POLSKA AKADEMIA NAUK
INSTYTUT BADAŃ SYSTEMOWYCH

PROCEEDINGS OF THE 3rd ITALIAN-POLISH CONFERENCE ON APPLICATIONS OF SYSTEMS THEORY TO ECONOMY, MANAGEMENT AND TECHNOLOGY

WARSZAWA 1977

Redaktor techniczny
Iwona Dobrzyńska

Korekta
Halina Wołyniec

POLSKA AKADEMIA NAUK
INSTYTUT BADAŃ SYSTEMOWYCH

PROCEEDINGS OF THE 3rd ITALIAN-POLISH CONFERENCE ON APPLICATIONS OF SYSTEMS THEORY TO ECONOMY, MANAGEMENT AND TECHNOLOGY

BIAŁOWIEŻA, POLAND
MAY 26—31, 1976

EDITED BY J. GUTENBAUM

WARSZAWA 1977

Redaktor techniczny
Iwona Dobrzyńska

Korekta
Halina Wołyniec

PREFACE

The present volume comprises the papers from the 3rd Italian-Polish Conference which was held in [illegible] from [illegible] 20 to 25. The conference organized by the Centre of Control Systems and Computation of the Italian National Research Council in Rome and the Institute for Organization, Management and Control Sciences of Polish Academy of Sciences and the Ministry of Science, Higher Education and Technology in Warsaw was the third conference of this type.

The first one took place in Kraków (Poland) in 1972, the second in Pugnochiuso in 1974.

This conference, like both the previous ones, gathered the leading representatives of many scientific centres from Italy as well as from Poland, representing also the different aspects of the systems analysis applied to the economic, management, data processing and technological systems.

From the past, such meetings permit the exchange of views and experiences between the specialists from both countries what is [illegible] to both sides. Hence the natural part, the [illegible] of the systems and control theory methods in the [illegible] constructions, the essential development of the quantitative analysis and the particularly great utility value of large computing systems.

The meaning of such meetings however goes far beyond that, what is happening during the conference because the established contacts are also fruitful in form of common scientific research works and — "last but not least" — begin the friendly links between the scientists of both countries which last for years.

The proceedings of the conference were divided into three parts:

1. Optimization and Control Theory;
2. Systems Theory in Economics;
3. Technological, Managerial and Information Systems.

While the first two parts are of rather homogeneous character, the third part contains the papers concerning the different types of models — for the economic, technological, management and data processing systems.

A. P. Wierzbicki
Institute of Automatic Control, Technical University of Warsaw

A PRIMAL-DUAL LARGE SCALE OPTIMIZATION METHOD BASED ON AUGMENTED LAGRANGE FUNCTIONS AND INTERACTION SHIFT PREDICTION

1. VARIABLE METRIC ALGORITHMS FOR SADDLE-POINT DETERMINATION

A new class of saddle-point seeking and constrained optimization algorithms has been introduced in [1]. These algorithms combine many advantages and desired properties of various known optimization methods. They solve quadratic programming problems in a finite member of iterations, are directly applicable and effective for non-quadratic problems with nonlinear constraints, have a straightforward generalization to infinite-dimmensional (dynamic) optimization problems, and are single-loop iterative procedures, as opposed to many other multiplier or penalty-shift algorithms. They do not require the programming of second-order derivatives, nor the inversion of matrices, and are equally effective for linear and nonlinear constraints, as opposed to Newtonlike projection methods. These algorithms consist of double variable metric approximation for saddle-point seeking, applied to an augmented Lagrange function for constrained optimization. The main ideas of the saddle-point seeking algorithms are reviewed here.

Suppose there exists an unique saddle-point $(\hat{y}, \hat{v})$ of a function $\varphi : R^n \times \times R^m \to R^1$

$$(\hat{y}, \hat{v}) = \arg \max_{v \in R^m} \min_{y \in R^n} \varphi(y, v) = \arg \min_{y \in R^n} \sup_{v \in R^m} \varphi(y, v) \tag{1}$$

and let the function φ be twice differentiable (in a neighbourhood of $(\hat{y}, \hat{v})$) in both variables and posess a unique minimizer in y for each v; the function φ need not be nonlinear in v. The necessary conditions of the saddle-point

$$\varphi_y(\hat{y}, \hat{v}) = 0 \tag{2a}$$

$$\varphi_v(\hat{y}, \hat{v}) = 0 \tag{2b}$$

can be approximated by several Newton-like procedures. The basic one was used with several modifications by various authors and has the general form of the following **algorithm A1:**

$$v^{i+1} = v^i + (\varphi_{vy}\,\varphi_{yy}^{-1}\,\varphi_{yv} - \varphi_{vv})^{-1}(\varphi_v - \varphi_{vy}\,\varphi_{yy}^{-1}\,\varphi_y) \tag{3a}$$

$$y^{i+1} = y^i - \varphi_{yy}^{-1}(\varphi_y + \varphi_{vy}(v^{i+1} - v^i)) \tag{3b}$$

where all derivatives are evaluated at (y^i, v_i). This algorithm has the usual adventages and disadvantages of Newton-like procedures: itc onverges quadratically under appropriate assumptions, but only in a close neighbourhood of the solution $(\hat{y}, \hat{v})$, and requires the programming of second-order derivatives and matrix inversion.

In order to obtain a quasi-Newton, variable metric procedure it is useful to modify first the algorithm A1, allowing for more gradient computations and obtaining the following **algorithm A2:**

$$\tilde{d}_y^i = -\varphi_{yy}^{-1}\,\varphi_y(y^i, v^i) \tag{4a}$$

$$b_v^i = \varphi_v(y^i + \tilde{d}_y^i, v^i) \tag{4b}$$

$$v^{i+1} = v^i + (\varphi_{vy}\,\varphi_{yy}^{-1}\,\varphi_{yv} - \varphi_{vv})^{-1} b_v^i \tag{4c}$$

$$b_y^i = \varphi_y(y^i, v^{i+1}); \qquad d_y^i = -\varphi_{yy}^{-1} b_y^i \tag{4d}$$

$$y^{i+1} = y^i + \tau^i d_y^i \tag{4e}$$

where $\tau^i = \arg\min_{\tau\in(0,1)} \varphi(y^i + \tau d_y^i, v^{i+1})$, but the directional minimisation need not be very accurate; all second-roder derivatives e_{yy}, e_{yv}, e_{vv} are evaluated at (g^i, v^i). This algorithm requires actually more computations per iteration than the algorithm A1; but it has an interesting interpretation, useful for constructing variable metric algorithms. First, a Newton-type direction for changes of y to satisty (2a) is determined by (4a). Then the violation of the condition (2b) is **predicted** by (4a). Then the violation of the condition (2b) is **predicted** by (4b) and **compensated** by the changes of v determined from (4c). This allows the determination of the modified Newton-type direction (4d) and the directional search (4e) in y; the step-size coefficient τ_* converges to 1 in subsequent iterations.

