

# Decomposition Algorithms for Mathematical Programming and Generalization of the Dantzig-Wolfe Method

Nikolai Oskorbin and Dmitriy Khvalynskiy<sup>(✉)</sup>

Altai State University, Lenina Prospect 61, 656049 Barnaul, Russia  
hvalynskij@email.asu.ru

**Abstract.** The main result of the paper is the applicability of Dantzig-Wolfe method for Large-Scale Nonlinear Programming with composite (block) structure of the function and constraints. Equivalent transformation of this problem is a task decomposition and coordination. This result allows to propose a new class of decomposition methods, which differ in approximating a feasible solution set of the coordination problem.

The authors propose the modification of the Dantzig-Wolfe algorithm for solving mathematical programming problems, where coordinating solutions is a convex set. It was applied as a decomposition algorithm for a quadratic programming problem. The algorithm was implemented in MS Excel environment and its efficiency was studied and tested. The rate of convergence in the number of global iterations was defined in tests, and it is shown that the proposed algorithm is not significantly different from the Dantzig-Wolfe algorithm in linear block programming.

**Keywords:** Nonlinear Large-Scale optimization · Coordination problems · Dantzig-Wolfe decomposition · Decomposition algorithm for quadratic programming

## 1 Introduction

The paper discusses Large-Scale optimization problems with the number of decision variables and the number of constraints. Such problems arise, for example, in finding optimal solutions in complex systems (production, economic and social). It is discussed in the papers [1–3].

There are two methodologies of Large-Scale problem solving: parallel computing based on mathematical programming algorithms and problem decomposition at the stage of defining of numerical algorithms. The last method is developed within the framework of block programming and is examined in this work.

Theoretical and applied methods of block programming is not limited only to development of efficient computing schemes and algorithms for Large-Scale optimization. These methods allow to study of information systems and the processes of coordination in complex systems [3]. In a general sense, algorithmic computation may be an example of public hierarchical system with distributed decision-making.

Lower elements play a leading role in making decisions in industrial practice. These elements are also known as “blocks” in mathematical researches [2].

Coordination Centre provides decisions on resources and production constraints. However, the main role of the Centre is to ensure the optimality of block solutions to serve the target function well.

The hierarchical system is driven by information flows between the elements. The volume of flows is substantially less than task dimensions.

To the present time has been widely used block methods of linear programming, in particular, the Dantzig-Wolfe Decomposition [1, 2, 4]. In [2, 4] it is shown that the linear programming problems to be solved by this method are a special case of the composite (block) tasks of nonlinear programming problem. This fact can be used to solve convex mathematical Large-Scale programming problems. In this paper, this method is used to build hierarchical algorithms for solving quadratic programming problems in which the objective function is convex and the constraints coincide with the linear programming problems.

## 2 Methods

The method of Dantzig-Wolfe Decomposition is obtained and justified as a specialized case of the simplex method for solving Large-Scale linear programming problems with a block structure system of constraints.

In this section, we will show that this method is applicable for convex mathematical programming problems, where the objective function has a composite (block) structure and constraints.

Let's describe this class of Large-Scale problems [2].

Let  $Z_i$  be a set of function values  $z_i(x_i)$ .

We introduce the notation:

$$x = \{x_1, \dots, x_n\} \in X = \times \prod_{i=1}^n X_i; \quad z(x) = (z_1(x_1), \dots, z_n(x_n)) \in Z = \times \prod_{i=1}^n Z_i;$$

$$\tilde{X} = \{x \in X | g(z(x)) \in G\};$$

$$\tilde{Z} = \{z \in Z | g(z) \in G\},$$

where  $\times \prod_{i=1}^n X_i = X_1 \times \dots \times X_n$  &  $\times \prod_{i=1}^n Z_i = Z_1 \times \dots \times Z_n$  are the Cartesian product of the sets  $X_1, \dots, X_n$  and  $Z_1, \dots, Z_n$ , respectively.

Write n-block optimization problem:

$$\max \{f(z(x)) | x \in \tilde{X}\} \quad (1)$$

We introduce a system of tasks.

