Comput Optim Appl (2009) 43: 307–328
DOI 10.1007/s10589-007-9147-4

An inexact-restoration method for nonlinear bilevel programming problems

R. Andreani · S.L.C. Castro · J.L. Chela · A. Friedlander · S.A. Santos

Received: 11 December 2006 / Revised: 19 October 2007 / Published online: 20 November 2007

Abstract We present a new algorithm for solving bilevel programming problems without reformulating them as single-level nonlinear programming problems. This strategy allows one to take profit of the structure of the lower level optimization problems without using non-differentiable methods. The algorithm is based on the inexact-restoration technique. Under some assumptions on the problem we prove global convergence to feasible points that satisfy the *approximate gradient projection* (AGP) optimality condition. Computational experiments are presented that encourage the use of this method for general bilevel problems.

Keywords Bilevel programming · Inexact-restoration · Optimization

This work was supported by PRONEX-Optimization (PRONEX—CNPq/FAPERJ E-26/171.164/2003—APQ1), FAPESP (Grants 06/53768-0 and 05-56773-1) and CNPq.

R. Andreani · A. Friedlander (✉) · S.A. Santos
Department of Applied Mathematics, IMECC-UNICAMP, University of Campinas, CP 6065, 13081-970 Campinas, SP, Brazil
e-mail: friedlan@ime.unicamp.br

R. Andreani
e-mail: andreani@ime.unicamp.br

S.A. Santos
e-mail: sandra@ime.unicamp.br

S.L.C. Castro
Faculdades Integradas Metropolitanas de Campinas-METROCAMP, Campinas, SP, Brazil
e-mail: suzana.castro@metrocamp.com.br

J.L. Chela
Banco Itaú and Universidade Presbiteriana Mackenzie, São Paulo, SP, Brazil
e-mail: chela@mackenzie.br

1 Introduction

The bilevel programming problem that we address in this paper can be stated as follows:

$$\begin{aligned} &\underset{x,y}{\text{Minimize}} \quad F(x,y) \\ &\text{s.t.} \quad \begin{cases} H(x)=0, \quad x \in X \\ y = \arg\min_{y} f(x,y) \\ \text{s.t.} \quad \begin{cases} h(x,y)=0 \\ y \in Y. \end{cases} \end{cases} \end{aligned} \tag{1}$$

We assume that $\Omega = X \times Y$, $X \subset \mathbb{R}^n$ and $Y \subset \mathbb{R}^m$ are multidimensional bounded boxes, $F : \mathbb{R}^{n+m} \to \mathbb{R}$, $H : \mathbb{R}^n \to \mathbb{R}^q$, $f(x,y) : \mathbb{R}^{n+m} \to \mathbb{R}$ and $h : \mathbb{R}^{n+m} \to \mathbb{R}^p$. The essential feature of a bilevel problem is that a subset of its variables is required to solve another optimization problem, parameterized by the remaining variables, called the lower-level problem. Throughout the paper we assume that $\nabla F(x,y)$, $\nabla H(x)$, $\nabla^2 f(x,y)$, $\nabla^2 h(x,y)$ exist and are continuous in Ω.

Bilevel programming problems model two-level hierarchical systems and have been studied since the seventies [4, 5]. In the last years some surveys and bibliographic reviews appeared [9, 12, 13, 31], where history, applications, algorithms, theoretical questions and almost all relevant references can be found. Books on bilevel programming and related issues are [2, 11, 20, 29].

We subdivide the algorithms for solving bilevel programming problems into three classes, following Dempe [12]: algorithms that solve them globally, methods that find stationary points or points that satisfy some local optimality condition, and heuristics. Also inside these classes there are problems with particular structures for which special algorithms have been designed. In [12] extensive bibliography is given that covers most of the significant work done in the last years.

The reformulation of problem (1) as a single-level problem given next is frequently used to solve the original bilevel problem and to obtain optimality conditions for it:

$$\begin{aligned} &\underset{x,y,\mu,\gamma,\alpha}{\text{Minimize}} \quad F(x,y) \\ &\text{s.t.} \quad H(x)=0, \quad x \in X \\ &\qquad \nabla_y f(x,y) + \nabla_y h(x,y)\mu - \gamma + \alpha = 0 \\ &\qquad h(x,y) = 0 \\ &\qquad \gamma_i(y_i - \ell_i) = 0, \quad i = 1,\dots,m \\ &\qquad \alpha_i(u_i - y_i) = 0, \quad i = 1,\dots,m \\ &\qquad y \in Y \equiv [\ell, u], \quad \gamma \ge 0, \ \alpha \ge 0. \end{aligned} \tag{2}$$

The use of algorithms for nonlinear programming to solve bilevel problems used to be neglected in the past, because it was thought that the essential non-regularity of the feasible points of their reformulation as single-level problems, via their KKT conditions, was a serious drawback. These single-level problems include complementarity constraints to describe the feasible region. In [1] this issue is extensively discussed and it is shown that methods that converge to points that satisfy an optimality condition (AGP, which stands for Approximate Gradient Projection), introduced in [25], have good chances to converge to the solution of mathematical programming problems with complementarity constraints. In [17] superlinear local convergence is shown when SQP methods are used to solve a class of mathematical programs with equilibrium constraints near a strongly stationary point.

The results related with the reformulation (2), numerical experimentation with mechanical engineering problems, and the discussions in [24] encouraged us to present a new approach for solving two-level problems.

The algorithm we present here belongs to the second class in Dempe's classification and, under suitable conditions, finds points that verify the AGP local optimality condition given in [25]. The method is strongly based in the inexact-restoration technique introduced in [22]. Inexact restoration was designed to solve general nonlinear optimization problems, that is, problems with nonlinear objective function and nonlinear constraints. It is an iterative method that deals separately with feasibility and optimality at each iteration. In the feasibility stage, called restoration phase, it seeks for a more feasible point considering the true function and constraints. In the optimality phase it looks for a point (a trial point) that reduces sufficiently the value of a Lagrangian defined by the original data, in a tangent set that approximates the feasible region, within a trust region centered at the point obtained in the feasibility phase. Sufficient decrease of a merit function that balances feasibility and optimality determines the acceptance of the trial point obtained in the optimization phase. If the trial point is not accepted, the size of the trust region is reduced. The user is free to use any algorithm in each phase, making the choice of problem-oriented solvers possible.

In the problems addressed in the current work, a feasible point is a global solution of an optimization problem. Therefore, a global optimization algorithm should be used, if available, in the restoration phase. If the lower level problems are convex for any $x \in X$, there exist many efficient algorithms to solve them, and even algorithms designed for special instances of them. We require the algorithm used in the restoration phase to be globally convergent in the sense that limit points are stationary.

The restoration phase of the algorithm operates on the original lower level problem, that is, we do not reformulate the bilevel problem as a single-level optimization problem. In the optimization phase we obtain a new point that reduces the value of the true Lagrangian of the first-level objective function on the intersection of a linear approximation of the feasible set with a trust region centered at the point obtained in the restoration phase. The point is accepted as a new iterate if it reduces sufficiently the value of a merit function. It is proved that the algorithm is well defined with mild assumptions on the problem.

In the optimization phase any minimization algorithm for linearly constrained problems can be employed. It is important to note that we use neither non-differentiable nor smoothing techniques.

Our main contribution in this article is the introduction of a well defined algorithm for solving a class of nonlinear bilevel programming problems without reformulating them as single-level problems and without using non-differentiable techniques. The conditions under which we can apply the proposed algorithm are mild. We also present global convergence results under more restrictive conditions. We are aware that strong theoretical convergence results are important and desirable, but we do not think that the lack of them turns the algorithms unusable. For instance, the very popular BFGS method for unconstrained minimization, whose global convergence has been proved just for convex functions, is generally considered one of the most efficient methods for unconstrained optimization. Certainly, it will continue to be used although recently in [10, 26] examples were given showing that it can fail to converge in the general case. We think that it is essential to prove the well-definiteness of a proposed algorithm and then, of course, seek for results concerning its global convergence. Numerical experiments might encourage its usage and finally, extensive experimentation with a good implementation will tell if it is robust or not. In this paper we adopt this point of view.

