to λ and substituting $\lambda = \mathbf{0}$, we obtain

$$\nabla_{\mathbf{X}}\mathbf{\Phi}(\bar{\mathbf{X}}, \mathbf{0})^T \frac{\partial \mathbf{X}(\mathbf{0})}{\partial \lambda} + \nabla_{\lambda}\mathbf{\Phi}(\bar{\mathbf{X}}, \mathbf{0})^T = \mathbf{0}.$$

Since $\nabla_{\mathbf{X}}\mathbf{\Phi}(\bar{\mathbf{X}}, \mathbf{0})$ is nonsingular from Lemma 3.2, we have

$$\frac{\partial \mathbf{X}(\mathbf{0})}{\partial \lambda} = -[\nabla_{\mathbf{X}}\mathbf{\Phi}(\bar{\mathbf{X}}, \mathbf{0})^T]^{-1} \nabla_{\lambda}\mathbf{\Phi}(\bar{\mathbf{X}}, \mathbf{0})^T.$$

But since $\mathbf{X}(\mathbf{0}) = \bar{\mathbf{X}}$ by Theorem 3.1 and since

$$\mathbf{X}(\lambda) - \mathbf{X}(\mathbf{0}) = \frac{\partial \mathbf{X}(\mathbf{0})}{\partial \lambda} \lambda + o(\| \lambda \|),$$

the inequality (3.14) holds provided that the neighborhood Ω is sufficiently small. □

In practice, it is difficult to estimate the quantity $\| [\nabla_X \Phi(\bar{\mathbf{X}}, \mathbf{0})^T]^{-1} \nabla_\lambda \mathbf{\Phi}(\bar{\mathbf{X}}, \mathbf{0})^T \|$. Nevertheless, Theorem 3.2 shows that the Kuhn-Tucker solution $\mathbf{X}(\lambda)$ of problem (2.1) differs from that of the original problem (1.1) by at most a constant multiple of $\| \lambda \|$, provided that $\| \lambda \|$ is small enough.

4. Structured Problem

In this section, we consider the structured problem (1.4). When matrix $\mathbf{A}$ in problem (1.1) is block diagonal, we may rewrite problem (1.1) as problem (1.4), where

$$\mathbf{y} = \begin{pmatrix} \mathbf{y}_1 \\ \vdots \\ \mathbf{y}_K \end{pmatrix}, \quad \mathbf{g}(\mathbf{x}) = \begin{pmatrix} \mathbf{g}_1(\mathbf{x}) \\ \vdots \\ \mathbf{g}_K(\mathbf{x}) \end{pmatrix},$$

$$\mathbf{A} = \begin{pmatrix} \mathbf{A}_1 & & \mathbf{0} \\ & \ddots & \\ \mathbf{0} & & \mathbf{A}_K \end{pmatrix}, \quad \mathbf{b} = \begin{pmatrix} \mathbf{b}_1 \\ \vdots \\ \mathbf{b}_K \end{pmatrix}, \quad \mathbf{c} = \begin{pmatrix} \mathbf{c}_1 \\ \vdots \\ \mathbf{c}_K \end{pmatrix},$$

and for each $k = 1, \ldots, K$, $\mathbf{g}_k$: $R^n \to R^{\ell_k}$, $\mathbf{A}_k$ is an $n_k \times \ell_k$ matrix, and $\mathbf{b}_k$ and $\mathbf{c}_k$ are ℓ_k- and n_k-dimensional vectors, respectively. Note that $\ell = \sum_{k=1}^{K} \ell_k$ and $N = \sum_{k=1}^{K} n_k$ (see Figure 1).

As in Section 2, we consider the following approximate problem for (1.4):

$$\begin{array}{ll} \text{minimize}_{\mathbf{x},\mathbf{y},\mathbf{z}} & f(\mathbf{x}) + \sum_{k=1}^{K} \left(\mathbf{c}_k^T \mathbf{y}_k + \frac{\varepsilon}{2} \| \mathbf{y}_k \|^2 + \frac{M}{2} \| \mathbf{z}_k \|^2 \right) \\ \text{subject to} & \mathbf{g}_k(\mathbf{x}) + \mathbf{A}_k^T \mathbf{y}_k - \mathbf{z}_k \leq \mathbf{b}_k, \quad k = 1, \ldots, K, \\ & \mathbf{h}(\mathbf{x}) \leq \mathbf{0}, \end{array} \tag{4.1}$$

where the variables $\mathbf{z}_k$, $k = 1, \ldots, K$, are ℓ_k-dimensional vectors.