The task of the center:

$$\max\{f(z)|z \in \tilde{Z}\} \quad (2)$$

The objective of the unit  $i, i = 1, \dots, n$  ::

$$\delta_i = \min \{ \|z_i^* - z_i(x_i)\| | x_i \in X_i \} = 0 \quad (3)$$

where  $z^* = (z_1^*, \dots, z_n^*)$  is one of the solutions to problem (2).

Let's show that problems (1) and (2)–(3) are equivalent.

The vector  $x^* = (x_1^*, \dots, x_n^*)$  is obtained from solving all  $n$  problems (3). Let's call it the solution of the problem (2)–(3).

Thus, the vector  $z$  is considered to be an auxiliary vector.

Two extreme problems are equivalent if the set of solutions coincides. Suppose that there is the solution of problems.

For the problem (3) let  $z_i^* \in Z_i$ . Then for any  $z_i^* \in Z_i$  we get  $\delta_i = 0$ , i.e. there is true equality

$$z_i^* = z_i(x_i^*) \quad (4)$$

where  $x_i^*$  is the solution of the problem (3).

We will show that any solution of problem (1) is the solution of the system (2)–(3).

Let  $x^*$  be a solution of problem (1), i.e.  $x^* \in \tilde{X}$ ;  $f(z(x^*)) \geq f(z(x)) \quad \forall x \in \tilde{X}$ . Examine the vector  $z^* = z(x^*)$ . For this vector, the fair inclusion  $z^* \in Z$  and relationships  $f(z^*) = f(z(x^*)) \geq f(z) = f(z(x)) \quad \forall x \in \tilde{X}$ .

The inequality is fulfilled when  $z \in \tilde{Z}$ , i.e.  $z^* \in \tilde{Z}$ ;  $f(z^*) \geq f(z) \quad \forall z \in \tilde{Z}$ . Therefore  $z^*$  is the solution of the problem (2). Since (4) is satisfied, the vector  $x^* = (x_1^*, \dots, x_n^*)$  is the solution of the problem (3).

We will show that any solution of the system (2)–(3) is the solution of problem (1). Let  $z^* = (z_1^*, \dots, z_n^*)$ ;  $x^* = (x_1^*, \dots, x_n^*)$  be solutions of problems (2) and (3) respectively. The condition (4) is satisfied. We have  $g(z^*) = g(z(x^*)) \in G$ ;  $x_i^* \in X_i, \quad i = 1, \dots, n$ ; whence it follows that  $x^* \in \tilde{X}$ . Next, a vector  $z^*$ , by definition, satisfies the condition  $f(z^*) \geq f(z) \quad \forall z \in \tilde{Z}$ . Given this expression and the definition  $\tilde{Z}$ , we have  $f(z^*) = f(z(x^*)) \geq f(z(x)) \quad \forall x \in \tilde{X}$ . Thus,  $x^*$  is the solution of the problem (1).

It is easy to show that if the problem (1) has no solution, there is no solution in (2), and vice versa. In addition, the extreme values of the problem functions (1) and (2) coincide. So, we have proved the equivalence of problems (1) and (2)–(3). Therefore, it is proved that the system  $(n + 1)$  of extreme problems (2)–(3) is the block decomposition of extremal problem (1).

Problems (2), (3) can be solved with a set  $Z$  known in the task of center and, thus, all hierarchical algorithms have to find  $Z$  at the initial stage, or a subset  $\tilde{Z}^* \subset Z$  that contains the optimal solution or, finally, the point  $z^*$ . Here it is assumed that  $\tilde{Z}^* \cap \text{Arg max}\{f(z)|z \in Z; g(z) \in G\} \neq \phi$ .

Thus, the set of all algorithms may be classified in accordance with the following:

- (1) It restores a set  $Z$  as a whole.
- (2) It restores a significant part  $\bar{Z}^*$  of the set  $Z$ .
- (3) It builds a maximizing sequence  $\{z^k\}_{k=1}^\infty$ , converging to  $z^*$ .

Let's consider below the 2nd way of building hierarchical algorithms in the case of inner approximation of the set  $Z$ . In the case of a convex set  $Z$ , it is similar to Dantzig-Wolfe Decomposition. In a specific example, we consider the example of the algorithm for solving the quadratic programming problem.