Recently in [8] a trust-region algorithm is presented that also preserves the bilevel structure of the problem, and does not involve the use of non-differentiable techniques. In [7] a collection of test problems for the aforementioned algorithm is given. The kind of problems studied in these articles share the features of the problems we intend to solve here, thus their test problems have been included in our numerical experiments. It would be interesting to compare theoretical properties of this algorithm with ours, but in [8], where the algorithm is introduced, there are no convergence results. It is very difficult to find in the literature algorithms for problems with the generality proposed here, with strong theoretical properties, accompanied by numerical experiments. Even to find a good set of large test problems is not easy and we share the same difficulties mentioned in the last section of [8].

Our method was tested with many bilevel problems with good results. We wanted to check if our algorithm was able to find the global solutions of the problems of the collection [7] and other problems found in the literature. The efficiency will strongly depend on the particular algorithms chosen for the different phases, and this choice should consider the problem's structure if possible. Therefore, a comparison with the algorithm proposed in [8] in terms of efficiency is not made. Shape optimization and truss topology design problems in mechanical engineering motivated us to study and try to develop an algorithm for bilevel programming. We include some comments on experiments with this application. See [6].

In order to simplify as much as possible the notation of this paper we will define the algorithm and present the theoretical analysis considering the following bilevel problem:

$$\begin{aligned} \underset{x,y}{\text{Minimize}} \quad & F(x,y) \\ & x \in X \\ \text{s.t.} \quad & \begin{cases} y = \arg\min_{y} f(x,y) \\ \text{s.t.} \quad \begin{cases} h(x,y) = 0 \\ y \geq 0. \end{cases} \end{cases} \end{aligned} \tag{3}$$

It is easy to extend the analysis using the same arguments if we include the upper level constraints $H(x) = 0$ and the bounded box Y, together with the corresponding complementarities and the additional vector of multipliers associated with the upper bounds of y that arise in this case.

By this we mean that the algebraic manipulations in the proofs of the results given in the next sections can be extended straightforward when $y \in Y$ replaces $y \geq 0$. We take the liberty of using compactness of the y domain as assumed in the original formulation (1) of the problem.

This paper is organized as follows. In Sect. 2 we comment on the inexact-restoration method for standard nonlinear programming. In Sect. 3 we present the algorithm for bilevel programs and we prove that it is well defined under mild assumptions. In Sect. 4 we prove results concerning the feasibility of limit points of sequences generated by the algorithm. We also give conditions that enable us to prove global convergence to points satisfying the AGP optimality condition. Moreover, we discuss the hypotheses on the problem that imply the conditions given before. In Sect. 5 we present numerical experiments and analyze some possible outcomes when the hypotheses given in the previous section fail to be true. We comment on conclusions and future research in Sect. 6.

2 Inexact-restoration methods

Inexact-restoration methods are motivated by the bad behavior of feasible methods in the presence of strong nonlinearities. To overcome this drawback, in [21–23] the authors introduce algorithms that keep feasibility under control but are tolerant when the iterates are far from the solution. An interesting discussion about the background, the main features of these methods and their analogies and differences with sequential quadratic programming is presented in [24].

We describe now the main features of the inexact-restoration method that motivated our algorithm for bilevel problems. This method was proposed to solve problems of the form

$$\begin{aligned} &\underset{x\in\Omega}{\text{Minimize}} \quad f(x) \\ &\text{s.t.} \quad C(x)=0 \end{aligned} \tag{4}$$

where $f:\mathbb{R}^n\to\mathbb{R}$ and $C:\mathbb{R}^n\to\mathbb{R}^p$ are continuously differentiable and $\Omega\subset\mathbb{R}^n$ is a closed and convex set. It is an iterative method that generates feasible iterates with respect to Ω ($x^k\in\Omega$ for all k). Each iteration consists of two phases, called restoration and minimization phases, respectively. Through the whole paper we shall denote $\|.\|$ an arbitrary vector norm and $\|.\|_\infty$ the supremum norm. For simplicity $|.|$ will denote the Euclidean norm, although in many cases it can be replaced by an arbitrary norm. In the restoration phase, if the current point is not feasible with respect to $C(x)=0$, we find an intermediate point $y^k\in\Omega$ such that, for given $\beta>0$ and $0\le r<1$, $\|y^k-x^k\|\le\beta|C(x^k)|$ and the infeasibility at y^k is a fraction of the infeasibility at x^k ($|C(y^k)|\le r|C(x^k)|$). The values r and β are parameters of the algorithm and remain the same for all iterations. If the current point is feasible with respect to $C(x)=0$, we define $y^k=x^k$.

After the restoration phase we define a linear approximation of the feasible region of (4), containing the point y^k. This approximation is given by

$$\pi(y^k)=\{z\in\Omega \mid C'(y^k)(z-y^k)=0\}. \tag{5}$$

Next, a direction called *Approximate Gradient Projection* (AGP) and denoted by $d_{tan}(y^k)$ is computed as,

$$d_{tan}(y^k)=P_k[y^k-\eta\nabla_x L(y^k,\lambda^k)]-y^k,$$

where $L(x, \lambda) = f(x) + C(x)^T \lambda$, $P_k[w]$ is the orthogonal projection of w on $\pi(y^k)$ and $\eta > 0$ is an arbitrary scaling parameter, independent of k. It turns out that $d_{tan}(y^k)$ is a feasible descent direction for L on $\pi(y^k)$ and its norm is used to test the AGP optimality condition introduced in [25]. A feasible point w satisfies the AGP optimality condition if there exists a sequence $\{w^k\}$ that converges to w such that $d_{tan}(w^k) \to 0$. Finally, before proceeding to the minimization step, a trust region is defined by

$$\mathbb{B}_{k,i} = \{x \in \mathbb{R}^n \mid \|x - y^k\| \le \delta_{k,i}\} \tag{6}$$

and $\delta_{k,i} > 0$.

The goal of the minimization phase is to obtain a trial point $z^{k,i} \in \mathbb{B}_{k,i} \bigcap \pi(y^k)$ that sufficiently reduces $L(.\,, \lambda^k)$. The first trial point at each iteration is obtained using a trust-region radius $\delta_{k,0} \ge \delta_{min}$, where δ_{min} is independent of k. Successive trust-region radius are tried until a point $z^{k,i}$ is found that sufficiently reduces the value of a merit function compared with its value at x^k.

The merit function used is a variant of the sharp Lagrangian introduced in [28], given by

$$\Psi(x, \lambda, \theta) = \theta L(x, \lambda) + (1 - \theta)|C(x)| \tag{7}$$

where $\theta \in (0, 1]$ is a penalty parameter that gives different weights to the objective function and the feasibility. The choice of θ at each iteration depends on practical and theoretical considerations. See [22].

In the next section we present an algorithm to solve bilevel problems based on the inexact-restoration method described above. We will discuss in detail how the central ideas of the inexact-restoration technique are adapted for the solution of two-level problems. The reader interested in the method for ordinary nonlinear programming problems should consult [22].

3 Description of the algorithm

In this section we consider the simplified problem (3). Define

$$C(x, y, \mu, \gamma) = \begin{pmatrix} \nabla_y f(x, y) + \nabla_y h(x, y)\mu - \gamma \\ h(x, y) \\ \gamma_1 y_1 \\ \vdots \\ \gamma_m y_m \end{pmatrix}$$

where $(\mu, \gamma) \in \Delta \subset \mathbb{R}^p \times \mathbb{R}^m$. The Karush-Kuhn-Tucker conditions of the lower level problem in (3), parameterized by x and denoted by $\mathrm{KKT}(x)$, are

$$C(x, y, \mu, \gamma) = 0, \quad y \ge 0, \; \gamma \ge 0. \tag{8}$$

Define

$$L(x, y, \mu, \gamma, \lambda) = F(x, y) + C(x, y, \mu, \gamma)^T \lambda, \tag{9}$$

with $\lambda \in \mathbb{R}^m \times \mathbb{R}^p \times \mathbb{R}^m$. To facilitate the reading we denote $s = (x, y, \mu, \gamma) \in \Omega \times \Delta$, and so (9) turns into

$$L(s, \lambda) = F(x, y) + C(s)^T \lambda.$$

We propose the following algorithm to solve (3). It is an iterative method that generates a sequence $\{s^k\} = \{(x^k, y^k, \mu^k, \gamma^k)\}$. The parameters $\eta > 0$, $M > 0$, $\theta_{-1} \in (0\,1)$, $\delta_{min} > 0$, $\tau_1 > 0$, $\tau_2 > 0$ are given as well as the initial approximation $s^0 \in \Omega \times \Delta$, the initial vector of Lagrange multipliers $\lambda^0 \in \mathbb{R}^{2m+p}$ and a sequence of positive numbers ω^k such that $\sum_{k=0}^{\infty} \omega^k < \infty$.

