On Numerical Solution of Specially Constructed Quadratic Bilevel Test Problems

Andrei V. Orlov
Matrosov Institute for System Dynamics and Control Theory of Siberian Branch of RAS
Lermontov St., 134, Irkutsk, 664033, Russia.
anor@icc.ru

Abstract

This paper addresses the bilevel programming problems (BPPs) with the quadratic objective functions at the upper and the lower levels. The new solution method for such BPPs is developed. The main feature of the approach proposed is employment of the original A.S. Strekalovsky's Global Search Theory. Numerical testing of the method on specially constructed instances demonstrated the efficiency of the approach.

1 Introduction

A hierarchy has been one of the most principal paradigms in mathematical programming in recent years. Development of the new efficient numerical methods for solving different classes of bilevel programming problems (BPPs) is a challenge in the modern theory and methods of Mathematical Optimization [Pang, 2010].

In this work we investigate the BPPs with the quadratic objective functions at the upper and the lower levels. The solution method for such BPPs is based on the equivalent representation of a quadratic bilevel problem as a nonconvex optimization problem. For this purpose, we use the optimality conditions for the lower level problem and the penalty approach [Dempe, 2002]. To solve the resulting nonconvex problem, we apply the special A.S. Strekalovsky's Global Search Theory [Strekalovsky, 2003, Strekalovsky, 2014].

The field of test quadratic bilevel problems is constructed according to the approach by Calamai and Vicente [Calamai & Vicente, 1994]. It allows us to build problems of various dimension and complexity. The final section of the paper presents and analyzes the results of numerical solution of generated problems.

2 Problem Formulation and Its Reduction

Consider the following quadratic-quadratic problem of bilevel optimization in its optimistic statement. In this case, according to the theory [Dempe, 2002], at the upper level we perform the minimization with respect to the

In: Yu. G. Evtushenko, M. Yu. Khachay, O. V. Khamisov, Yu. A. Kochetov, V.U. Malkova, M.A. Posypkin (eds.): Proceedings of the OPTIMA-2017 Conference, Petrovac, Montenegro, 02-Oct-2017, published at http://ceur-ws.org

variables of both levels which are in cooperation:

$$\left.\begin{array}{c} F(x,y) := \frac{1}{2}\langle x, Cx\rangle + \langle c, x\rangle + \frac{1}{2}\langle y, Dy\rangle + \langle d, y\rangle \downarrow \min\limits_{x,y}, \\ x \in X := \{x \in I\!R^m | Ax \le b\}, \\ y \in Y_*(x) := \operatorname{Arg}\min\limits_{y}\{\frac{1}{2}\langle y, D_1 y\rangle + \langle d_1, y\rangle + \langle x, Qy\rangle \mid y \in Y(x)\}, \\ Y(x) \stackrel{\triangle}{=} \{y \in I\!R^n | A_1 x + B_1 y \le b_1\}, \end{array}\right\} \qquad (\mathcal{QBP})$$

where $A \in I\!R^{p\times m}$, $A_1 \in I\!R^{q\times m}$, $B_1 \in I\!R^{q\times n}$, $C \in I\!R^{m\times m}$, $D, D_1 \in I\!R^{n\times n}$, $Q \in I\!R^{m\times n}$, $c \in I\!R^m$, $d, d_1 \in I\!R^n$, $b \in I\!R^p$, $b_1 \in I\!R^q$. Additionally, $C = C^T \ge 0$, $D = D^T \ge 0$, $D_1 = D_1^T \ge 0$.

Here we will present a brief scheme of the reduction bilevel problem $(\mathcal{QBP})$ to a single-level problem. Additional details about this issue can be found in [Orlov, 2017].

According to the well known KKT-approach [Dempe, 2002, Strekalovsky et al., 2010], we can reduce Problem $(\mathcal{QBP})$ to the following single-level problem replacing the lower level problem in $(\mathcal{QBP})$ with its optimality conditions:

$$\left.\begin{array}{c} F(x,y) \downarrow \min\limits_{x,y,v}, \qquad Ax \le b, \\ D_1 y + d_1 + xQ + vB_1 = 0, \qquad v \ge 0, \qquad A_1 x + B_1 y \le b_1, \qquad \langle v, A_1 x + B_1 y - b_1\rangle = 0, \end{array}\right\} \qquad (\mathcal{DCC})$$

where the variable v is the vector of Lagrange multipliers.