By fixing $\mathbf{x}$ in problem (4.1), we can decompose the problem into smaller problems as follows: For $k = 1, \ldots, K$,

$$\begin{array}{ll} \text{minimize}_{\mathbf{y}_k,\mathbf{z}_k} & \mathbf{c}_k^T \mathbf{y}_k + \frac{\varepsilon}{2} \| \mathbf{y}_k \|^2 + \frac{M}{2} \| \mathbf{z}_k \|^2 \\ \text{subject to} & \mathbf{A}_k^T \mathbf{y}_k - \mathbf{z}_k \leq \mathbf{b}_k - \mathbf{g}_k(\mathbf{x}). \end{array} \tag{4.2}$$

Note that problems (4.2) are independent of each other.

For each k, problem (4.2) is feasible and has a unique Kuhn-Tucker solution for any fixed $\mathbf{x}$. As before, we define the functions $\hat{\phi}_k$, $k = 1, \ldots, K$, by

$$\hat{\phi}_k(\mathbf{x}; \varepsilon, M) = \min \left\{ \mathbf{c}_k^T \mathbf{y}_k + \frac{\varepsilon}{2} \| \mathbf{y}_k \|^2 + \frac{M}{2} \| \mathbf{z}_k \|^2 \,\middle|\, \mathbf{A}_k^T \mathbf{y}_k - \mathbf{z}_k \leq \mathbf{b}_k - \mathbf{g}_k(\mathbf{x}) \right\}.$$

Then the functions $\hat{\phi}_k(\cdot; \varepsilon, M)$ are also finite-valued and differentiable at each $\mathbf{x}$. Moreover, we have

$$\nabla \hat{\phi}_k(\mathbf{x}; \varepsilon, M) = \nabla \mathbf{g}_k(\mathbf{x}) \hat{\mathbf{u}}_k(\mathbf{x}; \varepsilon, M),$$

where $\hat{\mathbf{u}}_k(\mathbf{x}; \varepsilon, M)$ is the vector of Lagrange multipliers of (4.2) (cf. (2.4)).

Consequently, we can rewrite problem (1.4) as follows:

$$\begin{array}{ll} \text{minimize}_{\mathbf{x}} & f(\mathbf{x}) + \sum_{k=1}^{K} \hat{\phi}_k(\mathbf{x}; \varepsilon, M) \\ \text{subject to} & \mathbf{h}(\mathbf{x}) \leq \mathbf{0}. \end{array} \tag{4.3}$$

Problem (4.3) is again a differentiable nonlinear programming problem, so that any conventional nonlinear programming method can be used to solve it. Moreover, the values of $\hat{\phi}_k(\mathbf{x}; \varepsilon, M)$ and $\nabla \hat{\phi}_k(\mathbf{x}; \varepsilon, M)$ can be computed by solving the quadratic programming problem (4.2) or its dual for each k. Since problems (4.2) are independent of each other, we may solve them in parallel for any fixed $\mathbf{x}$.

5. Computational Results

In this section, we report some computational results with the proposed decomposition method for several test problems of medium size. The test problems used in our numerical experiments are of the structured form (1.4), in which the nonlinear functions f, $\mathbf{g}_k$ and $\mathbf{h}$ are all convex quadratic functions and the problem data are constructed using the method suggested in [15]. Throughout the experiments, we fix the numbers of nonlinear variables and nonlinear constraints as $n = 10$ and $m = 10$, respectively. We also fix the size of each linear block as $(n_k, \ell_k) = (10, 10)$ for $k = 1, \ldots, K$. To obtain test problems of various sizes,

Table 1
Appraisal of solutions of problems (4.1) as approximate solutions of problem (1.4)