Assume that for $k \in \{0, 1, 2, \ldots\}$, $s^k \in \Omega \times \Delta$, $\lambda^k \in \mathbb{R}^{2m+p}$ and $\theta_{k-1}, \theta_{k-2}, \ldots, \theta_0$ have been computed. The steps for obtaining $s^{k+1} = (x^{k+1}, y^{k+1}, \mu^{k+1}, \gamma^{k+1})$ and λ^{k+1} are given below.

Algorithm 3.1

Step 1. Initialization of penalty parameter.

Define

$$\theta_k^{min} = \min\{1, \theta_{k-1}, \ldots, \theta_{-1}\}, \tag{10}$$

$$\theta_k^{large} = \min\{1,\ \theta_k^{min} + \omega^k\} \tag{11}$$

and

$$\theta_{k,-1} = \theta_k^{large}.$$

Step 2. Restoration phase.

Find an approximate minimizer $\bar{y} \geq 0$ and a pair of estimated Lagrange multiplier vectors $(\bar{\mu}, \bar{\gamma}) \in \Delta$ of problem

$$\begin{aligned} \underset{y}{\text{Minimize}} \quad & f(x^k, y) \\ \text{s.t.} \quad & h(x^k, y) = 0 \\ & y \geq 0, \end{aligned} \tag{12}$$

and define $z^k = (x^k, \bar{y}, \bar{\mu}, \bar{\gamma})$.

Step 3. Tangent Cauchy direction.

Define $d_{tan}(z^k) \equiv d_{tan}^k$, $\pi(z^k) \equiv \pi_k$.

Compute $d_{tan}^k = P_k[z^k - \eta \nabla L_s(z^k, \lambda^k)] - z^k$, where $P_k[.]$ is the orthogonal projection on π_k, and

$$\pi_k = \{s \in \Omega \times \Delta \mid C'(z^k)(s - z^k) = 0\}. \tag{13}$$

If

$$z^k = s^k \quad \text{and} \quad d_{tan}^k = 0, \tag{14}$$

terminate the execution of the algorithm returning (x^k, y^k) as the final point. Else, set $i \leftarrow 0$, choose $\delta_{k,0} \geq \delta_{min}$ and continue.

Step 4. Minimization phase in π_k.

If $d_{tan}^k = 0$, set $v^{k,0} = z^k$. Else, compute $t_{break}^{k,i} = \min\{1, \delta_{k,i}/\ \|d_{tan}^k\|\}$ and find a point $v^{k,i} \in \pi_k$ such that for some $0 < t \leq t_{break}^{k,i}$,

$$L(v^{k,i}, \lambda^k) \leq \max\{L(z^k + t d_{tan}^k, \lambda^k),\ L(z^k, \lambda^k) - \tau_1 \delta_{k,i},\ L(z^k, \lambda^k) - \tau_2\} \tag{15}$$

with

$$\gamma \geq 0 \tag{16}$$

and

$$\|v^{k,i} - z^k\|_\infty \leq \delta_{k,i}. \tag{17}$$

Step 5. Trial multipliers.

If $d_{tan}^k = 0$ define $\lambda_{trial}^{k,i} = \lambda^k$. Else, Compute $\lambda_{trial}^{k,i} \in \mathbb{R}^{2m+p}$ such that $|\lambda_{trial}^{k,i}| \leq M$.

Step 6. Predicted reduction.

Define, for all $\theta \in [0, 1]$,

$$\begin{aligned} Pred_{k,i}(\theta) \equiv \theta[L(s^k, \lambda^k) - L(v^{k,i}, \lambda^k) - C(z^k)^T(\lambda_{trial}^{k,i} - \lambda^k)] \\ + (1-\theta)[|C(s^k)| - |C(z^k)|]. \end{aligned} \tag{18}$$

Compute $\theta_{k,i}$ as the maximum $\theta \in [0, \theta_{k,i-1}]$ that verifies

$$Pred_{k,i}(\theta) \geq \frac{1}{2}[|C(s^k)| - |C(z^k)|]. \tag{19}$$

Define $Pred_{k,i} = Pred_{k,i}(\theta_{k,i})$.

Step 7. Comparison between actual and predicted reduction.

Compute

$$Ared_{k,i} = \theta_{k,i}[L(s^k, \lambda^k) - L(v^{k,i}, \lambda_{trial}^{k,i})] + (1-\theta_{k,i})[|C(s^k)| - |C(v^{k,i})|].$$

If

$$Ared_{k,i} \geq 0.1 Pred_{k,i} \tag{20}$$

update

$$s^{k+1} = v^{k,i}, \qquad \lambda^{k+1} = \lambda^{k,i}, \qquad \theta_k = \theta_{k,i}, \qquad \delta_k = \delta_{k,i}$$

$$Ared_k = Ared_{k,i}, \qquad Pred_k = Pred_{k,i} \tag{21}$$

and terminate iteration k.

Else, choose $\delta_{k,i+1} \in [0.1\delta_{k,i}, 0.9\delta_{k,i}]$, set $i \leftarrow i+1$ and go to Step 4.

At Step 2 we apply any globally convergent optimization algorithm to solve the second level minimization problem parameterized by x^k. Once an approximate minimizer $\bar{y}$ and a pair of corresponding estimated Lagrange multiplier vectors are obtained, we compute the current set π_k and the direction d_{tan}^k. The set π_k is a linear approximation of the region described by KKT(x^k) containing $z^k = (x^k, \bar{y}, \bar{\mu}, \bar{\gamma})$. If the stopping criterion (14) is not satisfied we define $v^{k,0} = z^k$ and a radius for the trust region $\delta_{k,0} \geq \delta_{min}$. We use $v^{k,0}$ to initialize an algorithm for minimizing nonlinear problems with linear constraints to obtain a sufficient decrease of $L(s, \lambda)$. The trust region is described by simple bounds since we use the supremum norm. We update the multipliers $\lambda^{k,0}$ imposing that $|\lambda^{k,0}| \leq M$. As long as the merit function

$$\Psi(s, \lambda, \theta) = \theta L(s, \lambda) + (1 - \theta)|C(s)| \tag{22}$$

is not sufficiently decreased as required in Step 7, we reduce the trust-region radius and repeat the minimization step obtaining points $v^{k,i} \in \pi_k$. When the desired reduction in the merit function value is obtained, we stop the iteration and update all the data.

3.1 Well-definiteness of the algorithm

The well-definiteness of the inexact restoration algorithm for standard nonlinear programming problems discussed in the previous section is proved in Theorem 4.1 of [22]. In order to obtain a corresponding proof of this theorem for the bilevel programming case the following assumptions are sufficient.

A1. The set $\Omega \times \Delta$ is compact and convex.