The same algorithmic scheme can be applied when the second-order derivatives are not actually computed and inverted, but only approximated by a variable metric. Suppose the following relation holds for all y^i, y^{i+1}, v^{i+1} in a neighbourhood of $(\hat{y}, \hat{v})$:

$$r_y^i = A_y s_y^i; \qquad A_y = \varphi_{yy}(\hat{y}, \hat{v}) \tag{5a}$$

where

$$s_y^i = y^{i+1} - y^i; \quad r_y^i = \tilde{b}_y^{i+1} - b_y^i; \quad \tilde{b}_y^{i+1} = \varphi_y(y^{i+1}, v^{i+1}); \quad b_y^i = \varphi_y(y^i, v^{i+1}) \tag{5b}$$

Then it is possible to approximate A_y^{-1} by a variable metric V_y^{i+1} with help of the data $\{r_y^j, s_y^j\}_1^i$. The variable metric can have several forms — the known algorithm of Davidon, Fletcher and Powell [2], or the Fletcher-Convex algorithm can be used here. But the following rank-one formula with suitable well-conditioning checking is preferable, since it approximates A_y^{-1} independently of the step-size τ^i:

$$V_y^{i+1} = V_y^i + \alpha^i (s_y^i - V_y^i r_y^i > < s_y^i - V_y^i r_y^i) \tag{6a}$$

where

$$\alpha^i = \begin{cases} 0, & \text{if} \quad < s_y^i - V_y^i r_y^i, r_y^i > = 0 \\ & \text{or} \quad < s_y^i - V_y^i r_y^i, r_y^i > < 0 \quad \text{and} \quad < s_y^i - V_y^i r_y^i, b_y^i > < 0 \\ \dfrac{1}{< s_y^i - V_y^i r_y^i, r_y^i >} & \text{in other cases} \end{cases} \tag{6b}$$

Other definitions of α^i are also possible. The symbol $\langle\cdot, \cdot\rangle$ denotes here the scalar product and the symbol $\cdot\rangle\langle\cdot$ the outer product $(a\rangle\langle b(= a\langle b, y\rangle$ for all a, b, y in a Hilbert space, in this case R^n). If φ is quadratic in y and A_y is constant and strictly positive, then either $V^{i+1} = A_y^{-1}$ or y^{i+1} minimizes φ in y (or both) after at most n iterations.

The direction

$$d_y^i = -V_y^i b_y^i; \quad b_y^i = \varphi_y(y^i, v^{i+1}) \tag{7}$$

can be used in (4d) for the directional search (4e). However, the prediction (4a, b) and compensation (4c) of the violation of the necessary condition (2b) must be accordingly changed. It can be shown [1] that the compensating equation (4c) takes the form:

$$v^{i+1} = v_i^i + (\tau^i \varphi_{vy} V_y^i \varphi_{yv} - \varphi_{vv})^{-1} \varphi_v(y^i - \tau^i V_y^i \tilde{b}_y^i, v^i) \tag{8}$$

where $\tau^i = 1$ and the matrix $(\varphi_{vy} V_y^i \varphi_{yv} - \varphi_{vv})^{-1}$ can be approximated by another variable metric V_v^{i+1} with help of the data $\{r_v^i, s_v^i\}^i$, where

$$r_v^i = A_v s_v^i; \quad A_v = \tau^i \varphi_{vy} V_y^i \varphi_{yv} - \varphi_{vv} = \varphi_{vy} V_y^i \varphi_{yv} - \varphi_{vv} \tag{9a}$$

and

$$\begin{gathered} s_v^i = v^{i+1} - v^i; \quad r_v^i = \tilde{b}_v^i - \tilde{b}_v^{i+1} \approx b_v^i - \tilde{b}_v^{i+1}; \\ \tilde{b}_v^i = \varphi_v(y^i + \tau^i \tilde{d}_y^i, v^i); \quad b_v^i = \varphi_v(y^i + \tilde{d}_y^i, v^i); \quad \tilde{b}_v^{i+1} = \varphi_v(y^{i+1}, v^{i+1}) \end{gathered} \tag{9b}$$

The resulting variable metric procedure for saddle-point determination has the form of the following **algorithm A3:**

$$\tilde{b}_y^i = \varphi_y(y^i, v^i); \quad \tilde{b}_v^i = \varphi_v(y^i, v^i) \tag{10a}$$

$$(\text{if } i > 1) \quad s_y^i = y^i - y^{i-1}; \quad r_y^i = \tilde{b}_y^i - b_y^{i-1}; \quad V_y^i \text{ results from (6a,b)} \tag{10b}$$

$$s_v^i = v^i - v^{i-1}; \quad r_v^i = b_v^{i-1} - \tilde{b}_v^i; \quad V_v^i \text{ results from (6a,b)}$$

$$\tilde{d}_y^i = -V_y^i \tilde{b}_y^i; \quad b_v^i = \varphi_v(y^i + \tilde{d}_y^i, v^i); \quad v^{i+1} = v^i + V_v^i b_v^i \tag{10c}$$

$$b_y^i = \varphi_y^i(y^i, v^{i+1}); \quad d_y^i = -V_y^i b_y^i \tag{10d}$$

$$y^{i+1} = y^i + \tau^i d_y^i \tag{10e}$$

where $\tau^i \in (0; 1]$ results from an approximate directional minimisation. The step (10c) can be interpreted as the prediction and compensation of the violation of the necessary condition (2b). However, this prediction and compensation is fairly accurate first when V_y approsimates reasonably A^{-1} and, therefore, $_0{}^i$ is close to 1. Hence, the following modification of the algorithm is preferable — **algorithm** $A4$:

$$If \quad i \leqslant N, \quad \text{set} \quad v^i = v, \quad \tilde{b}_{yi} = \varphi_y(y^i, v) \tag{11a}$$

$$(if\ i > 1)\, s_y^i = y^i - y^{i-1}; \quad r_y^i = \tilde{b}_y^i - \tilde{b}_y^{i+1}; \quad V_y^i \quad \text{results from (6a,b)} \tag{11b}$$

$$\tilde{d}_y^i = -V_y^i b_y^i \tag{11c}$$

$$y^{i+1} = y^i + \tau^i \tilde{d}_y^i, \quad \text{where} \quad \tau^i \approx \arg\min_{\tau \in (0:1]} \varphi(y^i + \tau \tilde{d}_y^i, v) \tag{11d}$$

$$If\ i > N, \quad i \leqslant N+M \quad \text{go to} \quad A3(10a, \ldots c) \tag{11e}$$

$$\text{(Optional reset)}\ If\ i > N+M,\ \text{set}\ i = 1,\ V_y^i = V_y^1 = I_{nzn},\ V_v^i = V_v^1 = I_{mzm} \tag{11f}$$

The numbers of iterations N, M are chosen according to the nature of the problem. For not very large problems without distinctive structure, $N \geqslant n$, $M \geqslant m$ are preferable. For very large problems with dynamic or decomposable structure, much smaller numbers N, M can be chosen. Similarly, the starting variable metrics V_v^1 and V_y^1 can be chosen aceording to the information available and the unit matrices are assumed when no additional information is given.

The algorithm A4 requires less computational effort than the algorithm A3. For N interations, one gradient b_y^i per iteration is only computed, whereas in A3 four gradients b_y^i, b_y^i, b_v^i, b_v^i are required per iteration. Although these algorithms are quite new and not fully verified in practical computation, they are expected to be ones of the most powerful tools for solving saddle--point problems and optimization problems with consiraints. If the function φ is quadratic in y, bilinear in y, v and linear or quadratic in v, the algotirhms A3, A4 find the saddle-point in at most $n+m$ (or $N+m$) iterations. On the other hand, these algorithms are also disectly applicable to non-quadratic functions $\varphi : E \times F \to R^1$, where E and F are arbitrary Hilbert spaces.