The next step of our reduction is to use the so-called penalty approach [Dempe, 2002]. Thus we obtain the following σ-parametrized nonconvex problem with a convex feasible set:

$$\Phi(x,y,v) := \frac{1}{2}\langle x, Cx\rangle + \langle c, x\rangle + \frac{1}{2}\langle y, Dy\rangle + \langle d, y\rangle + \sigma\,\langle v, b_1 - A_1 x - B_1 y\rangle \downarrow \min_{x,y,v}, \quad (x,y,v) \in D, \qquad (\mathcal{DC}(\sigma))$$

where $\sigma > 0$ is a penalty parameter and $D := \{(x,y,v) \mid Ax \le b,\ D_1 y + d_1 + xQ + vB_1 = 0,\ v \ge 0,\ A_1 x + B_1 y \le b_1\}$. We can prove the standard results about the connection between problems $(\mathcal{DCC})$ and $(\mathcal{DC}(\sigma))$ [Dempe, 2002, Strekalovsky et al., 2010]. In particular, if the triplet $(x(\sigma), y(\sigma), v(\sigma))$ is a solution of problem $(\mathcal{DC}(\sigma))$ and $\langle v(\sigma), b_1 - A_1 x(\sigma) - B_1 y(\sigma)\rangle = 0$, then $(x(\sigma), y(\sigma), v(\sigma))$ turns out to be a solution of problem $(\mathcal{DCC})$. Moreover, it can be shown that there exists a finite value of σ with $\langle v(\sigma), b_1 - A_1 x(\sigma) - B_1 y(\sigma)\rangle = 0$.

Note that for a fixed σ problem $(\mathcal{DC}(\sigma))$ has a bilinear structure and belongs to the class of d.c. minimization problems [Strekalovsky, 2003, Strekalovsky, 2014] with a convex feasible set. It is easy to see that the objective function of $(\mathcal{DC}(\sigma))$ can be represented as a difference of two convex functions [Strekalovsky, 2003, Strekalovsky, 2014].

We will employ, for example, the following d.c. representation based on the known property of a scalar product:

$$\Phi(x,y,v) = g(x,y,v) - h(x,y,v), \tag{1}$$

where $g(x,y,v) = \frac{1}{2}\langle x, Cx\rangle + \langle c, x\rangle + \frac{1}{2}\langle y, Dy\rangle + \langle d, y\rangle + \sigma\langle b_1, v\rangle + \frac{\sigma}{4}\|A_1 x - v\|^2 + \frac{\sigma}{4}\|B_1 y - v\|^2$, $h(x,y,v) = \frac{\sigma}{4}\|A_1 x + v\|^2 + \frac{\sigma}{4}\|B_1 y + v\|^2$. Note that the so-called *basic nonconvexity* in Problem $(DC(\sigma))$ is provided by the function h (for more details, refer to [Strekalovsky, 2003, Strekalovsky, 2014]).

So we can apply the Global Search Theory in d.c. minimization problems [Strekalovsky, 2003, Strekalovsky, 2014] to seek a global solution to Problem $(\mathcal{DC}(\sigma))$ with a fixed σ.

3 Global Optimality Conditions and Solution Algorithm

The Global Search Procedure is based on the Global Optimality Conditions (GOCs) developed by A.S. Strekalovsky [Strekalovsky, 2003, Strekalovsky, 2014]. In particular, the necessary Global Optimality Conditions have the following form in terms of Problem $(\mathcal{DC}(\sigma))$.

Theorem 1. [Strekalovsky, 2003, Strekalovsky, 2014]. If the feasible point (x^*, y^*, v^*) is a (global) solution to Problem $(\mathcal{DC}(\sigma))$, then $\forall (z, u, w, \xi) \in R^{m+n+q+1}$:

$$h(z,u,w) = \xi - \zeta, \qquad \zeta := \Phi(x^*, y^*, v^*), \qquad g(x,y,v) \le \xi \le \sup_{x,y,v}(g, D), \tag{2}$$

the following inequality holds

$$g(x,y,v)-\xi \geq \langle \nabla h(z,u,w),(x,y,v)-(z,u,w)\rangle \qquad \forall (x,y,v)\in D. \tag{3}$$

The conditions (2)-(3) possess the so-called algorithmic (constructive) property: if the GOCs are violated, we can construct a feasible point that will be better than the (critical) point in question (the value of the objective function in the new point will be less) [Strekalovsky, 2003, Strekalovsky, 2014, Orlov, 2017].