A2. There exists $L_1 > 0$ such that, for all (x, y), $(v, w) \in \Omega$,

$$|\nabla F(x, y) - \nabla F(v, w)| \leq L_1|(x, y) - (v, w)|.$$

A3. There exists $L_2 > 0$ such that, for all (x, y), $(v, w) \in \Omega$,

$$|\nabla^2 f(x, y) - \nabla^2 f(v, w)| \leq L_2|(x, y) - (v, w)|,$$
$$|\nabla^2 h(x, y) - \nabla^2 h(v, w)| \leq L_2|(x, y) - (v, w)|.$$

A4. At the current point $C(x^k, y^k, \gamma^k, \mu^k) \neq 0$, the approximate solution and estimated Lagrange multipliers $(\bar{y}, \bar{\gamma}, \bar{\mu})$ obtained at Step 2 of Algorithm 3.1 verify that $\bar{\gamma} \geq 0$ and

$$|C(x^k, \bar{y}, \bar{\mu}, \bar{\gamma})| < |C(x^k, y^k, \mu^k, \gamma^k)|.$$

If $C(x^k, y^k, \mu^k, \gamma^k) = 0$, then $(x^k, \bar{y}, \bar{\mu}, \bar{\gamma}) = (x^k, y^k, \mu^k, \gamma^k)$.

With these assumptions the well-definiteness of Algorithm 3.1 is proved as in Theorem 4.1 in [22]. This result encouraged us to run the algorithm and we observed that the solution was obtained in a large number of test problems. Mild hypotheses on the problem will guarantee A4 and we discuss them in the next section. The algorithm can be applied to a large class of problems without getting stuck at an iteration.

We want to emphasize that in Step 2 of Algorithm 3.1 a minimization procedure should be used if we want to take profit of the bilevel structure. If the lower level problem is not convex, a global optimization algorithm, if available is the best option. As an example, consider a bilevel program such that the lower level problem is to find the smallest eigenvalue and the corresponding eigenvector of a symmetric matrix that depends on several parameters. If this bilevel program is reformulated as a standard single-level problem substituting the lower level problem by the parameterized KKT conditions, the other eigenvalues and corresponding eigenvectors are stationary points as well. The chances of any nonlinear programming algorithm to reach the desired solution are very low. Even SQP methods, that perform efficiently when applied to mathematical programs with complementarity constraints, will probably fail in this case.

4 Convergence results

In this section we discuss the theoretical results obtained for Algorithm 3.1.

4.1 Feasibility

We use a modification of A4 to prove that any limit point of Algorithm 3.1 is a stationary point of some lower level problem. A point (x, y) is feasible for the bilevel problem (3) if y is the global solution of the lower level problem parameterized by x.

Assumptions A1, A2, A3 and the next assumption instead of A4, are sufficient to prove the theorems of this section.

A5. There exists $r \in [0\ 1)$ independently of k, such that the approximate solution and estimated Lagrange multipliers $(\bar{y}, \bar{\gamma}, \bar{\mu})$, obtained at Step 2 of Algorithm 3.1 verify that $\bar{\gamma} \geq 0$ and

$$|C(x^k, \bar{y}, \bar{\mu}, \bar{\gamma})| \leq r|C(x^k, y^k, \mu^k, \gamma^k)|.$$

Moreover, if $C(x^k, y^k, \mu^k, \gamma^k) = 0$, $(x^k, \bar{y}, \bar{\mu}, \bar{\gamma}) = (x^k, y^k, \mu^k, \gamma^k)$.

The proof of the following theorems are obvious adaptations of Theorems 3.4 and 3.5 in [23].

Theorem 4.1 *If Algorithm 3.1 generates an infinite sequence, then*

$$\lim_{k\to\infty} Ared_k = 0.$$

Theorem 4.2 *If Algorithm 3.1 does not stop in a finite number of iterations*

$$\lim_{k\to\infty} |C(s^k)| = 0.$$

This means that any limit point of Algorithm 3.1 is a stationary point of some lower level problem. In other words, it is a feasible point of the reformulation of (3) given by

$$\begin{aligned} \underset{x,y,\mu,\gamma}{\text{Minimize}} \quad & F(x, y) \\ \text{s.t.} \quad & \nabla_y f(x, y) + \nabla_y h(x, y)\mu - \gamma = 0 \\ & h(x, y) = 0 \\ & \gamma_i y_i = 0, \ i = 1, \ldots, m \\ & y \geq 0, \ \gamma \geq 0. \end{aligned} \tag{23}$$

If we find the global minimizer at Step 2 or if the lower level problem is convex for all $x \in X$, any limit point of Algorithm 3.1 is a feasible point of the bilevel problem (3).

4.2 Optimality

We assume now A1, A2, A5 and assumption A6 stated below.

A6. There exists $\beta > 0$, independently of k, such that

$$\|s^k - z^k\| \leq \beta |C(s^k)|. \tag{24}$$

The AGP optimality condition was introduced in [25] for single-level nonlinear programming problems. The set of local minimizers of a nonlinear programming problem is contained in the set of AGP points and this set is strictly contained in the set of Fritz-John points.

The vector $d_{tan}(s)$ can be defined for any $s \in \Omega \times \Delta$ as in Step 3 of Algorithm 3.1. This definition depends just on the problem and we state two optimality conditions for the bilevel programming problem (3) based on the AGP condition:

- Weak AGP.
  A feasible point s^* of problem (23) satisfies the weak AGP condition of problem (3), if there exists a sequence $\{s^k\}$ that converges to s^* and such that $d_{tan}(s^k) \to 0$.
- Strong AGP.
  A point $s^* = (x^*, y^*, \mu^*, \gamma^*)$ such that $C(s^*) = 0$ and (x^*, y^*) is feasible for problem (3) satisfies the strong AGP condition of problem (3), if there exists a sequence $\{s^k\}$ that converges to s^* and such that $d_{tan}(s^k) \to 0$.

The proof of the next theorem is obtained using the same arguments as in the proof of the convergence results for the inexact-restoration method given in [22, 23].

Theorem 4.3 *If assumptions* A1, A2, A3 *and* A5 *are satisfied*, $\{s^k\}$ *is an infinite sequence generated by Algorithm* 3.1 *and* $\{z^k\}$ *is the sequence defined at Step* 2, *then*

1. $|C(s^k)| \to 0$.
2. *Every limit point of* $\{s^k\}$ *is a feasible point of the reformulated problem* (23).
3. *If, for all* $x \in X$, *a global solution of the lower level problem is found then any limit point is a feasible point of problem* (3).
4. *If* s^* *is a limit point of* $\{s^k\}$, *there exists an infinite set* $K_1 \subset \{0, 1, 2, \ldots\}$ *such that* $\lim_{k \in K_1} s^k = s^*$ *and* $\lim_{k \in K_1} z^k = s^*$.
5. *There exist an infinite set* $K_2 \subset \{0, 1, 2, \ldots\}$ *such that* $\lim_{k \in K_2} d_{tan}^k = 0$.
6. *There exists an infinite set* $K_3 \subset \{0, 1, 2, \ldots\}$ *and* $s^* \in \Omega \times \Delta$ *such that*

$$\lim_{k \in K_3} s^k = \lim_{k \in K_3} z^k = s^*, \ C(s^*) = 0 \quad \text{and} \quad \lim_{k \in K_3} d_{tan}^k = 0$$

7. *There exists a limit point and every limit point is a weak AGP point.*
8. *If, for all* $x \in X$, *a global solution of the lower level problem is found then any limit point is a strong AGP point.*

4.3 Sufficient conditions for the assumptions

The assumed compacity of the set Ω guarantees that for each $x \in X$, there exists a solution of the corresponding lower level problem. Assumptions A4, A5, A6 are made on the progress of the procedure and it is interesting to find conditions on the lower level problem that will ensure their validity. To verify A4 and A5, a sufficient condition is that for any given x, a stationary point of the lower level problem exists. The next hypotheses on the lower level problem in (3) guarantee that A6 is verified.

A7. For each $x \in X$, the solution of the lower-level problem $y(x) \in Y$ is a regular point, in the sense that the active constraints at $y(x)$ are linearly independent.
A8. The matrices $[\nabla_y^2 f(x, y(x)) + \sum_{i=1}^p \nabla_y^2 h_i(x, y(x))\mu(x)]$, associated with the second-order conditions of the lower level problems are positive definite in the subset of Y defined by the intersection of the null space of $\nabla h(x, y(x))^T$ with the set of vectors such that the j-th component is zero for any index j such that $y_j(x) = 0$.
A9. Every solution $s(x)$ of (8) verifies

$$y_i(x) + \gamma_i(x) > 0 \quad \forall i \in \{1, \ldots, m\}. \tag{25}$$

We proceed to prove, through some lemmas, that the error bound hypothesis A6 is verified under assumptions A1, A2, A3, A7, A8, A9.