In many applications, the function φ is linear in v, $\varphi_{vv} = 0$ and φ_{vy} is easy to determine computationally; this occurs, for example, when φ is normal or augmented Lagrange fuction for an optimization problem with equality constraints and v ist the corresponding Lagrange multiplier. Following a suggestion by Fletcher [3] for other multiplier methods, the algotithms A3, A4 can me modified in such a case to utilize the additional information. Namely, applying the Householder formula, the inverse $V^{-1}{}^{,i}$ can be computed parallely to V_y determined as in (6a, b). Since $\varphi_{vv}=0$ and φ_{vy} is known, the inverse $(\varphi_{vy} V_y^i \varphi_{yv})^{-1}$ required in the compensation equation (8), can be computed in each iteration. If φ_{vy} is constant, as for Lagrange functions for problems with linear equality constraints, then this inverse is also easy to compute by the Householder formula. This leads to the following **algorithm A5:**

$$\tilde{d}_y^i = \varphi_y(y^i, v^i) \tag{12a}$$

$$(If\ i>1)\ s_y^i = y^i - y^{i-1};\quad r_y^i = \tilde{b}_y^i - b_y^{i-1};\quad V_y^i \text{ results from (6a,b)}; \tag{12b}$$

$$V_v^i = (\varphi_{yy} V_y^i \varphi_{yv})^{-1} \text{ with all possible computational simplifications}$$

$$\tilde{d}_y^i = -V_y^i \tilde{b}_y^i;\quad b_o^i = \varphi_v(y^i + \tilde{d}_y^i,);\qquad v^{i+1} = v^i + V_v^i b_v^i \tag{12c}$$

$$b_y^i = \varphi_y(y^i, v^{i+1});\quad d_y^i = -V_y^i b_y^i \tag{12d}$$

$$y^{i+1} = y^i + \tau^i d_y^i,\quad \text{where}\quad \tau^i \approx \arg\min_{\tau\in(0:1]} \varphi(y^i + \tau d_y^i, v^{i+1}) \tag{12e}$$

and to the **algorithm A6** identical to A4 but for the step (11e) where "go to A5 (12a...12e)" is used. The algorithms A5, A6 can solve quadratic optimization problems with linear equality constraints in a smaller number of iterations (n) than the algorithms A3, A4 ($n+m$). But the computational effort per iteration is increased, particularly if φ_{vy} is not constant, and the algorithms A5, A6 are less general: their extension to optimization problems with inequality constraints or with a large number of constraints is more complicated.

2. A QUADRATIC PROGRAMMING PROBLEM

Consider the following optimization problem:

$$\hat{y} = \arg\min_{y\in Y_z} f(y);\quad f(y) = \frac{1}{2} y^* \mathscr{A} y + b_0^* y + C_0 \tag{13}$$

$$Y_z = \{y \in R^n : g(y) = \mathscr{B} y = z \in R^m\};\quad m \quad n$$

where star denotes transposition. The problem is strictly convex, if the matrix $\mathscr{A} : R^n \to R^n$ is strictly positive, $y^* \mathscr{A} y > 0$ for all $y \neq 0$, $y \in R^n$, and normal, if the matrix $\mathscr{B} : R^n \to R^m$ has its full rank. If these assumptions are satisfied, then the solution $\hat{y}$ of the problem exists and corresponds to the unique saddle-point of the normal Lagrange function:

$$L(\eta, y) = f(y) + \eta^*(g(y) - z) = \frac{1}{2} y^* \mathscr{A} y + (b_0^* + \eta^* \mathscr{B}) y + C_0 - \eta^* z \tag{14a}$$

$$(\hat{\eta}, \hat{y}) = \arg\max_{\eta \in R^m} \min_{y \in R^n} L(\eta, y) = \arg\min_{y \in R^n} \sup_{\eta \in R^m} L(\eta, y) \tag{14b}$$

To find this saddle-point, one can use the algorithm A3 or A4 with $v \sim \eta$ and

$$\varphi_y \sim L_y = \mathscr{A} y + b_0 + \mathscr{B}^* \eta \tag{15a}$$

$$\varphi_v \sim L_\eta = \mathscr{B} y - z \tag{15b}$$

where the prediction and compensation of the violation of $L\eta(\eta, y) = 0$ corresponds to the **prediction and compensation of the violation of constraints.** If $N = n$ is used in the algorithm A4, then the solution (η, y) is found after at most $n+m$ iterations.

If the matrix $\mathscr{B}$ has not its full rank, then the saddlepoint is not unique in v; this case shall not be considered i this paper. If the matrix $\mathscr{A}$ is not positive, then the saddle-point does not exist, though the problem (13) may still have a solution. It fact, a sufficient condition for the existence of the solution $\hat{y}$ is that the matrix $\mathscr{A}$ is strictly positive in the subopace $Y_0 = \{y \in R^n : \mathscr{B} y = 0\}$, $y^* \mathscr{A} y > 0$ for all $y \neq 0, y \in Y_0$. In this case, there exists a constant $\varrho_0 > 0$ such that for all $\varrho > \varrho_0$ the solution $\hat{y}$ corresponds to the unique saddle-point $(\hat{\vartheta}, \hat{y})$ of the augmented Lagrange function — see [4], [5], [6]:

$$\Lambda(\varrho, \vartheta, y) = f(y) + \frac{1}{2} \varrho \|g(y) - z + \vartheta\|^2 - \frac{1}{2} \varrho \|\vartheta\|^2 =$$

$$= \frac{1}{2} y^* (\mathscr{A} + \varrho \mathscr{B}^* \mathscr{B}) y + (b_0^* + \varrho \vartheta^* \mathscr{B}) y - \varrho \vartheta^* z - \frac{1}{2} \varrho \|z\|^2 + C_0 \tag{15a}$$

$$(\hat{\vartheta}, \hat{y}) = \arg\max_{\vartheta \in R^m} \min_{y \in R^n} \Lambda(\varrho, \vartheta, y) = \arg\min_{y \in R^n} \sup_{\vartheta \in R^m} \Lambda(\varrho, \vartheta, y) \tag{15b}$$

It should be noted that

$$\Psi(y, \varrho, \vartheta) = \Lambda(\varrho, \vartheta, y) + \frac{1}{2} \varrho \|\vartheta\|^2 = f(y) + \frac{1}{2} \varrho \|g(y) - z + \vartheta\|^2 \tag{16}$$

is a shifted penalty function as introduced in [7] and examined further in [8], [3] and by other authors. Therefore, the variable $\vartheta \in R^m$ has two interpretations: first, it is a Lagrange multiplier, $\vartheta = \frac{1}{\varrho} \eta$; secondly, it is a penalty shift and penalty shifting algorithms [7], [8] could be used for finding the saddle-point (15b). It can be concluded from both of these interpretations that the gradient $\Lambda\vartheta$ should be multiplied by a factor $\frac{1}{\varrho}$, if the algorithms A3, A4 are applied in order to find the saddle-point $(\hat{\vartheta}, \hat{y})$ with $\vartheta \sim v$ and

$$\varphi_y \sim \Lambda_y = (\mathscr{A} + \varrho\mathscr{B}^*\mathscr{B})\, y + b_0 + \varrho\mathscr{B}^*\vartheta \tag{17a}$$

$$\varphi_v \sim \frac{1}{\varrho}\Lambda_\vartheta = \mathscr{B}y - z \tag{17d}$$

Again, the prediction and compensation of the violation of $\Lambda_\vartheta(\varrho, \hat{\vartheta}, \hat{y}) = 0$ corresponds to the prediction and compensation of the violation of constraints. The compensation (10c) in the first (or $N+1$) iteration, with $V_v^1 = I_{mxm}$, is actually analogous to the penalty shift $\vartheta^{i+1} = \vartheta^i + (\mathscr{B}y - z)^i$ as introduced in [7]; however, it is supplemented with the prediction and the variable metric approximation of $A_v^{-1} \sim \frac{1}{\varrho}\left(\mathscr{B}(\mathscr{A} + \varrho\mathscr{B}^*\mathscr{B})^{-1}\mathscr{B}^*\right)^{-1}$ These improvements allow for the use of the algorithm A3 or A4 in a single-loop iterative procedure, whereas original penalty shift algorithms, though very effectivecomputationally, are double-loop iterative procedures and do not solve quadratic programming problems in a finite number of steps. The algorithms A3 or A4 (with $N=n$) do find the solution $(\hat{\vartheta}, \hat{y})$ in $n+m$ iterations, if ϱ is sufficiently large. If ϱ is not large enough then the minimization with respect to Y is disturbed; but this case can be recognized algorithmically and an automatic increase of ϱ in $n+m+1$ (or $N+M+1$) iteration can supplement the original algorithm.