		$\varepsilon = 10^{-2}$	$\varepsilon = 10^{-3}$	$\varepsilon = 10^{-4}$	$\varepsilon = 10^{-5}$
$M = 10^2$	Residual of K-T equations	0.107×10^{-1} (0.310×10^{-2})	0.107×10^{-2} (0.486×10^{-3})	0.106×10^{-3} (0.451×10^{-2})	0.141×10^{-4} (0.425×10^{-2})
	Feasibility	0.936×10^{-2} (0.0)	0.899×10^{-2} (0.0)	0.852×10^{-2} (0.0)	0.836×10^{-2} (0.0)
	Complementarity	0.917×10^{-2} (0.201×10^{-4})	0.879×10^{-2} (0.117×10^{-4})	0.262×10^{-1} (0.135×10^{-4})	0.580×10^{0} (0.155×10^{-4})
	CPU sec (No. of iterations)	2.296 (17)	2.165 (17)	2.114 (16)	2.138 (16)
$M = 10^3$	Residual of K-T equations	0.105×10^{-1} (0.239×10^{-2})	0.105×10^{-2} (0.475×10^{-3})	0.105×10^{-3} (0.262×10^{-3})	0.135×10^{-4} (0.492×10^{-2})
	Feasibility	0.865×10^{-3} (0.0)	0.853×10^{-3} (0.0)	0.844×10^{-3} (0.0)	0.706×10^{-3} (0.0)
	Complementarity	0.808×10^{-3} (0.175×10^{-4})	0.100×10^{-1} (0.919×10^{-5})	0.314×10^{-1} (0.138×10^{-4})	0.575×10^{0} (0.189×10^{-4})
	CPU sec (No. of iterations)	2.294 (17)	2.162 (17)	2.163 (17)	2.115 (16)
$M = 10^4$	Residual of K-T equations	0.105×10^{-1} (0.255×10^{-2})	0.105×10^{-2} (0.445×10^{-3})	0.105×10^{-3} (0.739×10^{-3})	0.135×10^{-4} (0.506×10^{-2})
	Feasibility	0.839×10^{-4} (0.0)	0.839×10^{-4} (0.0)	0.760×10^{-4} (0.0)	0.666×10^{-4} (0.0)
	Complementarity	0.934×10^{-3} (0.212×10^{-4})	0.957×10^{-2} (0.125×10^{-4})	0.305×10^{-1} (0.156×10^{-4})	0.575×10^{0} (0.154×10^{-4})
	CPU sec (No. of iterations)	2.292 (17)	2.161 (17)	2.169 (17)	2.138 (16)
$M = 10^5$	Residual of K-T equations	0.105×10^{-1} (0.258×10^{-2})	0.105×10^{-2} (0.342×10^{-3})	0.105×10^{-3} (0.377×10^{-3})	0.135×10^{-4} (0.487×10^{-2})
	Feasibility	0.726×10^{-5} (0.0)	0.732×10^{-5} (0.0)	0.675×10^{-5} (0.0)	0.699×10^{-5} (0.0)
	Complementarity	0.911×10^{-3} (0.181×10^{-4})	0.952×10^{-2} (0.140×10^{-4})	0.304×10^{-1} (0.149×10^{-4})	0.575×10^{0} (0.128×10^{-4})
	CPU sec (No. of iterations)	2.305 (17)	2.177 (17)	2.181 (17)	2.123 (16)

"Residual of K-T equations", "Feasibility" and "Complementarity" represent $||\nabla_{\mathbf{y}} L(\mathbf{x}, \mathbf{y}, \mathbf{u}, \mathbf{v})||_\infty$, $||(\mathbf{g}(\mathbf{x}) + \mathbf{A}^T\mathbf{y} - \mathbf{b})_+||_\infty$ and $||\mathbf{U}(\mathbf{g}(\mathbf{x}) + \mathbf{A}^T\mathbf{y} - \mathbf{b})||_\infty$, respectively, where $L(\mathbf{x}, \mathbf{y}, \mathbf{u}, \mathbf{v})$ is the Lagrangian for problem (1.4) and $(\mathbf{z})_+$ denotes the vector with components $\max(0, z_i)$. The quantities in the parentheses represent $||\nabla_{\mathbf{x}} L(\mathbf{x}, \mathbf{y}, \mathbf{u}, \mathbf{v})||_\infty$, $||(\mathbf{h}(\mathbf{x}))_+||_\infty$ and $||\mathbf{V}\mathbf{h}(\mathbf{x})||_\infty$. These are all evaluated at the computed solutions of problems (4.1).

we vary the number K of linear blocks. Namely, if $K = 2$, then we have a problem with 30 variables and 30 constraints, while if $K = 10$, then we obtain a problem containing 110 variables and 110 constraints.

We have employed the dual method of Goldfarb and Idnani [7] and the modified successive quadratic programming method of Fukushima [4] to solve the quadratic programming problems (4.2) and the nonlinear programming problem (4.3), respectively. In particular, we have used FORTRAN programs called QPDUAL and SQP, which have been coded by Fukushima to implement the above-mentioned methods. Complete listings of these programs may be found in the book [8]. In our numerical experiments, the default values were used for all parameters in the programs and the scaling option was not taken. The runs were made in double precision on a FACOM M780-30 computer at the Data Processing Center, Kyoto University.