We define

$$C'_{y,\mu,\gamma}(s(x)) = \begin{pmatrix} W(s(x)) & \nabla_y h(x, y(x)) & -I_m \\ \nabla_y h(x, y(x))^T & 0 & 0 \\ \operatorname{diag}(\gamma(x)) & 0 & \operatorname{diag}(y(x)) \end{pmatrix},$$

where $W(s(x)) = [\nabla_y^2 f(x, y(x)) + \sum_{i=1}^p \nabla_y^2 h_i(x, y(x))\mu_i(x)]$. Given a vector v, $D = \operatorname{diag}(v)$ is a diagonal matrix such that $d_{ii} = v_i$ and I_m is the $m \times m$ identity matrix.

Lemma 4.4 *The matrix $C'_{y,\mu,\gamma}(s(x))$ is non-singular for any $x \in X$.*

Proof We will prove that if a vector is in the null space of $C'_{y,\mu,\gamma}(s(x))$ it is necessarily the null vector.

Let $u \in \mathbb{R}^m$, $v \in \mathbb{R}^p$ and $w \in \mathbb{R}^m$ be such that:

$$C'_{y,\mu,\gamma}(s(x)) \begin{pmatrix} u \\ v \\ w \end{pmatrix} = 0$$

then,

$$W(s(x))u + \nabla_y h(x, y(x))v - w = 0, \tag{26}$$

$$\nabla_y h(x, y(x))^T u = 0, \tag{27}$$

$$\operatorname{diag}(\gamma(x))u + \operatorname{diag}(y(x))w = 0. \tag{28}$$

By A9 and (28) it follows that

$$u^T w = 0. \tag{29}$$

Pre-multiplying equation (26) by u^T we get

$$u^T W(s(x))u + u^T \nabla_y h(x, y(x))v - u^T w = 0,$$

by (27) and (29) we obtain

$$u^T W(s(x))u = 0,$$

and by (27) and A8 we get $u = 0$.

Now, we need to analyze the following equations

$$\nabla_y h(x, y(x))v - w = 0, \tag{30}$$

$$\mathrm{diag}(y(x))w = 0. \tag{31}$$

Let $J \subset \{1, \dots, m\}$ be such that $y_i(x) = 0$ for $i \in J$ and let $\bar{J} = \{1, \dots, m\} \backslash J$. Due to A9, for $i \in \bar{J}$ we have that $\gamma_i(x) = 0$.

By (31) we have that $w_i = 0$ for $i \in \bar{J}$. Therefore (30) reads

$$(\nabla_y h(x, y(x)) \quad -I_m) \begin{pmatrix} v \\ w \end{pmatrix} = 0$$

or

$$(\nabla_y h(x, y(x)) \quad -I_J) \begin{pmatrix} v \\ w_J \end{pmatrix} = 0$$

where I_J is a sub-matrix of the m-dimensional identity matrix I_m corresponding to the indices in J and w_J is the vector of the non zero components of w.

Assumption A7 means that $(\nabla_y h(x, y(x)) \quad -I_J)$ is a full-rank matrix, therefore

$$v = 0, \qquad w_J = 0.$$

We conclude that necessarily $u = 0$, $v = 0$ and $w = 0$ and $C'_{y,\mu,\gamma}(s(x))$ is nonsingular for any $x \in X$. □

To simplify the notation in the next lemmas we define $\upsilon(x) \in \mathbb{R}^{2m+p}$, $\upsilon(x) \equiv (y(x), \mu(x), \gamma(x))$ and $\upsilon \in \mathbb{R}^{2m+p)}$, $\upsilon \equiv (y, \mu, \gamma)$.

By the previous lemma we can define a function $\Upsilon : X \to \mathbb{R}^{2m+p \times 2m+p}$, such that

$$\Upsilon(x) = [C'_{y,\mu,\gamma}(x, \upsilon(x))]^{-1}.$$

Let $V(\upsilon(x), \varepsilon) \equiv \{\upsilon \in Y \times \Delta \mid \|\upsilon - \upsilon(x)\| < \varepsilon\}$.

Lemma 4.5 *There exist ε_1 and $\beta > 0$ such that for all $x \in X$, $|\Upsilon(x)| \le \beta$ and $\Upsilon(x)$ coincides with the local inverse operator of $C(x, .)$ for all $\upsilon \in V(\upsilon(x), \varepsilon_1)$.*

Proof The continuity of $\Upsilon(x)$ follows from the continuity of C' and $\upsilon(x)$. The compactness of X implies that there exists a number $\beta > 0$ such that $|\Upsilon(x)| \leq \beta$ for all $x \in X$. For each fixed value of $x \in X$ the continuously differentiable operator that associates to each υ the vector $C(x, \upsilon)$ verifies the hypothesis of the inverse function theorem at $\upsilon(x)$. Therefore, there exists $\varepsilon_1 > 0$ such that $C(x, .)$ admits a continuously differentiable local inverse operator $G(x) : C(x, V(\upsilon(x), \varepsilon_1)) \mapsto V(\upsilon(x), \varepsilon_1)$ and the Jacobian $[G(x)]'$ coincides with $\Upsilon(x)$. By hypotheses A1, A2 and the existence of the uniform bound β of $\Upsilon(x)$, ε_1 is independent of x. □

In the next lemma we prove the existence of a local error bound independent of $x \in X$.

Lemma 4.6 *There exist $\varepsilon > 0$ and $\beta > 0$ such that for all $x \in X$*

$$\|\upsilon - \upsilon(x)\| \leq \beta |C(x, \upsilon)|, \quad \forall \upsilon \in V(\upsilon(x), \varepsilon).$$

Proof The Taylor expansion of $G(x)$ around $C(x, \upsilon(x)) = 0$ reads

$$\begin{aligned} G(x)(C(x, \upsilon)) = {} & G(x)(C(x, \upsilon(x))) \\ & + [G(x)]'(C(x, \upsilon(x)))(C(x, \upsilon) - C(x, \upsilon(x))) \\ & + R(C(x, \upsilon) - C(x, \upsilon(x))) \end{aligned}$$

where

$$\lim_{C(x,\upsilon) \to C(x,\upsilon(x))} \frac{R(C(x, \upsilon) - C(x, \upsilon(x)))}{|C(x, \upsilon) - C(x, \upsilon(x))|} = 0.$$

Let ε_1 be as in Lemma 4.5, then using that $G(x)$ is a local inverse of $C(x, .)$ in $V(\upsilon(x), \varepsilon_1)$, the definition of $\Upsilon(x)$ and that $C(x, \upsilon(x)) = 0$, we get that, for all $\upsilon \in V(\upsilon(x), \varepsilon_1)$,

$$(x, \upsilon) = (x, \upsilon(x)) + \Upsilon(x) C(x, \upsilon) + R(C(x, \upsilon))$$

and

$$(x, \upsilon) - (x, \upsilon(x)) = \Upsilon(x) C(x, \upsilon) + R(C(x, \upsilon)),$$

where

$$\lim_{C(x,\upsilon) \to 0} \frac{R(C(x, \upsilon))}{|C(x, \upsilon)|} = 0.$$

Therefore

$$\|\upsilon - \upsilon(x)\| \leq \beta |C(x, \upsilon)| + |R(C(x, \upsilon))|. \tag{32}$$

Given any $\alpha > 0$, by the definition of R, there exists $\varepsilon_2 > 0$ such that

$$|C(x, \upsilon)| \leq \varepsilon_2 \quad \Rightarrow \quad \left\| \frac{R(C(x, \upsilon))}{|C(x, \upsilon)|} \right\| \leq \alpha.$$

By the continuity of C, the compactness of X and the fact that $C(x, \upsilon(x)) = 0$ there exists $\varepsilon_3 > 0$, independently of x, such that

$$|C(x, \upsilon)| \leq \varepsilon_2 \tag{33}$$

for all $\upsilon \in V(\upsilon(x), \varepsilon_3)$.