### 3 Results

Consider the second way of building hierarchical algorithms in the case of inner approximation of the set  $Z$

Let's write it in the following form:

Step 1. (starting the algorithm). Set  $Z_i^1$  with a defined condition  $Z_i^1 \subseteq Z_i$ ;  $\prod_{i=1}^n Z_i^1 = Z^1 \cap \{z | g(z) \in G\} \neq \emptyset$ . Believe  $k = 1$ .

Step 2. Solve the problem of the Centre in the formulation (2). Find  $z^k$  and the possible ways of checking the optimality of this solution for the set  $Z$ .

Step 3. Check the optimality condition of the point  $z^k$  on the set  $Z$ . If the point  $z^k$  is optimal on the set  $Z$ , go to step 5. Otherwise, find the point  $\tilde{z}^k$ , at which the optimality conditions on  $Z$  are violated.

Step 4. Go to set  $Z^{k+1}$ , apply  $k = k + 1$  and go to step 2.

Step 5 (finding solutions). For  $z_i^k$  solve the problem (3) with  $\delta_i^k = 0$ , because  $z_i^k \in Z_i$ .

For step 3 of the algorithm, consider a general nonlinear programming problem:

$$\max \{f(z) | g_j(z) \geq 0, j = \overline{1, m_1}; g_j(z) = 0, j = \overline{m_1 + 1, m}; z \in Z\} \quad (5)$$

where  $f, g_j(z)$  is a continuously differentiable functions;  $Z$  - convex closed set containing interior points.

It is known the following property of problem (5), named as a generalized rule of Lagrange multipliers [5]. Let  $z^0$  - the solution of problem (5). Suppose that the vectors  $g_j^i(z^0)$   $j \in \{m_1 + 1, \dots, m\} \cup \{i = 1, 2, \dots, m_1 | g_i(z_0) = 0\}$  are linearly independent. Then there exist non-zero multipliers  $\lambda_j^0$  ( $j = 0, 1, \dots, m$ ), such that  $\lambda_j^0 \geq 0$ ,  $j = 0, 1, \dots, m_1$ ;  $\lambda_j^0 \cdot g_j(z^0) = 0, j = 1, \dots, m$ , and for all  $z \in Z$  is fulfilled  $(\lambda_0 f'(z^0) + \sum_{j=1}^m \lambda_j^0 g_j'(z^0), z - z^0) \leq 0$ .

Let discuss the use of this result for block quadratic programming. In step 4 of the algorithm, it is possible to apply the approach of method Dantzig-Wolfe Decomposition to construct a sequence of sets  $\bar{Z}^k$  of convex linear combinations  $k$  of the known points from the set  $Z$ .

Consider the mathematical model of production planning  $T$  industrial enterprises in the following form:

$$\max_{x \in X} \left\{ \sum_{t=1}^T -(P_t^H - p_t x_t)^2 \mid x_t \in X_t, t = 1, \dots, T; \sum_{t=1}^T \bar{A} x_t \leq B \right\} \quad (6)$$

where  $P_t^H$  - the profit potential, and  $X_t$  - a valid set of unit plans  $t$ :  $X_t = \{x_t \in R^n \mid A_t x_t \leq B_t; x_t \geq 0\}$ .

The matrix in the above expressions have the following dimensions:  $p_t - (1 \times n_t)$ ;  $x_t - (n_t \times 1)$ ;  $\bar{A}_t - (m \times n_t)$ ;  $B - (m \times 1)$ ;  $A_t - (m_t \times n_t)$ ;  $B_t - (m_t \times 1)$ , where the first number indicates the number of rows, the second - the number of columns.

The economic meaning of the planning model is to optimize production plans of enterprises, where the potential possibilities for profit are maximally implemented.

Let's introduce the units  $P_t = p_t x_t$ ,  $S_t = \bar{A}_t x_t$  for problem (6) and the set  $Z_t$  of possible values for these units. Then problem (2) in this case has the form:

$$\max_{(P,S)} \left\{ \sum_{t=1}^T -(P_t^H - P_t)^2 \mid (P_t, S_t) \in Z_t, t = 1, \dots, T; \sum_{t=1}^T S_t \leq B \right\} \quad (7)$$

Let is found set  $Z_t^k \subseteq Z_t$  on iteration  $k$  of the algorithm. By solving the problem (7) for these sets, there is obtained points  $z_t^k \in Z_t$  and the dual variables  $u^k$ . According to (5) the point is optimal for the corresponding block problem (7), when for all  $z_t \in Z_t$  fulfilled the inequality:

$$2(P_t^H - P_t^k)(P_t - P_t^k) + u^k(S_t - S_t^k) \leq 0 \quad (8)$$

If inequality (8) is fulfilled for all  $t = 1, \dots, T$ , then the solution of problem (7) is found.