In order to verify the results established in Section 3, we first varied the values of parameters ε and M for a test problem with ten linear blocks, *i.e.*, $K = 10$. The results are summarized in Table 1, which shows the closeness of solutions of the parametric problems (4.1) to that of the original problem (1.4) for various values of ε and M. To appraise the solution of problem (4.1) as an approximate solution of (1.4), we evaluate the residual of the Kuhn-Tucker (K-T) equations, feasibility and complementarity conditions for problem (1.4). Specifically, "Residual of the K-T equations", "Feasibility" and "Complementarity" in Table 1 represent $||\nabla_{\mathbf{y}} L(\mathbf{x},\mathbf{y},\mathbf{u},\mathbf{v})||_\infty$, $||(\mathbf{g}(\mathbf{x})+\mathbf{A}^T\mathbf{y}-\mathbf{b})_+||_\infty$ and $||\mathbf{U}(\mathbf{g}(\mathbf{x})+\mathbf{A}^T\mathbf{y}-\mathbf{b})||_\infty$, respectively, where $L(\mathbf{x},\mathbf{y},\mathbf{u},\mathbf{v})$ is the Lagrangian for problem (1.4) and $(\mathbf{z})_+$ denotes the vector with components $\max(0, z_i)$. Moreover, the quantities in the parentheses represent $||\nabla_{\mathbf{x}} L(\mathbf{x},\mathbf{y},\mathbf{u},\mathbf{v})||_\infty$, $||(\mathbf{h}(\mathbf{x}))_+||_\infty$ and $||\mathbf{V}\mathbf{h}(\mathbf{x})||_\infty$. These have all been evaluated at the computed solutions of problems (4.1). The table clearly indicates that the solution of problem (4.1) becomes a better approximate solution of problem (1.4) as $\varepsilon \to 0$ and $M \to +\infty$. In particular, it may be observed that $||\nabla_{\mathbf{y}} L(\mathbf{x},\mathbf{y},\mathbf{u},\mathbf{v})||_\infty$ decreases in proportion to ε, while $||(\mathbf{g}(\mathbf{x}) + \mathbf{A}^T\mathbf{y} - \mathbf{b})_+||_\infty$ is inversely proportional to M, supporting the validity of Theorem 3.2. We should point out, however, that errors in the complementarity conditions $||\mathbf{U}(\mathbf{g}(\mathbf{x}) + \mathbf{A}^T\mathbf{y} - \mathbf{b})||_\infty$ grows as ε gets smaller, though we have been unable to find any rationale.

Table 1 also contains the number of iterations and CPU time for each choice of the parameter pair (ε, M). It is seen that the choice of parameters ε and M does not affect the computational cost, as long as the values of ε and M lie in the specified range. This fact was rather surprising, because we expected that the approximate problem (4.1) would become harder to solve as ε was decreased or M was increased. It may be challenging to find the reason for this seemingly curious phenomenon. At this moment, however, we have no satisfactory answer to it.

Incidentally, we have solved the original problem (1.4) by applying the above-mentioned program SQP directly, *i.e.*, without resort to decomposition. The direct application of SQP required 18 iterations before termination, spending 11.027 seconds.

Next, we examine the behavior of the proposed method under variation of problem size. In this experiment, parameters ε and M are fixed as $\varepsilon = 10^{-3}$ and $M = 10^3$, respectively. To construct test problems of various sizes, the number of linear blocks is varied as $K = 2, 4, 6, 8$ and 10. Three problem instances were generated for each problem size and solved through the proposed decomposition method. For comparison purposes, the same problems were also solved by applying the program SQP directly. The computational results are illustrated in Figure 2, where the CPU time for each of the three test problems is plotted for every K and the medians are linked together by line segments. Observe that the CPU

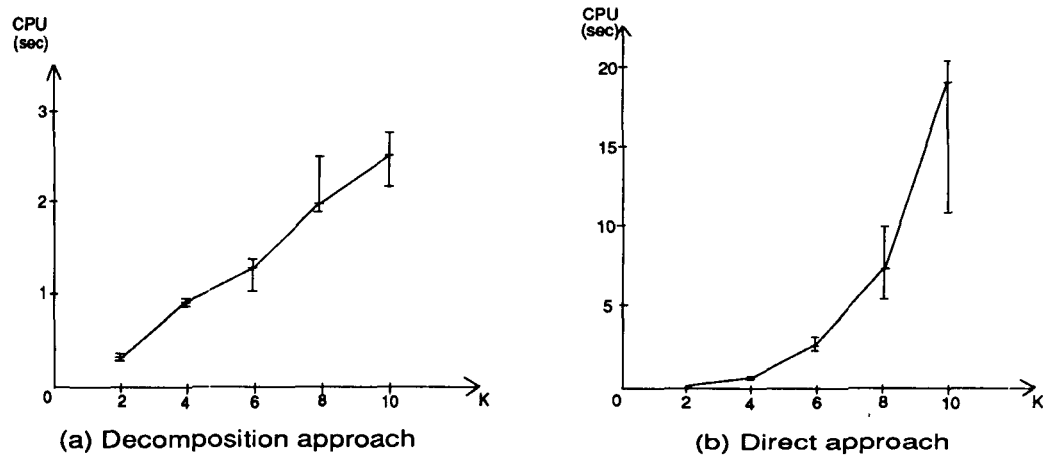

Figure 2: Decomposition approach versus direct approach.