Therefore, for the given $\alpha > 0$

$$\|R(C(x, \upsilon))\| \leq \alpha |C(x, \upsilon)|, \tag{34}$$

for all

$$\upsilon \in V(\upsilon(x), \varepsilon_3) \cap V(\upsilon(x), \varepsilon_1). \tag{35}$$

Using (33), (34) and (32), we get

$$\|\upsilon - \upsilon(x)\| \leq (\beta + \alpha)|C(x, \upsilon)| \tag{36}$$

for all $\upsilon \in V(\upsilon(x), \varepsilon_3) \cap V(\upsilon(x), \varepsilon_1)$.

Taking $\varepsilon = \min\{\varepsilon_3, \varepsilon_1\}$ and renaming $\beta + \alpha$ as β we get the desired result. □

Now we proceed to prove the existence of a global error bound. Although we assume in the next lemma that $y(x)$ is the unique stationary point of the lower level problem, it is possible to extend the arguments used for an arbitrary number or stationary points satisfying A7, A8 and A9.

Lemma 4.7 *There exists* $\beta > 0$ *such that*

$$\|\upsilon - \upsilon(x)\| \leq \beta |C(x, \upsilon)|$$

for all $(x, \upsilon) \in \Omega \times \Delta$.

Proof By Lemma 4.6 there exist ε and β such that

$$\|\upsilon - \upsilon(x)\| \leq \beta |C(x, \upsilon)|$$

for all $\upsilon \in V(\upsilon(x), \varepsilon)$. For each $x \in X$ consider the following set

$$\Theta^\varepsilon(x) \equiv \{\upsilon \in Y \times \Delta \mid \|\upsilon - \upsilon(x)\| \geq \varepsilon\}.$$

The set $\Theta^\varepsilon = \bigcap_{x \in X} \Theta^\varepsilon(x)$ is compact and $\upsilon(x) \notin \Theta^\varepsilon$ for all $x \in X$.

Due to the compactness of Θ^ε, $|C(x, \upsilon)|$ assumes maximum and minimum values in Θ^ε, WM and Wm, respectively, independently of x. $Wm > 0$ because $\upsilon(x) \notin \Theta^\varepsilon$. Similarly $\|\upsilon - \upsilon(x)\|$ also assumes maximum and minimum values UM and Um in Θ^ε, respectively. Moreover, $Um > 0$ because $\upsilon(x) \notin \Theta^\varepsilon$.

For any $(x, \upsilon) \in \Omega \times \Delta$ one of the next two possibilities is true,

$$\upsilon \in V(\upsilon(x), \varepsilon)$$

or

$$\upsilon \notin V(\upsilon(x), \varepsilon).$$

In the first case we apply directly Lemma 4.5. In the second case we obtain, using the upper and lower bounds UM and Wm, that

$$\frac{\|\upsilon - \upsilon(x)\|}{|C(x,\upsilon)|} \leq \frac{UM}{Wm}.$$

Renaming $\max\{\beta, \frac{UM}{Wm}\}$ as β we finish our proof. □

Theorem 4.8 *Let $(x,\upsilon) \in \Omega \times \Delta$ be such that $|C(x,\upsilon)| \neq 0$ and $r \in [0\,1)$. There exist $\bar{\upsilon} = (\bar{y}, \bar{\mu}, \bar{\gamma}) \in \Omega \times \Delta$ and a positive constant β, such that β is independent of x, that verify*

$$|C(x,\bar{\upsilon})| \leq r|C(x,\upsilon)| \tag{37}$$

and

$$\|(x,\bar{\upsilon}) - (x,\upsilon)\| \leq \beta|C(x,\upsilon)|. \tag{38}$$

Proof By the uniform continuity of C, there exists an $\epsilon > 0$, independent of x, such that if

$$\|(x,\bar{\upsilon}) - (x,\upsilon(x))\| \leq \epsilon,$$

then

$$|C(x,\bar{\upsilon})| \leq r|C(x,\upsilon)|, \tag{39}$$

where $(x,\upsilon(x))$ is the unique solution of $C(x,\upsilon) = 0$.

Let $(x,\bar{\upsilon})$ be a point within distance ϵ from the solution point.

By the previous lemmas we know that there exists a positive number β, independent of x, such that

$$\|(x,\upsilon) - (x,\upsilon(x))\| \leq \beta|C(x,\upsilon)| \tag{40}$$

and

$$\|(x,\bar{\upsilon}) - (x,\upsilon(x))\| \leq \beta|C(x,\bar{\upsilon})|. \tag{41}$$

Applying the triangular inequality and (39) we obtain

$$\|(x,\bar{\upsilon}) - (x,\upsilon)\| \leq \beta(1+r)|C(x,\upsilon)|. \tag{42}$$

Finally, renaming $\beta(1+r)$ as β we get the desired result. □

5 Numerical experiments

Algorithm 3.1 was implemented in `Fortran 77` in a Pentium(R) 4, 2.20 GHz, 512 MB of RAM. The main algorithmic parameters used were $r = 0.99$, $\theta_{-1} = 0.5$, $\delta_0 = 10$. The first two were inherited from Martínez and Pilotta [23], whereas the third was a choice that fitted in well with all the tests. The penalty parameter θ is decreased when the feasibility progress in Step 2 of the algorithm largely exceeds the optimality progress. Large feasibility improvements generally happen at the first iterations, therefore a significant decrease of θ will probably occur in the beginning. This is the motivation for using a non monotonic strategy to update the penalty parameter. The parameter β is not given explicitly. Instead, in problems where the sufficient conditions for its existence fail to be true, we monitor the growth of the quotients $\|z^k - s^k\|/|C(s^k)|$. We finish the restoration step when the feasibility error is sufficiently reduced and the corresponding quotient is not greater than ten times the quotient obtained in the restoration step of the previous iteration. If this is not possible we stop the algorithm and declare failure in this step. Actually this situation never happened and we comment this fact later. Convergence at Step 2 was declared whenever $|C(s^k)| \le 10^{-4}$. A worth mentioning aspect that helps to improve the performance of the algorithm is to count on good approximations for the Lagrange multipliers, not only for the lower level problem at Step 2, but also for the reformulated constraints at Step 5.

To solve the restoration and the optimization subproblems for the bilevel instances we used the box constrained solver `GENCAN` of Birgin and Martínez [3] and the active set software `MINOS` [27], which also provided estimates to update, in Step 5, the multipliers λ associated to the Lagrangian of the reformulation of the bilevel problem by its KKT system.

Three family of bilevel test problems were considered: linear and quadratic problems given in [18], and nonlinear ones presented in [7]. The results are reported in Tables 1, 2 and 3, respectively, where the following notation was used: in the first column we include the numbering or name used in the corresponding source; n and m indicate the dimension of the outer and inner variables x and y, respectively; $(x, y)_{\text{Initial}}$ denotes the initial point (the remaining variables were all set to zero); column Iter gives the number of iterations performed by the inexact-restoration algorithm; $(x, y)_{\text{Best}}$ reports the point with the best functional value F_{Best}, extracted from the literature; and $(x, y)_{\text{IR}}$ provides the solution computed by the algorithm, with corresponding functional value F_{IR}. We should mention that, in Table 3, we did not have access to the points $(x, y)_{\text{Best}}$ nor to the value F_{Best} for problem `BIPA1`.

As far as the choices for the initial points, we have always tried the suggestion available in the literature, among a few other choices. To show the viability of our approach we only report the results that produced the solution with smallest objective function value. For instance, the initial points for problems `BIPA2`, `BIPA3` and `BIPA5` are not the same suggested in [8], starting from which we have obtained local solutions distinct from those reported in Table 3.