The constructive property is a foundation of global search algorithms for nonconvex problems [Strekalovsky, 2003, Strekalovsky, 2014]. Taking into account the d.c. representation (1), on the basis of Theorem 1, the Global Search Algorithm (GSA) for quadratic bilevel problems can be formulated in the following way.

Let there be given a starting point $(x_0,y_0,v_0)\in D$, numerical sequences $\{\tau_k\}$, $\{\delta_k\}$. (τ_k, $\delta_k>0$, $k=0,1,2,...,\tau_k\downarrow 0$, $\delta_k\downarrow 0,(k\to\infty)$), a set $Dir=\{(\bar z^1,\bar u^1,\bar w^1),...,(\bar z^N,\bar u^N,\bar w^N)\in I\!R^{m+n+q}\ |(\bar z^i,\bar u^i,\bar w^i)\neq 0$, $i=1,...,N\}$, the numbers $\xi_-\stackrel{\triangle}{=}\inf(g,D)$ and $\xi_+\stackrel{\triangle}{=}\sup(g,D)$, and the algorithm's parameters M and η.

Step 0. Set $k:=0$, $(\bar x^k,\bar y^k,\bar v^k):=(x_0,y_0,v_0)$, $i:=1$, $\xi:=\xi_-$, $\triangle\xi=(\xi_+-\xi_-)/M$.

Step 1. Proceeding from the point $(\bar x^k,\bar y^k,\bar v^k)$ by a local search method, build a τ_k-critical point $(x^k,y^k,v^k)\in D$ to Problem $(\mathcal{DC}(\sigma))$. Set $\zeta_k:=\Phi(x^k,y^k,v^k)$.

Step 2. Using $(\bar z^i,\bar u^i,\bar w^i)\in Dir$, construct a point (z^i,u^i,w^i) of the approximation $\mathcal{A}_k=\{(z^1,u^1,w^1),...,(z^N,u^N,w^N)\mid h(z^i,u^i,w^i)=\xi-\zeta_k,\quad i=1,...,N\}$ of the level surface $\mathcal{U}(\zeta_k)=\{(x,y,v)\mid h(x,y,v)=\xi-\zeta_k\}$ of the convex function $h(x,y,z)$, such that $h(z^i,u^i,w^i)=\xi-\zeta_k$.

Step 3. If $g(z^i,u^i,w^i)>\xi+\eta\xi$, then $i:=i+1$ and return to Step 2.

Step 4. Find a δ_k-solution $(\bar x^i,\bar y^i,\bar v^i)$ of the following linearized problem:

$$g(x,y,v)-\langle\nabla h(z^i,u^i,w^i),(x,y,v)\rangle\downarrow\min_{x,y,v},\quad (x,y,v)\in D. \tag{$\mathcal{PL}_i$}$$

Step 5. Starting at the point $(\bar x^i,\bar y^i,\bar v^i)$, build a τ_k-critical point $(\hat x^i,\hat y^i,\hat v^i)\in D$ to Problem $(\mathcal{DC}(\sigma))$ by means of the local search method.

Step 6. If $\Phi(\hat x^i,\hat y^i,\hat v^i)\geq\Phi(x^k,y^k,v^k)$, $i<N$, then set $i:=i+1$ and return to Step 2.

Step 7. If $\Phi(\hat x^i,\hat y^i,\hat v^i)\geq\Phi(x^k,y^k,v^k)$, $i=N$ and $\xi<\xi_+$, then set $\xi:=\xi+\triangle\xi$, $i:=1$ and go to Step 2.

Step 8. If $\Phi(\hat x^i,\hat y^i,\hat v^i)<\Phi(x^k,y^k,v^k)$, then set $\xi:=\xi_-$, $(\bar x^{k+1},\bar y^{k+1},\bar v^{k+1}):=(\hat x^i,\hat y^i,\hat v^i)$, $k:=k+1$, $i:=1$ and return to Step 1.