Observe that the algorithm A4, when applied to a quadratic programming problem, theoretically does not make use of the prediction in (9c). After n iterations, the minimum of a Lagrange function in y for a given Lagrange multiplier is found, and $\tilde{d}_y^i = \varphi_y(y^i, v^i) = 0$, $\tilde{b}_y^i = 0$, $b_v^i = \tilde{d}_v^i = \varphi_v(y^i, v^i)$. Since the matrix $\mathscr{A}^{-1}$ or $(\mathscr{A} + \varrho\mathscr{B}^*\mathscr{B})^{-1}$ is determined by the variable metric V_y^{i+1} hence, after each change of $v \sim \eta$ or ϑ, the corresponding minimizing $\hat{y}_v$ is found in one iteration. But the prediction b_v^i, originally devised for non-quadratic situation, is also useful to suppress possible numerical errors in the quadratic case.

There are many other algorithms related to augmented Lagrangians, called generally multiplier algorithms — see [3], [9], [10], [11]. But these algorithms either do not solve a quadratic programming problem in a finite number of iterations, or do involve matrix inversions similar to Newton-type procedures or to gradient-projection techniques. Of course, a quadratic programming problem can be solved in a finite number of steps by linear programming techniques, gradient projection techniques or Newton-type methods, but each of these methods has disadvantages when generalized to nonlinear or higher-dimmensional problems, whereas the application of the algorithms A3, A4 to nonlinear optimization problems posed even in Hilbert space is straightforward.

3. A LARGE SCALE OPTIMIZATION PROBLEM

Cansider again the problem (12), but with the additional assumption that n and m are large and the matrices $\mathscr{A}$, $\mathscr{B}$ have a distinctive structure

$$\mathscr{A}=\begin{bmatrix} P & O & O \\ O & Q & O \\ O & O & R \end{bmatrix}; \quad y=\begin{bmatrix} x \\ u \\ w \end{bmatrix}; \quad \mathscr{B}=\begin{bmatrix} A & B & C \\ -MD & O & I \end{bmatrix}; \quad z=\begin{bmatrix} z_1 \\ O \end{bmatrix} \tag{18}$$

where the matrices P, Q, R, A, B, C, D are also block-diagonal and the matrix M consists only of zero and unit elements (though the latter do not occur on ist diagonal). Suppose that the matrix $\mathscr{B}$ has its maximal rank (in particular, the matrix $\mathscr{B}_1=[A\ B\ C]$ has its maximal rank), and that $\dim x=$ $=n_x=\dim z_1$, $\dim u=n_u$, $\dim w=n_w \ll n_x$, $n=n_x+n+n_w$, $m=n_x+n_w$. The problem (12) can be written as

$$(\hat{x}, \hat{u}, \hat{w}) = \underset{(x,u,w)\in Y_{z1}\cap Y_{o2}}{\arg\min} \frac{1}{2}(x^*Px+u^*Qu+w^*Rw) \tag{19a}$$

$$Y_{z_1} = \{(x, u, w) \in R^n : \mathscr{B}_1 y = Ax+Bu+Cw = z_1\} \tag{19b}$$

$$Y_{0_2} = \{(x, u, w) \in R^n : \mathscr{B}_2 y = w-MDx = 0\} \tag{19c}$$

Due to the special structure of the problem, it can be decomposed into several (say, k) subproblems:

$$(\hat{x}_j, \hat{u}_j, \hat{w}_j) = \underset{(x_j,u_j,w_j)\in Y_{z1j}}{\arg\min} \frac{1}{2}(x_j^*P_jx_j+j_j^*Q_ju_j+w_j^*R_jw_j) \tag{20a}$$

$$Y_{z_{1j}} = \{(x_j, u_j, w_j) \in R^{nj} : A_jx_j+B_ju_j+C_jw_j = z_{1j} \in R^{n_{xj}}\},\ j=1, \dots k \tag{20b}$$

provided the following global interaction constraints are satisfied

$$D_jx_j = w_{ij} \in R^{n_{wij}}, \quad j=1, \dots k \quad \to MDx = w \tag{20c}$$

The index ij denotes a variable of the i-th subproblem related to the j-th subproblem by the structural interaction matrix M. The vector D_jx_j can be interpreted as the output variable of a subsystem j, where x_j is determined by the internal realtion $A_jx_j+B_ju_j+C_jw_j=z_{1j}$ of the subsystem (e.g. a state equation in a steady-state regime). The output variable D_jx_j is acting as an input variable to another subsystem ij, determined by the matrix M which represents the structure of output-input feedbacks.

The following problems shall be considered here:

— how to make use of the structure of the problem (20a, b, c) in order to diminish the computational effort when applying the algorithm A4;

— what are the bounds of the computational effort;

— how can the algorithm A4 be interpreted and modified for possible applications in hierarchical control. The problem of the bounds of the computational effort shall be considered briefly first.

If the problem (19a, b, c) is solved globally by the algorithm A4, then the necessary number of iterations is

$$n+m = 2n_x + n_u + 2n_w \tag{21a}$$

If there are no global interactions, $n_w = 0$ or $M = 0$, then all subproblems (19a, b) can be solved parallely. The computational effort per interation is roughly the same for all subproblems solved parallely and for the global problem, but the necessary number of iterations drops to

$$l = \max_j (2n_{xj} + n_{uj} + n_{wj}) < n+m \tag{21b}$$

If global interactions do exists, then the necessary number of iterations is contained somewhere between b an $n+m$. To get more close estimates, it is necessary distinguish the folowing three cases:

Case A. All matrices P_j, Q_j, R_j are strictly positive, $\mathscr{A} > 0$, so that the global problem is strictly convex and normal and the saddle-point of the normal Lagrange function corresponds to the optimal solution. In this case, all subproblems are strictly convex, can be solved computationally by normal Lagrangian technique, and are coordinable by nornal Lagrange multipliers for global interaction constraints. It will be shown that the necessary number of iterations of a modified algorithm A4 to solve the problem is at most $l+n_w$ in this case.

Case B. The matrices P_j, Q_j, R_j and $\mathscr{A}$ are not strictly positive, but $\mathscr{A}$ is strictly positive in the subspace $Y_{01} = \{(x, u, w) \in R^n : Ax+Bu+Cw = 0\}$, $y^* \mathscr{A} y > 0$ for all $y \neq 0$, $y \in Y_{01}$. The solutions of the local problems exist and can be determined computationally as saddle-points of corresponding augmented Lagrange functions. Since the matrix $\mathscr{A} + \varrho \mathscr{B}^* \mathscr{B}_1$ is strictly positive for sufficiently large ϱ, the augmented local problems are coordinable by normal Lagrange multipliers for global interaction constraints and there is (at least theoretically) no need to penalize for the global constraints. The necessary member of iterations is $l+n_w$, the same as in case A.