To assess the effect of the assumptions A7–A9 on the problem, we have analyzed the performance of Algorithm 3.1 applied to the following bilevel problem

$$\begin{aligned} &\underset{x,y}{\text{Minimize}} \quad x^2 + y^2 \\ &\text{s.t.} \quad \begin{cases} x \ge 0 \\ y = \arg\min (y-x)^2 \\ \text{s.t.} \quad 0 \le y \le x. \end{cases} \end{aligned} \tag{43}$$

Clearly, its solution is the origin, a non-regular point where strict complementarity fails. Applying Algorithm 3.1 to problem (43), starting from $(x_0, y_0) = (5, 1)$ (and

Table 1 Linear bilevel problems solved by inexact-restoration technique

Test problem	n	m	$(x, y)_{\text{Initial}}$	Iter	$(x, y)_{\text{Best}}$ F_{Best}	$(x, y)_{\text{IR}}$ F_{IR}
1 (9.2.2)	1	2	(4, 0, 0)	2	(5, 4, 2) −13	(5, 4, 2) −13
2 (9.2.3)	1	1	(5, 0, 0)	2	(4, 4) −16	(4, 4) −16
3 (9.2.4)	2	6	(0.5, 0.5, 0.5, 0.5, 0.5, 0.5, 0.5, 0.5)	3	(0, 0.9, 0, 0.6, 0.4, 0, 0, 0) −29.2	(0, 0.9, 0, 0.6, 0.4, 0, 0, 0) −29.2
4 (9.2.5)	1	1	(2, 2)	11	(19, 14) −37	(19, 14) −37
5 (9.2.6)	1	2	(0.5, 0.5, 0)	2	(1, 0, 0) −1	(0, 0, 1) −1
6 (9.2.7)	1	1	(15, 15)	2	(16, 11) −49	(16, 11) −49
7 (9.2.8)	2	3	(0.5, 0.5, 0.5, 0.5, 0.5)	3	(0, 0.9, 0, 0.6, 0.4) −26	(0, 0.9, 0, 0.6, 0.4) −26
8 (9.2.9)	2	2	(1, 1, 1, 1)	2	(1, 0, .5, 1) −1.75	(2, 0, 1.5, 0) −3.25
9 (9.2.10)	1	1	(1, 1)	2	(0.889, 2.222) 3.111	(0.889, 2.222) 3.111
10 (9.2.11)	2	2	(1, 1, 0, 0)	2	(2, 0, 1.5, 0) −3.25	(2, 0, 1.5, 0) −3.25

remaining variables set to zero), after 3 iterations the origin is reached, under the prescribed accuracy, and the quotients $\|z^k - s^k\|/|C(s^k)|$ remain bounded by 1.14.

In [6], Algorithm 3.1 is used to solve truss topology design problems. In these problems we know that strict complementarity is not verified in some feasible non-optimal points. In these points the function $\upsilon(x) = (y(x), \mu(x), \gamma(x))$ is not differentiable. We initialized the algorithm with these points and we never observed a dramatic growth of the critical quotients. This unexpected performance deserves future research. We illustrate this situation with the following bilevel problem, that models a one dimensional simplification of the problem of minimizing the deformation energy subject to volume constraints and frictionless contact

$$\underset{x,\nu}{\text{Minimize}} \quad \tfrac{1}{2}k(x)\nu^2$$

$$\text{s.t.} \quad \begin{cases} 0 \le x \le \frac{V}{L} \\ \nu = \arg\min \frac{1}{2}k(x)\nu^2 - P\nu \\ \text{s.t.} \quad -d \le \nu \le 0 \end{cases}$$

where x represents the cross section area of a vertical truss with the upper extreme not allowed to displace, L its fixed length, V is the maximal volume allowed, P a load

Table 2 Quadratic bilevel problems solved by inexact-restoration technique

Test problem	n	m	$(x, y)_{\text{Initial}}$	Iter	$(x, y)_{\text{Best}}$ F_{Best}	$(x, y)_{\text{IR}}$ F_{IR}
1 (9.3.2)	1	1	$(2, 2)$	3	$(1, 0)$	$(1, 0)$
					17	17
2 (9.3.3)	1	1	$(1, 1)$	4	$(10, 10)$	$(10, 10)$
					100	100
3 (9.3.4)	2	2	$(5, 5, 0, 0)$	2	$(0, 0, -10, -10)$	$(0, 0, -10, -10)$
					0	0
4 (9.3.5)	1	2	$(0, 0, 0)$	3	$(3, 1, 2)$	$(3, 1, 2)$
					0.5	0.5
5 (9.3.6)	2	1	$(0, 0)$	2	$(1, 3)$	$(1, 3)$
					5	5
6 (9.3.7)	2	2	$(0, 0, 0, 0)$	2	$(0.5, 0.5, 0.5, 0.5)$	$(0.5, 0.5, 0.5, 0.5)$
					-1	-1
7 (9.3.8)	1	1	$(2, 2)$	2	$(1, 0)$	$(1, 0)$
					17	17
8 (9.3.9)	1	1	$(0, 0)$	3	$(0.25, 0)$	$(0.25, 0)$
					1.5	1.5
9 (9.3.10)	1	2	$(1, 1, 1)$	3	$(2, 6, 0)$	$(2, 6, 0)$
					2	2

Table 3 Nonlinear bilevel problems solved by inexact-restoration technique

Problem	n	m	$(x, y)_{\text{Initial}}$	Iter	F_{Best}	$(x, y)_{\text{IR}}$ (F_{IR})
`BIPA1`	1	1	$(10, 10)$	2	–	$(6.08, 4.46)$ (230.26)
`BIPA2`	1	1	$(3, 0)$	3	17	$(1, 0)$ (17)
`BIPA3`	1	1	$(1, 1)$	2	2	$(4, 0)$ (2)
`BIPA4`	1	1	$(1.5, 2.25)$	2	88.79	$(0, 0.6)$ (88.79)
`BIPA5`	1	2	$(2, 2, 2)$	2	2.75	$(1.94, 0, 1.21)$ (2.75)

at the free extreme ($P < 0$ by convention), v is the vertical displacement of the truss (negative by convention), d is the distance between the free extreme and an obstacle. Finally, $k(x) = \frac{E}{L}x$ is the stiffness coefficient and E is the Young modulus of the material of which the truss is made.

It is easy to see that the solution function $v(x)$, whose graphic represents the feasible region for the upper level problem determined by the lower level problem is given by:

$$v(x) = \begin{cases} -d & \text{if } 0 \le x < -\frac{LP}{Ed}, \\ (\frac{LP}{E})\frac{1}{x} & \text{if } -\frac{LP}{Ed} \le x \le \frac{V}{L}. \end{cases}$$

Substituting this expression in the upper level objective function, we obtain explicitly the expression as a function just of x,

$$F(x)=\begin{cases}(\frac{Ed^2}{2L})x & \text{if } 0\le x<-\frac{LP}{Ed},\\ (\frac{LP^2}{2E})\frac{1}{x} & \text{if } -\frac{LP}{Ed}\le x\le \frac{V}{L}.\end{cases}$$

We can easily conclude that the global minimizer of $F(x)$ is $x=0$ and the unique local minimizer is $x=\frac{V}{L}$. The value of x that breaks the function F in two differentiable parts is a maximum. If we start our algorithm at this point it will abandon it and find the global minimizer.

If the lower level problem is a variational inequality problem and a good algorithm is available to solve it, the inexact restoration strategy can be applied. Therefore the generalized bilevel problem, as it is called in the literature, can be solved without reformulating it as a single-level optimization problem. To address this instance, a `Matlab` (version 6.1) implementation was developed, where the restoration was performed combining the projection algorithm of Solodov and Svaiter [30] to obtain an approximate solution of the variational problem, with a least-squares strategy to update the multipliers μ, γ of the lower-level constraints. The optimization subproblem of minimizing the Lagrangian in the intersection of the linear approximation π_k with the trust region was solved using the internal routine `fmincon` of `Matlab`.

As the optimality conditions of the lower level in bilevel instances may be written as a variational inequality problem, test 7 from Table 1 and problems 4 to 8 from Table 2 were solved with this strategy as well, reaching the same objective function value F, and usually performing an additional iteration.