Step 9. If $\Phi(\hat x^i,\hat y^i,\hat v^i)\geq\Phi(x^k,y^k,v^k)$, $i=N$ and $\xi=\xi_+$, then stop. (x^k,y^k,v^k) is the obtained solution of the problem.

It can be readily seen that this algorithm is not an algorithm in the conventional sense, because some of its steps are not specified. For example, we do not know how to construct a starting point and the set Dir, how to implement a local search, how to solve the problem $(\mathcal{PL}_i)$ etc. These issues will be considered below.

4 Implementation of the Global Search Algorithm

First, to construct a feasible starting point, we used the projection of the chosen infeasible point (x^0,y^0,v^0) onto the feasible set D by solving the following quadratic programming problem:

$$\frac{1}{2}\|(x,y,v)-(x^0,y^0,v^0)\|^2\downarrow\min_{x,y,v},\qquad (x,y,v)\in D. \tag{$\mathcal{PR}(x^0,y^0,v^0)$}$$

The solution to Problem $(\mathcal{PR}(x^0,y^0,v^0))$ was taken as a starting point $(x_0,y_0,v_0)\in D$. In this work $(x^0,y^0,v^0)=(0,0,0)$. The value of the penalty parameter σ was chosen experimentally: $\sigma=10$.

The local search (see Steps 1 and 5) can be based on the consecutive solution of the following convex quadratic (QP) and linear programming (LP) problems derived from Problem $(\mathcal{DC}(\sigma))$:

$$\left.\begin{array}{c}\frac{1}{2}\langle x,Cx\rangle+\langle c,x\rangle+\frac{1}{2}\langle y,Dy\rangle+\langle d,y\rangle-\sigma\,\langle v^sA_1,x\rangle-\sigma\,\langle v^sB_1,y\rangle\downarrow\min\limits_{x,y},\\ Ax\leq b,\quad A_1x+B_1y\leq b_1,\quad D_1y+d_1+xQ+v^sB_1=0.\end{array}\right\} \tag{$\mathcal{QP}(v^s)$}$$

$$\left.\begin{array}{c}\langle b_1-A_1x^s-B_1y^s,v\rangle\downarrow\min\limits_{v},\\ D_1y^{s+1}+d_1+x^sQ+vB_1=0,\quad v\geq 0,\end{array}\right\} \tag{$\mathcal{LP}(x^s,y^s)$}$$

where $(x^s, y^s, v^s) \in D$ is a feasible point in Problem $(\mathcal{DC}(\sigma))$. Such local search methods show their efficiency in optimization problems with bilinear structure [Orlov et al., 2016, Orlov & Strekalovsky, 2016, Strekalovsky & Orlov, 2007, Strekalovsky et al., 2010, Strekalovsky, 2014]. Note that the accuracy for auxiliary problems is $\rho_s = 10^{-7}$. The accuracy of the local search is $\tau_k = 10^{-5}$.

The key element of the above GSA consists in constructing an approximation of the level surface of the convex function $h(\cdot)$, which generates the basic nonconvexity in the problem under consideration. For Problem $(\mathcal{DC}(\sigma))$ the approximation $\mathcal{A}_k = \mathcal{A}(\zeta_k)$ has been constructed with the help of a special set of directions $Dir = \{((x,y)+e^l, v+e^j),\ l = 1,...,m+n,\ j = 1,...,q\}$, where $e^l \in I\!R^{m+n},\ e^j \in I\!R^q$ are the Euclidean basis vectors of the corresponding dimension, (x, y, v) is a current critical point.

Dir is the standard direction set for problems with a bilinear structure according to our previous experience [Orlov & Strekalovsky, 2005, Orlov, 2008, Orlov et al., 2016, Strekalovsky et al., 2010]. Unfortunately, we cannot theoretically guarantee the global optimality of the point generated by the GSA with the set Dir. But numerically we obtain global solutions in most cases. In addition, we apply a special technique for reducing the approximation, because the number of points in the original set is rather large (especially when the dimension of the problem grows).