Case C. The matrices P_j, Q_j, R_j and $\mathscr{A}$ are not strictly positive and $\mathscr{A}$ is strictly positive only in the subspace $Y_{01} \cap Y_{02}$, with Y_{02} defined by (19c). The solutions of the local problems (20a, b) might not exist, although there exist a unique solution of the global problem (20a, b, c) (19a, b, c). Since the global interaction constraints are responsible for the existence of the solution, ist is necessary to use an augmented Lagrange function for the global constraints (20c). Due to the particular nature of these constraints, the global augmented Lagrange function can be decomposed, into local goal functions, but only if the minimization with respect to w is performed globally, on the coordination level. Nevertheless, the algorithm A3 can be still applied and the necessary number of iterations is $\bar{l}+2n_w$, where

$$\bar{l} = \max_j (2n_{xj} + n_{uj}) \tag{21c}$$

and $\bar{l}+2n_w$ is only slightly greater than $1+n_w$.

4. CASE A: NORMAL LAGRANGE MULTIPLIER COORDINATION

The normal Lagrange function for the problem (19a, b, c) is

$$L(\lambda, \eta, y) = \frac{1}{2} y^* \mathscr{A} y + \eta^*(\mathscr{B}_1 y - z_1) + \lambda^* \mathscr{B}_2 y \tag{22a}$$

$$L(\lambda, \eta, x, u, w) = \frac{1}{2}(x^* P x + u^* Q u + w^* R w) + \\ + \eta^*(Ax + Bu + Cw - z_1) + \lambda^*(w - MDx) \tag{22b}$$

and the solution of the problem can be found by determining the saddle-point $((\hat{\lambda}, \hat{\eta}), (\hat{x}, \hat{u}, \hat{w}))$ of this function.

If λ is considered to be a coordinating parameter, then the Lagrange function can be decomposed into modified, but normal Lagrange functions for the subproblems (20a, b) with influence of the global constraints (20c)

$$L_j(\lambda, \eta_j, x_j, u_j, w_j) = \frac{1}{2}(x_j^* P_j x_j + u_j^* Q_j u_j + w_j^* R_j u_j) + \\ + \eta_j^*(A_j x_j + B_j u_j + C_j w_j - z_{1j}) + \lambda_j^* w_j - (\lambda^* M)_j D_j x_j \tag{23}$$

where $(\lambda^* M)_j = \lambda_{ij}$ according to (20c). The saddle-points $(\hat{\eta}_j(\lambda), (\hat{x}_j(\lambda), \hat{u}_j(\lambda), \hat{w}_j(\lambda)))$ of these local Lagrange functions can be found by the algorithm A4 in at most $l = \max n_j + m_j$, (where $n_j = n_{xj} + n_{uj} + n_{wj}$, $m_j = n_{xj}$) iterations. The analytical expressions for these saddle-points are

$$\hat{\eta}_j(\lambda) = \mathscr{A}_{\eta j}^{-1}(A_j P_j^{-1} D_j^* \lambda_{ij} - C_j R_j^{-1} \lambda_j) - \mathscr{A}_{\eta j}^{-1} z_{1j} \tag{24a}$$

$$\hat{x}_j(\lambda) = -P_j^{-1}(A_j^* \hat{\eta}_j(\lambda) - D_j^* \lambda_{ij}) \tag{24b}$$

$$\hat{u}_j(\lambda) = -Q_j^{-1} B_j^* \hat{\eta}_j(\lambda) \tag{24c}$$

$$\hat{w}_j(\lambda) = -R_j^{-1}(C_j^* \hat{\eta}_j(\lambda) + \lambda_j) \quad () \tag{24d}$$

where

$$\mathscr{A}_{\eta j} = A_j P_j^{-1} A_j^* + B_j Q_j^{-1} B_j^* + C_j R_j^{-1} C_j^* \tag{24e}$$

has its inverse approximated by a variable metric V_{vj}^i in the algorithm A4. The violation of the global interatcion constraints has the form

$$L_\lambda(\lambda, \hat{\eta}(\lambda), \hat{x}(\lambda), \hat{u}(\lambda), \hat{w}(\lambda)) = \hat{w}(\lambda) - MD\hat{x}(\lambda) = -\mathscr{A}_\lambda \lambda - \mathscr{C}_\eta \mathscr{A}_\eta^{-1} z_1 \tag{25a}$$

$$\mathscr{A}_\lambda = R^{-1} + MDP^{-1} D^* M^* - \mathscr{C}_\eta \mathscr{A}_\eta^{-1} \mathscr{C}_\eta^*$$

where $\mathscr{A}_\eta$ is defined by dropping out the indexes j in (24e), and

$$\mathscr{C}_\eta = MDP^{-1} A^* + R^{-1} C^* \tag{25b}$$

To determine λ such that $L_\lambda=0$, the inverse of the matrix $\mathscr{A}_\lambda = R^{-1}+ +MDP^{-1}D^*M^*-\mathscr{C}_\eta\mathscr{A}_\eta^{-1}\mathscr{C}_\eta^*$ must be approximated by another variable metric on the global level. The algorithm A4 must be accordingly modified. At the beginning, λ and η are kept constant for $\max(n_{xj}+n_{uj}+n_{wj})$ iterations (precisely speaking, $\max(2n_{xj}+n_{uj}+n_{wj})$—$\max n_{xj}$ iterations are sufficient). Thereafter, the algorithm A3 with two variable metrics (one for η and one for $y=(x, u, w)$) is utilized for $\max n_{xj}$ iterations. First after $l=\max(2n_{xj}+ +n_{uj}+n_{wj})$ iterations, a third variable metric (for λ^j) is built up on the coordination level with help of the gradients L_λ and utilized parallelly to the other two. Since $\mathscr{A}^{-1}$ and $\mathscr{A}_\eta^{-1}$ are already determined by the corresponding variable metrics, each change of λ results in the optimal (for subproblems) $\eta(\lambda)$, $\hat{x}(\lambda)$, $\hat{u}(\lambda)$, $\hat{w}(\lambda)$. After at most n_w iterations, $\mathscr{A}_\lambda^{-1}$ is determined by the third variable metric and the global solution is found.

The above method is actually a dual coordination technique, but with a special feature: there is no need for iterative optimization of subproblems after each change of coordinating paraneter.