Let denote  $\alpha_t^k = 2(P_t^H - P_t^k)$ ;  $\gamma_t^k = \alpha_t^k P_t^k$ ;  $\beta_t^k = u^k S_t^k$ . Then, during the transition to the original variables, the inequality (8) for  $t$ -block is equivalent to the following inequality:

$$\delta_t^k = \left[ \max_{x_t \in X_t} (\alpha_t^k p_t + u^k \bar{A}_t) x_t - (\gamma_t^k + \beta_t^k) \right] \leq 0 \quad (9)$$

Thus, for the nonlinear problem (6) we found the indicator of optimality of the current solution (step 3 of the algorithm). It can be checked with a solution of separate task blocks. There is a finite speed of convergence of the solution to problem (7) for precise calculations.

Finite convergence of such block programming algorithms for accurately calculating is proved for example in [4]. According to (9) at each global iteration, the algorithm has to solve a linear programming problem on a polyhedral set with a limited number of corner points.

The study of nonlinear analog Dantzig-Wolfe method and its implementation in a hierarchical algorithm was conducted in Excel for an enterprise production planning system (6) with 5 companies ( $T = 5$ ). All calculations of the lower level (the information of companies) was carried out in Excel. The numerical values provided with the matrix (6) were filled with uniformly-distributed pseudo-random numbers.

The marginal profit of the companies ( $P_i^H$ ) was defined as the maximum, without the resource constraints, by the decision of local linear programming problems. Exact solution of the problem (6) was obtained from a solver of Excel.

The formation of the initial coordinating task (7) performed by the classical scheme of the Danzig Wolfe method. For that, we selected the 2T columns corresponding to zero and to maximum plans. Additional columns of constraints were formed with using the expression (9). Each global iteration and the corresponding approximation of coordination tasks were performed in separate Excel worksheets.

The information exchange between the Centre and agents included current and optimal values of resource units, i.e. values of variables  $P_i^k$ ;  $S_i^k$ ;  $u^k$ . We show the need to use performance structure of the optimal plan (Step 5 of the algorithm) because a price coordination (9) does not allow finding the solution of the problem (6). This effect is similar to the classical Dantzig Wolfe method, but in our case it has decentralized decisions, i.e. tasks (3) do not require of knowledge Centre for variables of lower levels. The proposed algorithm has obtained exact solutions for all considered numerical tasks (6). So the problem (6) with 12 variables and 24 constraints of 2 resources had error estimates for 4 initial iterations of functions 9,66%; 1,28%; 0,06%; 0,00% respectively.

## 4 Discussions

The main result of this paper is to prove the applicability of the Dantzig-Wolfe method for solving not only linear mathematical programming Large-Scale problems. It is proposed the modification of the Dantzig-Wolfe algorithm for solving convex mathematical programming problems with block (composite) structure in the function and constraints.

The algorithm for quadratic programming was implemented in MS Excel environment and its efficiency was studied. The rate of convergence in the number of global iterations was defined in tests, and it is shown that the proposed algorithm is not significantly different for this indicator from the Dantzig-Wolfe algorithm in linear block programming.

## References

1. Tsurkov, V.I.: Decomposition in Large-Scale Problems. Monography, Nauka, Moscow (1981)
2. Mamchenko, O.P., Oskorbin, N.M.: Modeling Hierarchical Systems: Textbook for Higher Education Institutions. Publishing House of ASU, Barnaul (2007)

3. Oskorbin, N.M., Bogovis, A.V., Zharikov, A.V.: Informational processes coordination of enterprise solutions and computer modeling. *Vestn. Novosib. State Univ. Ser. Inf. Technol.* **8**(1), 54–59 (2010)
4. Oskorbin, N.M.: On the circuits of block programming. *Econ. Math. Methods* **5**, 964–972 (1981)
5. Moiseev, N.N., Ivanilov, Y.P., Stolyarova, E.M.: *Methods of optimization*. Monography, Nauka, Moscow (1978)