Consider an arbitrary set of directions with $q(m + n)$ points:

$$Dir_0 = \{(\bar{z}^i, \bar{w}^j) \mid (\bar{z}^i, \bar{w}^j) \neq 0,\ i = 1,...,m+n,\ j = 1,...,q\}.$$

In the matrices A_1 and B_1, find rows and columns with the maximum sum of their elements. Denote them as i_A, i_B and j_A, j_B, respectively. Then the new direction set will have the following form:

$$Cut(Dir_0) = \{(\bar{z}^{i_A}, \bar{w}^j),\ (\bar{z}^{i_B}, \bar{w}^j),\ j = 1,...,q; \quad (\bar{z}^i, \bar{w}^{j_A}),\ (\bar{z}^i, \bar{w}^{j_B}),\ i = 1,...,m+n\}.$$

The number of points in the approximation based on the set $Cut(Dir_0)$ is equal to $2(q + m + n)$. Therefore, as the dimension of the problem grows, it increases slower than the number of points in the approximation based on the set Dir_0. Note that here we use the matrices A_1 and B_1, because they are included in the definition of the function $h(\cdot)$ which generates the basic nonconvexity of the problem in question.

We employ the following technique to construct approximation points $\mathcal{A}_k = \mathcal{A}(\zeta_k)$ on the basis of the direction set: the triples (z^i, u^i, w^i) are found in the form $(z^i, u^i, w^i) = \lambda_i(\bar{z}^i, \bar{u}^i, \bar{w}^i),\ i = 1,...,N$, where $\lambda_i \in I\!R$ are computed using the condition $h(\lambda_i(\bar{z}^i, \bar{u}^i, \bar{w}^i)) = \xi - \zeta_k$. In that case the search of λ_i can be performed analytically (see also [Orlov & Strekalovsky, 2005, Orlov, 2008, Orlov et al., 2016, Strekalovsky et al., 2010]).

Further note that the selection of the algorithm parameters M and η can be carried out on the basis of our previous experience in solving problems with a bilinear structure [Orlov & Strekalovsky, 2005, Orlov, 2008, Orlov et al., 2016, Strekalovsky et al., 2010]. The parameter η is responsible for the accuracy of the inequality (2) from the GOCs (in order to diminish the computer rounding errors) [Orlov & Strekalovsky, 2005, Orlov, 2008, Orlov et al., 2016, Strekalovsky et al., 2010] (Step 3). Different values of the parameter M are responsible for splitting the interval $[\xi_-, \xi_+]$ into a suitable number of parts to realize a passive one-dimensional search along ξ. Here we use the following sets: 1) $M = 2$, $\eta = 0.0$; 2) $M = 5$, $\eta = 0.02$; 3) $M = 33$, $\eta = 0.1$. If it is required that an approximation to the global solution to Problem $(\mathcal{DC}(\sigma))$ be found rapidly, we may use option 1). With increase of M and η (options 2) and 3)), the algorithm gains in precision but loses in performance rate.

To compute the segment $[\xi_-, \xi_+]$ for one-dimensional search according to the GOCs, we need to solve two problems: on the minimum and maximum of a convex quadratic function $g(\cdot)$. The minimum problem can be solved by any quadratic programming method and an appropriate software subroutine. By the way, the same is true for the linearized problem $(\mathcal{PL}_i)$ at Step 4 ($\delta_k = 10^{-5}$). To tackle the maximum problem, we can employ a known global search strategy for convex maximization problems [Strekalovsky, 2003]. But in this case the computational process does not require an exact knowledge of these bounds. It is sufficient to have comparatively rough estimates [Strekalovsky, 2003]. Therefore, here we use: $\xi_- := 0.0;\ \ \xi_+ := (m + n + l) * \sigma$.

Finally, Steps 6-9 represent verification of the main inequality (3) from the GOCs, the stopping criteria, and looping.

5 Test Problem Generation Method and Numerical Computations

One of the most important issues in the testing of new numerical methods is the selection or construction of test cases. In the present work, we use the method for generation of bilevel test cases proposed in [Calamai & Vicente, 1994]. The idea of such generation is based on constructing bilevel problems of an arbitrary dimension with the help of the so-called kernel problems, which are one-dimensional bilevel problems with known local and global solutions (see also [Orlov, 2008, Strekalovsky et al., 2010]).