5. CASE B: AUGMENTED LAGRANGIANS FOR SUBPROBLEM SOLVING, NORMAL LAGRANGE MULTIPLIER COORDINATION

If the subproblems are not strictly convex, but can be convexified by penalty terms for local constraints, the corresponding partly augmented Lagrange function has the form:

$$\Lambda_1(\lambda, \varrho_1, \vartheta_1, y) = \frac{1}{2} y^*\mathscr{A}y+\frac{1}{2}\varrho_1\|\mathscr{B}_1 y-z_1+\vartheta_1\|^2-\frac{1}{2}\varrho_1\|\vartheta_1\|^2+\lambda^*\mathscr{B}_2 y \tag{26}$$

If ϱ_1 is sufficiently large and Λ_1 is strictly convex in y, then λ can be again used as a coordinating parameter. Due to the particular structure of the matrices $\mathscr{A}, \mathscr{B}_1, \mathscr{B}_2$, modified and augmented Lagrange functions for the subproblems (20a, b) with the influence of the global constraints (20c) can be defined

$$\Lambda_{1j}(\lambda, \varrho_1, \vartheta_{1j}, y_j) = \frac{1}{2} y_j^*(\mathscr{A}_j+\varrho_1\mathscr{B}_{1j}^*\mathscr{B}_{1j})y_j+ + \varrho_1\vartheta_{1j}^*(\mathscr{B}_{1j}y_j-z_{1j})+\frac{1}{2}\varrho_1\|z_{1j}\|^2+(\lambda^*\mathscr{B}_2 y)_j \tag{27a}$$

$$\Lambda_{1j}(\lambda, \varrho_1, \vartheta_{1j}, x_j, u_j, w_j) = \frac{1}{2}(x_j^*P_jx_j+u_j^*Q_ju_j+w_j^*R_jw_j)+ +\frac{1}{2}\varrho_1\|A_jx_j+B_ju_j+C_jw_j-z_{1j}+\vartheta_{1j}\|^2- -\frac{1}{2}\varrho_1\|\vartheta_{1j}\|^2+\lambda_j^*w_j-\lambda_{ij}^*D_jx_j \quad (27b$$

The algorithm A4 can be used to determine the saddle-points $(\bar{v}_{ij}(\lambda),\hat{y}_j(\lambda))$ of the functions Λ_{1j}. Since the penalty term for local constraints contains bilinear forms in x_j, u_j, w_j, the analytical expressions for $\hat{x}_j(\lambda)$, $\hat{u}_j(\lambda)$, $\hat{w}_j(\lambda)$ are fairly complicated and it is more convenient to write down the joint expression for $\hat{y}_j(\lambda)$

$$\hat{\vartheta}_{1j}(\lambda) = -\mathscr{A}_{\vartheta j}^{-1}(\mathscr{B}_{1j}\mathscr{A}_{yj}^{-1}(\mathscr{B}_2^*\lambda)_j + z_{1j}) \tag{28a}$$

$$\hat{y}_j(\lambda) = -\mathscr{A}_{yj}^{-1}(\varrho_1\mathscr{B}_{1j}^*\hat{\vartheta}_{1j}(\lambda) + (\mathscr{B}_2^*\lambda)_j) \tag{28b}$$

where

$$\mathscr{A}_{yj} = \mathscr{A}_j + \varrho_1\mathscr{B}_{1j}^*\mathscr{B}_{1j}; \qquad \mathscr{A}_{\vartheta j} = \varrho_1\mathscr{B}_{1j}\mathscr{A}_{yj}^{-1}\mathscr{B}_{1j}^* \tag{28c}$$

have their inverses approximated by two variable metrics in the algorithm A4. The violation of the global constraints has the form

$$\Lambda_{1\lambda}(\lambda, \varrho_1, \hat{\vartheta}_1(\lambda), \hat{y}(\lambda)) = \mathscr{B}_2\hat{y}(\lambda) = -\mathscr{A}_\lambda\lambda + \mathscr{B}_2\mathscr{A}_y^{-1}\mathscr{B}_1^*\mathscr{A}_\vartheta^{-1}z_1 \tag{29a}$$

where

$$\mathscr{A}_\lambda = \mathscr{B}_2\mathscr{A}_y^{-1}\mathscr{B}_2^* - \varrho_1\mathscr{B}_2\mathscr{A}_y^{-1}\mathscr{B}_1^*\mathscr{A}_\vartheta^{-1}\mathscr{B}_1\mathscr{A}_y^{-1}\mathscr{B}_2^* \tag{29b}$$

and the matrices $\mathscr{A}_y$, $\mathscr{A}_\vartheta$ are defibed as in (28c) for global variables. The inverse of the matrix $\mathscr{A}_\lambda$ must be approximated by a third variable metric on the global level in the algorithm A4 modified in the same way as in case A. The number of necessary iterations is also the same as in case A. The above method differs from the method applied in case A only by penalizing for the local constraints.

There are some reasons, however, for using the above method as a universal one for both cases A and B. First, it is not allways a priori known, whether the matrix $\mathscr{A}$ is strictly positive or not. Secondly, by choosing the value of the penalty coefficient ϱ_1, one can influence the conditioning of matrices $\mathscr{A}_y$, $\mathscr{A}_\vartheta$, $\mathscr{A}_\lambda$. The matrix $\mathscr{A}_y = \mathscr{A} + \varrho_1\mathscr{B}_1^*\mathscr{B}_1$ becomes badly conditioned, if ϱ_1 is too large; but by increasing ϱ_1 one can only improve the conditioning of the matrix $\mathscr{A}_\vartheta$. It can be proven (see e.g. [3]) that

$$\mathscr{A}_\vartheta^{-1} = \frac{1}{\varrho_1}(\mathscr{B}_1(\mathscr{A} + \varrho_1\mathscr{B}_1^*\mathscr{B}_1)^{-1}\mathscr{B}_1^*)^{-1} =$$

$$= \frac{1}{\varrho_1}(\mathscr{B}_1(\mathscr{A} + \varrho_0\omega_1^*\mathscr{B}_1)\mathscr{B}_1^*)^{-1} + \frac{\varrho_1 - \varrho_0}{\varrho_1}I \tag{30}$$

where ϱ_0 is such that $(\mathscr{A} + \varrho_0\mathscr{B}_1^*\mathscr{B}_1)^{-1}$ exists. Hence, $\mathscr{A}_\vartheta^{-1} \to I$ if $\varrho_1 \to \infty$ and the conditioning of the matrix $\mathscr{A}_\vartheta$ improves. Similarly, it can be shown that the matrix $\mathscr{A}_\lambda$ is arbitrarily close to $\mathscr{B}_2\mathscr{A}_y^{-1}\mathscr{B}_2^*$ for large ϱ_1 and $\mathscr{A}_\lambda$ becomes badly conditioned if ϱ_1 is too large. Therefore, there is a compormise betweeh the conditioning indices of $\mathscr{A}_y$, $\mathscr{A}_\lambda$ and $\mathscr{A}_\vartheta$. The experience in practical applications of shifted penalty algorithms shows that, in most cases, one can choose a sufficiently large value of ϱ_1 such that the matrices $\mathscr{A}_y$, $\mathscr{A}_a$ are reasonably well conditioned; however, the matrix $\mathscr{A}_\lambda$ can be badly conditioned and influence adversely the computations.