We have also solved two test problems from [15], whose results are reported in Table 4. The first problem is a legitimate generalized bilevel programming problem, and the second one is a bilevel problem with a constraint in the upper level that is addressed by means of an external penalization, as done by [15]. In both cases a known solution was obtained.

By analyzing the reported results one can see that the proposed approach is a valid alternative to address nonlinear bilevel programming problems. Our preliminary results indicate a promising perspective. We show that this approach is reliable and its efficiency will certainly depend on the methods used in each of the algorithm's phases, a choice that the user should make.

Table 4 Generalized bilevel programming problems solved by inexact-restoration technique

Test	n	m	$(x,y)_{\text{Initial}}$	Iter	$(x,y)_{\text{Best}}$ F_{Best}	$(x,y)_{\text{IR}}$ F_{IR}
1 (problem 9)	2	2	(0, 0, 0, 0)	6	(5, 9, 5, 9) 0	(5, 9, 5, 9) 10^{-17}
2 (problem 7)	2	2	(20, 35, 0, 0)	4	(25, 30, 5, 10) 5	(25, 30, 5, 10) 4.9999767

6 Conclusions and future research

We introduced a new algorithm for solving bilevel problems that preserves the two-level structure of the problem, and is based in the inexact restoration technique. In the feasibility phase the lower level problem can be solved inexactly taking advantage of its properties and the user is free to use special purpose solvers, mainly global optimization algorithms if they are available. In the optimality phase any algorithm for minimization with linear constraints can be used. The decrease of a merit function, that combines a Lagrangian of the upper level objective with a measure of stationarity of the lower level problem, is obtained using a trust-region approach.

We prove that the algorithm is well defined and converges to feasible points under mild conditions. Under more restrictive assumptions we prove that a sequence generated by the algorithm converges to feasible points that satisfy an AGP optimality condition. These conditions are also imposed in most of the previous work in this field that present a convergence result, with the exception of the strict complementarity at the solutions of the lower level problem. The possibility of relaxing this hypothesis depends on the existence of error bounds when strict complementarity fails to be true, using other feasibility measures. It could also be possible to use other merit functions or optimality conditions, or a combination of all these. We are doing research on these questions. Some interesting ideas that could be exploited for this purpose are in [14, 16, 19].

We present numerical experiments that show that the algorithm is capable of solving the problems, even if this hypothesis is not verified. This suggests that the condition might not be necessary and motivates us to extend our theoretical result. However there exist many interesting problems that satisfy all the hypotheses. We believe that our approach gives a valid alternative to methods that strongly use the implicit dependence of the upper level variable on the lower level one. We do not need to evaluate the derivative of this implicit function as we treat these variables as "independent".

The results of the computational experiments encourage us to design a special program that should allow the user to take profit of his or hers preferred nonlinear programming routines.

References

1. Andreani, R., Martínez, J.M.: On the solution of mathematical programming problems with equilibrium constraints. Math. Method. Oper. Res. **54**, 345–358 (2001)
2. Bard, J.F.: Practical Bilevel Optimization: Algorithms and Applications. Kluwer Academic, Dordrecht (1998)
3. Birgin, E.J., Martínez, J.M.: Large-scale active-set box-constrained optimization method with spectral projected gradients. Comput. Optim. Appl. **23**, 101–125 (2002)
4. Bracken, J., McGill, J.: Mathematical programs with optimization problems in the constraints. Oper. Res. **21**, 37–44 (1973)
5. Candler, W., Norton, R.: Multilevel programming. Technical Report 20, World Bank Development Research Center, Washington DC, USA
6. Castro, S.L.C., Fancello, E.A., Friedlander, A., Santos, S.A.: Computational experiments in truss topology design with an inexact restoration method. Technical Report (2006), available at http://www.ime.unicamp.br/~friedlan

7. Colson, B.: BIPA (Bilevel programming with approximation methods): Software guide and test problems. Technical Report CRT-2002-38, Centre de Recherche sur les Transports, Université de Montréal, Montréal, QC, Canada (2002)
8. Colson, B., Marcotte, P., Savard, G.: A trust-region method for nonlinear bilevel programming: Algorithm and computational experience. Comput. Optim. Appl. **30**(3), 211–227 (2005)
9. Colson, B., Marcotte, P., Savard, G.: Bilevel programming: A survey. 4OR: Q. J. Oper. Res. **3**, 87–107 (2005)
10. Dai, Y.: Convergence properties of the BFGS Algorithm. SIAM J. Optim. **13**(3), 693–701 (2002)
11. Dempe, S.: Foundations of Bilevel Programming. Nonconvex Optimization and its Applications Series, vol. 61. Kluwer Academic, Dordrecht (2002)
12. Dempe, S.: Annottated bibliography on bilevel programming and mathematical problems with equilibrium constraints. Optimization **52**, 333–359 (2003)
13. Dempe, S.: Bilevel programming—A survey. Technical Report TU 2003-11, Bergakademie Freiberg (2003)
14. Facchinei, F., Fischer, A., Kanzow, C.: On the accurate identification of active constraints. SIAM J. Optim. **9**(1), 14–32 (1998)
15. Facchinei, F., Jiang, H., Qi, L.: A smoothing method for mathematical programs with equilibrium constraints. Math. Program. **85**, 107–134 (1999)
16. Fischer, A.: Modified Wilson method for nonlinear programs with nonunique multipliers. Math. Oper. Res. **24**, 699–727 (1999)
17. Fletcher, R., Leyffer, S., Ralph, D., Scholtes, S.: Local convergence of SQP methods for mathematical programs with equilibrium constraints. SIAM J. Optim. **17**(1), 259–286 (2006)
18. Floudas, C.A., et al.: Handbook of Test Problems in Local and Global Optimization. Kluwer Academic, Dordrecht (1999)
19. Hager, W.W., Gowda, M.S.: Stability in the presence of degeneracy and error estimation. Math. Program. **85**, 181–192 (1999)
20. Luo, Z.Q., Pang, J.S., Ralph, D.: Mathematical Programs with Equilibrium Constraints. Cambridge University Press, Cambridge (1996)
21. Martínez, J.M.: Two-phase model algorithm with global convergence for nonlinear programming. J. Optim. Theory Appl. **96**(2), 397–436 (1998)
22. Martínez, J.M.: Inexact-restoration method with Lagrangian tangent decrease and new merit function for nonlinear programming. J. Optim. Theory Appl. **111**(1), 39–58 (2001)
23. Martínez, J.M., Pilotta, E.A.: Inexact-restoration algorithm for constrained optimization. J. Optim. Theory Appl. **104**, 135–163 (2000)
24. Martínez, J.M., Pilotta, E.A.: Inexact restoration methods for nonlinear programming: Advances and perspectives. In: Qi, L.Q., Teo, K.L., Yang, X.Q. (eds.) Optimization and Control with Applications, pp. 271–292. Springer, New York (2005)
25. Martínez, J.M., Svaiter, B.F.: A practical optimality condition without constraint qualifications for nonlinear programming. J. Optim. Theory Appl. **118**, 117–133 (2003)
26. Mascarenhas, W.F.: The BFGS method with exact line searches fails for non-convex objective functions. Math. Program. Ser. B **99**, 49–61 (2004)
27. Murtagh, B.A., Saunders, M.A.: MINOS 5.4 User's Guide. Technical Report SOL 83-20, Stanford University (1995)
28. Rockafellar, R.T., Wets, R.J.B.: Variational Analysis. Springer, Berlin (1998)
29. Shimizu, K., Ishizuka, Y., Bard, J.F.: Nondifferentiable and Two-Level Mathematical Programming. Kluwer Academic, Dordrecht (1997)
30. Solodov, M.V., Svaiter, B.F.: A new projection method for variational inequality problems. SIAM J. Control Optim. **37**(3), 756–776 (1999)
31. Vicente, L.N., Calamai, P.H.: Bilevel and multilevel programming: A bibliography review. J. Global Optim. **5**, 1–23 (1994)