In generation of quadratic bilevel test problems of the type ($\mathcal{QBP}$), we used the kernel problems of the form

$$\left.\begin{array}{c} F(x,y) = \frac{1}{2}x^2 - x + \frac{1}{2}y^2 \downarrow \min\limits_{x,y}, \\ x \in X = \{x \in I\!R^1 \mid x \geq 0\}, \quad y \in Y_*(x) = \mathrm{Arg}\min\limits_{y}\{\frac{1}{2}y^2 - xy \mid y \in Y(x)\}, \\ Y(x) = \{y \in I\!R^1 \mid x - y \leq 1,\ x + y \leq \nu,\ -x-y \leq -1\}, \end{array}\right\} \tag{4}$$

where ν is a parameter whose value affects the number of local and global solutions in the problem.

In particular, according to [Calamai & Vicente, 1994], we can prove that: 1) if $\nu = 1$, then the optimal value of the problem (4) $F_* = -0.5$, and $(x^*, y^*) = (1, 0)$; 2) if $\nu = 1.5$, then $F_* = -0.4375$, and $(x^*, y^*) = (1.25, 0.25)$; 3) if $\nu = 2$ then $F_* = -0.25$, $(x^*, y^*) = (0.5, 0.5)$ or $(x^*, y^*) = (1.5, 0.5)$; 4) if $\nu = 3$ then $F_* = -0.25$, $(x^*, y^*) = (0.5, 0.5)$. So we separate 4 classes of kernel problems (4).

Then, in accordance with the generation scheme, an arbitrary number of kernel problems of various classes are united into the problem

$$\left.\begin{array}{c} F(x,y) = \sum\limits_{i=1}^{r}(\frac{1}{2}x_i^2 - x_i + \frac{1}{2}y_i^2) \downarrow \min\limits_{x,y}, \quad x \in X = \{x \in I\!R^m \mid x_i \geq 0,\ \ i = 1, ..., r\}, \\ y \in Y_*(x) = \mathrm{Arg}\min\limits_{y}\{\sum\limits_{i=1}^{r}(\frac{1}{2}y_i^2 - x_i y_i) \mid y \in Y(x)\}, \\ Y(x) = \{y \in I\!R^n \mid x_i - y_i \leq 1,\ x_i + y_i \leq \nu_i,\ \ -x_i - y_i \leq -1\}, \end{array}\right\} \tag{5}$$

where $\nu_i \in \{1;\ 1.5;\ 2;\ 3\}, \quad i = 1, ..., r$.

Let $cl1$, $cl2$, $cl3$, $cl4$ be the number of kernel problems of each class included into the "big" problem. Then the dimension of the "big" problem will be $m = n = p = cl1 + cl2 + cl3 + cl4$; $\;q = 3 * (cl1 + cl2 + cl3 + cl4)$.

Note that we can compute the number local and global solutions to the "big" problem.

Proposition 1 [Calamai & Vicente, 1994]. Problem (5) has 2^{cl3} global solutions, and it has additional local solutions only when $cl2 + cl4 > 0$. In this case the number of additional local minima to problem (5) equals $2^{cl2+cl3+cl4} - 2^{cl3}$.

Let us now describe the numerical testing of the global search method on series of problems generated by the method described above. The software that implements the method developed was coded in MATLAB 7.11.0.584 R2010b. To run the software, we used the computer with Intel Core i5-2400 processor (3.1 GHz) and 4Gb RAM. In total, 333 problems of dimension from 2 up to 200 were generated and solved.

The most interesting and typical results are presented in Table 1 with the following denotations: $NN = m = n = p$ is the dimension of the problem; M/ξ is the number of the set of parameters by which we can find a solution; GIt is the number of iterations of the global search method; Loc stands for the number of start-ups of the local search procedure performed to find the approximate global solution to the problem; LP, QP are the numbers of the LP and QP problems solved, respectively; T is the operating time of the program (in seconds); $Glob/Loc$ are the numbers of global solutions in the "big" problem and local solutions which are not global, respectively.

First of all, note that all generated problems were solved with the prescribed accuracy $\varepsilon = 10^{-3}$. Further, pay attention to a huge number of global and local solutions in some cases. It is interesting that the hardness of a problem does not depend on these values directly. Here we can see easy problems which are solved at the local search stage. And we can also see hard problems which take more than one hour to find a global solution. We propose a hypothesis that the hardness of problems depends on the set of classes of kernel problems in the "big" problem. At the end of the paper we present an overall statistics about complexity of classes combinations in the "big" problem.