6. CASE C: AUGMENTED LAGRANGIANS FOR SUBPROBLEM SOLVING AND COORDINATION: A PRIMAL-DUAL METHOD

If the subproblems cannot be convexified by penalty terms for local constraints, the following fully augmented Lagrange function must be used:

$$\Lambda_2(\varrho_2, \vartheta_2, \varrho_1, \vartheta_1, y) = \frac{1}{2} y^* \mathscr{A} y + \frac{1}{2} \varrho_1 \|\mathscr{B}_1 y - z_1 + \vartheta_1\|^2 -$$

$$-\frac{1}{2} \varrho_1 \|\vartheta_1\|^2 + \frac{1}{2} \varrho_2 \|\mathscr{B}_2 y + \vartheta_2\|^2 - \frac{1}{2} \varrho_2 \|\vartheta_2\|^2 =$$

$$= \frac{1}{2} y^* (\mathscr{A} + \varrho_1 \mathscr{B}_1^* \mathscr{B}_1 + \varrho_2 \mathscr{B}_2^* \mathscr{B}_2) y + \varrho_1 \vartheta_1^* (\mathscr{B}_1 y - z_1) +$$

This function could be decomposed into local Lagrangians but for the term $\frac{1}{2} \varrho_2 y^* \mathscr{B}_2^* \mathscr{B}_2 y$ which — due to the assumption (20c) — can be expressed as:

$$\frac{1}{2} \varrho_2 y^* \mathscr{B}_2^* \mathscr{B}_2 y = \frac{1}{2} \varrho_2 \Big(\sum_{j=1}^{k} (\|w_j\|^2 + \|D_j x_j\|^2) - 2 \sum_{j=1}^{k} w_{ij}^* D_j x_j \Big) \qquad (32)$$

The last term of this expression cannot be decomposed into local functions, if the minimization in respect to w_j, x_j is to be performed locally. Therefore, the vector w composed of w_j or w_{ij} must be used as a coordinating parameter and the local minization can be performed in respect to $y_j = (x_j, u_j)$ only. The following change of notation is needed:

$$\overline{\mathscr{A}} = \begin{bmatrix} P & O \\ O & Q \end{bmatrix}; \qquad f(\bar{y}, w) = \frac{1}{2} (\bar{y}^* \overline{\mathscr{A}} \bar{y} + w^* R w) \qquad (33a)$$

$$\overline{\mathscr{B}}_1 = [A \quad B]; \qquad \overline{\mathscr{B}}_1 \bar{y} + C w = z_1 \qquad (33b)$$

$$\overline{\mathscr{B}}_2 = [-MD \quad O]; \qquad \overline{\mathscr{B}}_2 \bar{y} + w = 0 \qquad (33c)$$

and the fully augmented Lagrange function (31) can be rewritten as

$$\Lambda_2(\varrho_2, \vartheta_2, \varrho_1, \vartheta_1, \bar{y}, w) = \frac{1}{2}\, \bar{y}^*\bar{\mathscr{A}}_y \bar{y} + \varrho_1 \vartheta_1^*(\bar{\mathscr{B}}_1 \bar{y} + Cw - z_1) +$$

$$+ \varrho_2 \vartheta_2^*(\bar{\mathscr{B}}_2 \bar{y} + w) + \varrho_1 (Cw - z_1)^* \bar{\mathscr{B}}_1 \bar{y} + \varrho_2 w^* \bar{\mathscr{B}}_2 \bar{y} +$$

$$+ \frac{1}{2} \varrho_1 \| Cw - z_1 \|^2 + \frac{1}{2} \varrho_2 \| w \|^2 + \frac{1}{2} w^* R w \qquad (34a)$$

where

$$\bar{\mathscr{A}}_y = \bar{\mathscr{A}} + \varrho_1 \bar{\mathscr{B}}_1^* \bar{\mathscr{B}}_1 + \varrho_2 \bar{\mathscr{B}}_2^* \bar{\mathscr{B}}_2 \qquad (34b)$$

The coordination method consists in seeking for the saddlepoint

$$(\hat{\vartheta}_2, \hat{\vartheta}_1, \hat{\bar{y}}, \hat{w}) = \arg \max_{\vartheta_2 \in R^{n_w}} \min_{w \in R^{n_w}} \max_{\vartheta_1 \in R^{n_x}} \min_{y \in R^{n_x\ n_w}} \Lambda_2(\varrho_2, \vartheta_2, \varrho_1, \vartheta_1\ \bar{y}, w) \qquad (35a)$$

on the global level in ϑ_2, w and on the local level in $\vartheta_1, \bar{y}$. The saddle-point exists under the assumptions of case C, if ϱ_2, ϱ_1 are sufficiently large. The fully augmented Lagrange functions can be decomposed into local modified and augmented Lagrangians

$$\Lambda_{2j}(\varrho_2, \ _2, \varrho_1, \vartheta_{1j}, y_j, w) = \frac{1}{2}\, \bar{y}_j^* \bar{\mathscr{A}}_{yj} \bar{y}_j + \varrho_1 \vartheta_{1j}^*(\bar{\mathscr{B}}_{1j} \bar{y}_j + C_j w_j - z_1) +$$

$$+ \varrho_1 (C_j w_j - z_{1j})^* \bar{\mathscr{B}}_{1j} \bar{y}_j + \varrho_2 \vartheta_{2j}^*(\bar{\mathscr{B}}_{2j} \bar{y}_j + w_{ij}) + \varrho_2 w_{ij}^* \bar{\mathscr{B}}_{2j} \bar{y}_j \qquad (35b)$$

The remaining terms in (34a) do not depend on ϑ_1, y. The saddle-points in ϑ_{1j}, y_j for subproblems can be determined by the algorithm A4 in at most $\bar{l} = \max\,(2n_{xj} + n_{uj})$ iterations. The analytical forms for $\hat{y}, \hat{\vartheta}_1$ are:

$$\hat{\bar{y}}_j(\vartheta_2, \vartheta_1, w) = -\bar{\mathscr{A}}_{yj}^{-1}(\varrho_1 \bar{\mathscr{B}}_{1j}^*(C_j w_j - z_{1j} + \vartheta_{1j}) + \varrho_2 \bar{\mathscr{B}}_{2j}^*(w_{ij} + \vartheta_{ij})) \qquad (36a)$$

$$\hat{\vartheta}_{1j}(\vartheta_2, w) = \bar{\mathscr{A}}_{\vartheta 1j}^{-1}((I - \bar{\mathscr{A}}_{\vartheta 1j})(C_j w_j - z_{1j}) - \varrho_2 \bar{\mathscr{B}}_{1j} \bar{\mathscr{A}}_{yj}^{-1} \bar{\mathscr{B}}_* (w_{ij} + \vartheta_{ij})) \qquad (36b)$$

where

$$\bar{\mathscr{A}}_{\vartheta 1j} = \varrho_1 \bar{\mathscr{B}}_{1j} \bar{\mathscr{A}}_{yj}^{-1} \bar{\mathscr{B}}_{1j}^* \qquad (36c)$$

and

$$\hat{\bar{y}}_j(\vartheta_2, w) = -\bar{\mathscr{A}}_{yj}^{-1}((I - \varrho_1 \bar{\mathscr{B}}_{1j}^* \bar{\mathscr{A}}_{\vartheta 1j}^{-1} \bar{\mathscr{B}}_{1j} \bar{\mathscr{A}}_{yj}^{-1})\, \varrho_2 \bar{\mathscr{B}}_{2j}^*(w_{ij} + \vartheta_{2j}) +$$

$$+ \varrho_1 \bar{\mathscr{B}}_{1j}^* \bar{\mathscr{A}}_{\vartheta 1j}^{-1}(C_j w_j - z_{1j})) \qquad (36d)$$