time of the direct application of SQP grows nonlinearly as K increases, while that of the proposed decomposition method grows almost linearly. These phenomena may be explained as follows: In the former approach, the size of a quadratic programming subproblem to be solved at each iteration increases with K, and the computational complexity of the quadratic programming problem is by no means linear in problem size. On the other hand, in the latter approach, the number of subproblems (4.2) to be solved at each iteration is K, but the size of those subproblems remains constant as K increases. Consequently we may expect that the relative efficiency of the proposed decomposition method grows with the increase in problem size.

6. Conclusion

We have proposed a decomposition approach, based on penalty and perturbation techniques, for solving a class of nonlinear programming problems. This approach gives an approximate optimal solution to the given problem, but its closeness to the exact solution may be controlled by means of the penalty and perturbation parameters. The primary advantage of the proposed approach is that it can readily be put into practice if general-purpose nonlinear programming and quadratic programming codes are available. The computational experience reported in the previous section is encouraging enough to claim the practical usefulness of the proposed approach.

References

[1] Auslender, A.: Numerical Methods for Nondifferentiable Convex Optimization. *Mathematical Programming Study*, Vol. 30 (1987), 102-126.

[2] Benders, J. F.: Partitioning Procedures for Solving Mixed-Variables Programming Problems. *Numerische Mathematik*, Vol. 4 (1962), 238-252.

[3] Fiacco, A. V.: *Introduction to Sensitivity and Stability Analysis in Nonlinear Programming*, Academic Press, New York, 1983.

[4] Fukushima, M.: A Successive Quadratic Programming Algorithm with Global and Superlinear Convergence Properties. *Mathematical Programming*, Vol. 35 (1986), 253-264.

[5] Fukushima, M.: A Conjugate Gradient Algorithm for Sparse Linear Inequalities. *Journal of Computational and Applied Mathematics*, Vol. 30 (1990), 329-339.

[6] Fukushima, M.: A Successive Quadratic Programming Method for a Class of Constrained Nonsmooth Optimization Problems. *Mathematical Programming*, Vol. 49 (1991), 203-251.

[7] Goldfarb, D. and Idnani, A.: A Numerically Stable Dual Method for Solving Strictly Convex Quadratic Programs. *Mathematical Programming*, Vol. 27 (1983), 1-33.

[8] Ibaraki, T. and Fukushima, M.: *FORTRAN 77 Programs of Optimization Algorithms* (in Japanese). Iwanami-Shoten, Tokyo, 1991.

[9] Lasdon, L. S.: *Optimization Theory for Large Systems.* Macmillan, New York, 1970.

[10] Lazimy, R.: An Exact Penalty Function and Relaxation Approach for Solving Decomposable Nonlinear Programs. *SIAM Journal on Control and Optimization*, Vol. 26 (1988), 127-137.

[11] Mangasarian, O. L.: *Nonlinear Programming.* McGraw-Hill, New York, 1969.

[12] Mangasarian, O. L.: Normal solutions of linear programs. *Mathematical Programming Study*, Vol. 22 (1984), 206-216.

[13] Mangasarian, O. L.: Sufficiency of Exact Penalty Minimization. *SIAM Journal on Control and Optimization*, Vol. 23 (1985), 30-37.

[14] Mangasarian, O. L. and Meyer, R. R.: Nonlinear Perturbation of Linear Programs. *SIAM Journal on Control and Optimization*, Vol. 17 (1979), 745-752.

[15] Rosen, J. B. and Suzuki, S.: Construction of Nonlinear Programming Test Problems. *Communications of the ACM*, Vol. 8 (1965), 113.

[16] Wakahara, T., Fukushima, M. and Ibaraki, T.: A Successive Quadratic Programming Decomposition Method for Large-Scale Nonlinear Programming Problems (in Japanese). *Transactions of the Institute of Systems, Control and Information Engineers*, Vol. 4 (1991), 473-480.

Masao FUKUSHIMA:
Graduate School of Information Science,
Advanced Institute of Science and Technology, Nara,
Takayama, Ikoma, Nara 630-01, Japan