Let us introduce a special complexity coefficient (CC): $CC = (Loc/NN)/Cnt$, where Loc is the number of the local search procedures for a given classes combination; $NN = m = n = p$ is the dimension of the problem with the given classes combination; Cnt is the total number of problems solved with the given classes combination.

The diagram about values of CC in different cases is presented in Figure 1, where the denotation 0x0x, for example, means that in the "big" problem we use kernel problems of classes 2 and 4; at the same time, the denotation 00x0 means that in the "big" problem we use the kernel problems of class 3 only etc.

Table 1: Global search in generated problems

NN	**cl1**	**cl2**	**cl3**	**cl4**	**M/ξ**	**GIt**	**Loc**	**LP**	**QP**	**T**	**Glob/Loc**
100	0	0	0	100	-	-	1	2	2	0.2	$1/1.2677 \cdot 10^{30}$
75	0	0	75	0	2	1	2998	5462	8460	70.5	$3.7779 \cdot 10^{22}/0$
100	0	0	100	0	-	-	1	2	2	0.1	$1.2677 \cdot 10^{30}/0$
100	0	0	5	95	-	-	1	2	2	0.1	$32/1.2677 \cdot 10^{30}$
200	0	0	100	100	2	1	7195	14390	21585	490	$3.8686 \cdot 10^{25}/1.4966 \cdot 10^{51}$
10	0	10	0	0	2	19	368	552	920	4.4	$1/1023$
6	0	5	0	1	1	20	100	139	239	1.2	$1/63$
50	0	25	25	0	2	60	5854	8998	14852	120	$33555532/1.1259 \cdot 10^{15}$
30	0	10	10	10	2	35	1442	2660	4102	24.5	$1024/1.0737 \cdot 10^{9}$
60	0	20	20	20	3	161	170095	271481	441576	3600	$1048576/1.1529 \cdot 10^{18}$
150	150	0	0	0	1	2	1503	1505	3008	47.6	$1/0$
175	5	0	0	170	2	4	7005	14141	21146	418	$1/1.4966 \cdot 10^{51}$
100	50	0	50	0	2	19	3219	4065	7284	84.7	$1.1259 \cdot 10^{15}/0$
100	40	0	30	30	1	4	1005	2445	3450	40.7	$1.0737 \cdot 10^{9}/1.1529 \cdot 10^{18}$
120	20	100	0	0	1	4	1205	2001	3206	43.2	$1/1.2677 \cdot 10^{30}$
150	50	50	0	50	1	3	1504	3079	4583	74.2	$1/1.2677 \cdot 10^{30}$
100	30	30	40	0	2	38	3061	4051	7112	89.3	$1.0995 \cdot 10^{12}/1.1806 \cdot 10^{21}$
20	5	5	5	5	1	5	206	412	618	3.6	$32/32736$
50	5	20	20	5	3	230	205849	315978	521827	4143	$1048576/3.5184 \cdot 10^{13}$
100	5	5	45	45	2	4	4009	10423	14432	155	$3.5184 \cdot 10^{13}/3.9614 \cdot 10^{28}$
100	5	45	45	5	1	252	427477	657020	1084497	12094	$3.5184 \cdot 10^{13}/3.9614 \cdot 10^{28}$
130	5	50	25	50	2	55	10834	18084	28918	680	$33554432/4.2535 \cdot 10^{37}$
200	2	194	2	2	2	59	18858	28532	47390	1335	$4/4.0173 \cdot 10^{59}$

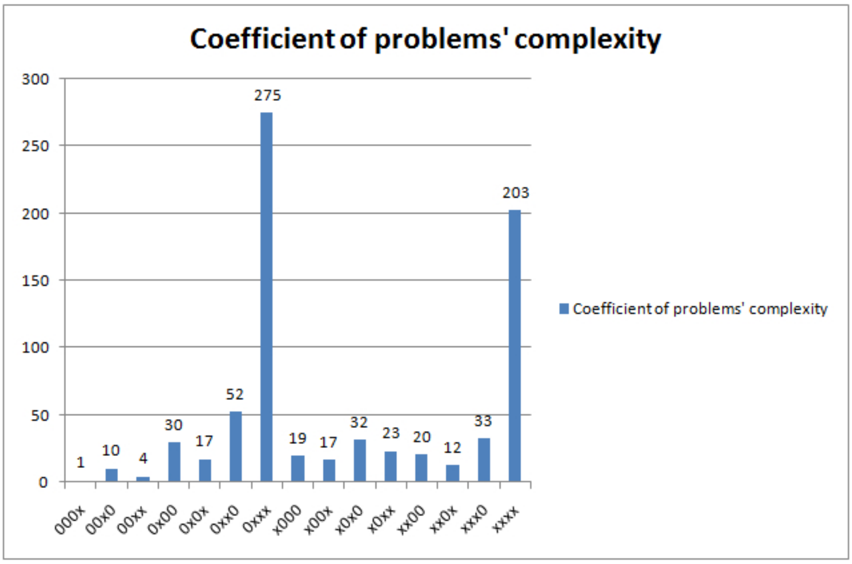

Figure 1: Complexity diagram of classes combinations

So, we can conclude that the hardest problems either consist of all four classes of kernel problems or comprise 2, 3, and 4 classes. This conclusion will be used in our future works when we will address more complicated test problems with a special random transformation (see [Calamai & Vicente, 1994, Orlov, 2008, Strekalovsky et al., 2010]).

Acknowledgements

This work has been supported by the Russian Science Foundation (Project no. 15-11-20015).

References

[Calamai & Vicente, 1994] Calamai, P. & Vicente, L. Generating Quadratic Bilevel Programming Test Problems. *ACM Transaction on Mathematical Software, 20*, 103-119. doi:10.1145/174603.174411.

[Dempe, 2002] Dempe, S. *Foundations of Bilevel Programming.* Dordrecht: Kluwer Academic Publishers.

[Orlov & Strekalovsky, 2005] Orlov, A.V. & Strekalovsky, A.S. Numerical search for equilibria in bimatrix games. *Computational Mathematics and Mathematical Physics, 45*, 947-960.

[Orlov, 2008] Orlov, A.V. Numerical solution of bilinear programming problems. *Computational Mathematics and Mathematical Physics, 48*, 225-241.

[Orlov et al., 2016] Orlov, A.V., Strekalovsky, A.S. & Batbileg, S. On computational search for Nash equilibrium in hexamatrix games. *Optimization Letters, 10(2)*, 369-381. doi:10.1007/s11590-014-0833-8.

[Orlov & Strekalovsky, 2016] Orlov A.V. & Strekalovsky A.S. On a Local Search for Hexamatrix Games. In A. Kononov, I. Bykadorov, O. Khamisov, I. Davydov, & P. Kononova (Eds.), *CEUR Workshop Proceedings. DOOR-SUP 2016, 1623* (pp.477-488).

[Orlov, 2017] Orlov, A.V. A Nonconvex Optimization Approach to Quadratic Bilevel Problems. In: R. Battiti, D. Kvasov, Y. Sergeyev (eds.) *"Learning and Intelligent Optimization Conference LION11*, Nizhny Novgorod, Russia", LNCS 10556, Springer. To appear.

[Pang, 2010] Pang, J.-S. (2010). Three modeling paradigms in mathematical programming. *Mathematical Programming. Series B., 125*, 297-323. doi:10.1007/s10107-010-0395-1.

[Strekalovsky, 2003] Strekalovsky, A.S. *Elements of nonconvex optimization* [in Russian]. Novosibirsk: Nauka.

[Strekalovsky & Orlov, 2007] Strekalovsky, A.S. & Orlov, A.V. *Bimatrix games and bilinear programming* [in Russian]. Moscow: FIZMATLIT.

[Strekalovsky et al., 2010] Strekalovsky, A.S., Orlov, A.V. & Malyshev, A.V. On computational search for optimistic solution in bilevel problems. *Journal of Global Optimization, 48*, 159-172. doi:10.1007/s10898-009-9514-z

[Strekalovsky, 2014] Strekalovsky, A.S. On solving optimization problems with hidden nonconvex structures. In T.M. Rassias, C.A. Floudas, & S. Butenko (Eds.), *Optimization in science and engineering* (pp. 465-502). New York: Springer.