By omitting the indexes j or ij one obtains the global variables $\hat{y}(\vartheta_2, w)$ and $\hat{\vartheta}_1(\vartheta_2, w)$. The fully augmented Lagrange function (34a) takes the form

$$\Lambda_2(\varrho_2, \vartheta_2, \varrho_1, w) = \max_{\vartheta_1 \in R^{n_x}} \min_{y \in R^{n_x+n_u}} \Lambda_2(\varrho_2, \vartheta_2, \varrho_1, \vartheta_1, \bar{y}, w) =$$

$$= \frac{1}{2} w^* \bar{\mathscr{A}}_w w + w^*(\varrho_2 I - \Omega - C^* \Pi^*) \vartheta_2 -$$

$$- \frac{1}{2} \vartheta_2^* \Omega \vartheta_2 + \vartheta_2 \Pi z_1 - w^*(\varrho_1 C^*(\bar{\mathscr{A}}_{\vartheta 1}^{-1} - I) - \Pi) z_1 \qquad (37a)$$

where

$$\Omega = \varrho_2^2 \bar{\mathscr{B}}_2(\bar{\mathscr{A}}_y^{-1} - \varrho_1 \bar{\mathscr{A}}_y^{-1} \bar{\mathscr{B}}_1^* \bar{\mathscr{A}}_{\vartheta 1}^{-1} \bar{\mathscr{B}}_1 \bar{\mathscr{A}}_y^{-1}) \bar{\mathscr{B}}_2^* \qquad (37b)$$

$$\pi = \varrho_1 \varrho_2 \bar{\mathscr{B}}_2 \bar{\mathscr{A}}_y^{-1} \bar{\mathscr{B}}_1^* \bar{\mathscr{A}}_{\vartheta 1}^{-1} \qquad 37(c)$$

$$\bar{\mathscr{A}}_w = \varrho_2 I + R + \varrho_1 C^*(\bar{\mathscr{A}}_{\vartheta 1}^{-1} - I) C - \Omega - \Pi C - C^* \Pi^* \qquad (37d)$$

If ϱ_2 is sufficiently large, then $\bar{\mathscr{A}}_w > 0$ and there exists a saddle-point of the Lagrangian (37a). This saddle-point can be determined by the algorithm A4 applied to the global coordination level. More precisely, a suitable modification of the algorithm A3 can be applied simulataneously to local and global problems: first, the inverses of $\bar{\mathscr{A}}_y$ and $\bar{\mathscr{A}}_{\vartheta i}$ are approximated by corresponding variable metric for τ iterations, then w is changed and the inverse of $\bar{\mathscr{A}}_w$ is approximated by a third variable metric for n_w iterations. Thus, the minimizing argument $\hat{w}(\vartheta_2)$ of Λ_2 is determined

$$\hat{w}(\vartheta_2) = -\bar{\mathscr{A}}_w^{-1}((\varrho_2 I - \Omega - C^* \Pi^*) \vartheta_2 - (\varrho_2 C^*(\bar{\mathscr{A}}_{\vartheta 1}^{-1} - I) - \Pi) z_1) \qquad (38a)$$

and the coordination of the global constraints can be started. The violation of these constraints takes the form

$$\bar{\mathscr{B}}_2 \bar{y} + w = \frac{1}{\varrho_2} \Lambda_{2\vartheta 2} = \frac{1}{\varrho_2} ((\varrho_2 I - \Omega - \Pi C) \hat{w}(\vartheta_2) - \Omega \vartheta_2 + \Pi z_1) \qquad (38b)$$

$$= -\frac{1}{\varrho_2} (\bar{\mathscr{A}}_{\vartheta 2} \vartheta_2 - \Theta z_1)$$

where

$$\bar{\mathscr{A}}_{\vartheta 2} = \Omega + (\varrho_2 I - \Omega - \Pi C) \bar{\mathscr{A}}_w^{-1} (\varrho_2 I - \Omega - C^* \Pi^*) \qquad (38c)$$

$$\Theta = (\varrho_2 I - \Omega - \Pi C) \bar{\mathscr{A}}_w^{-1} (\varrho_1 C^*(\bar{\mathscr{A}}_{\vartheta 1}^{-1} - I) - \Pi) + \Pi \qquad (38d)$$

and $\bar{\mathscr{A}}_2^{-1}$ can be approximated by a fourth variable metric in n_w iterations, resulting in

$$\hat{\vartheta}_2 = \bar{\mathscr{A}}_{\vartheta 2}^{-1} \Theta z_1 \qquad (38e)$$

after a total number of $\tau + 2n_w$ iterations.

As an example, consider a problem composed of $k = 10$ subproblems with equal dimmensions $n_{xj} = 5$, $n_{uj} = 3$, $n_{wj} = 1$. The global dimmension of variables is $n_x+n_u+n_w=90$, the dimmension of constraints is $n_x+n_w=60$. Without decomposition, the problem can be solved in 150 iterations. If the subproblems are strictly convex or can be convexified by penalties for local constraints (cases A or B), then an application of the modified algorithm A4 gives the solution in $\bar{l}+n_w=\max(2n_{xj}+n_{uj}+n_{wj})+n_x=24$ iterations. If the global constraints must be penalized in order to convexify the subproblems (case C), the modified algorithm A4 finds the solution in $\bar{l}+2n_w=\max(2n_{xj}+$ $+n_{uj})+2n_w=33$ iterations. Thus, the number of necessary iterations slightly inereases.

The coordination method used in (35a) is actually a primal-dual method, as opposed to dual methods used in cases A and B. The primal-dual method requires slightly more iterations, but there are reasons to use it as a universal method. The reasons are similar to those stated in the case B: it might be not apriori known, whether the subproblems can be convexified by penalizing local constraints only. Moreover, it can be proved (by arguments similar to (30) that the conditioning indices of $\bar{\mathscr{A}}_{\varrho i}$ and $\bar{\mathscr{A}}_{\varrho 2}$ converge to one for sufficiently large ϱ_1, ϱ_2. It is expected that ϱ_1 and ϱ_2 can be chosen to guarantee a reasonable conditioning of $\bar{\mathscr{A}}_y$ and $\bar{\mathscr{A}}_w$ and a good conditioning of $\bar{\mathscr{A}}_{\varrho 1}, \bar{\mathscr{A}}_{\varrho 2}$. In fact, an automatical choice of ϱ_1 and ϱ_2 can be incorporated into the modified algorithm A4. Theorefore, the primal-dual coordination method based on augmented Lagrangians seems to overcome known difficulties with the conditioning of dual coordination.

7. POSIBLE EXTENSIONS

The main advantage of the algorithm A4 when modified for large scale problems is not that it solves a quadratic problem with linear constraints in substantially reduced number of iterations; this could be achieved also by other methods, for example, by typical quadratic programming methods with suitable decomposition. But the algorithm A4 can be directly extended to applications for nonquadratic problems with nonlinear constraints. It is only required for large scale problems that the global constraints should have a simple structure such that a decomposition of the fully augmented Lagrange function is possible similarly, as in case C. In fact, consider the problem

$$(\hat{x}, \hat{u}, \hat{w}) = \underset{(x,u,w)\in Y_{z1}\cap Y_{o2}}{\arg\min} f(x, u, w); \quad f(x, u, w) = \sum_{j=1}^{k} f_j(x_j, u_j, w_j) \tag{39a}$$

where

$$Y_{z1} = \{(x, u, w) \in R^n : g_j(x_j, u_j, w_j) = z_{1j},\ j = 1, \dots k\} =$$

$$= \{(x, u, w) \in R^n : g(x, u, w) = z_1\} \tag{39b}$$

is the set of admissible solutions defined by local constraints, and

$$Y_{02} = \{(x, u, w) \in R^n : h_j(x_j) = w_{ij}, \ j = 1, \dots k\} =$$
$$= \{(x, u, w) \in R^n : Mh(x) = w\} \tag{39c}$$

is the set of admissible solutions defined by global interaction constraints (the structural matrix M is composed of zero and unit elements). The generalization of the methods presented above to this problem is straightforward. For example, the fully augmented Lagrange function (as is case C) is

$$\Lambda_2(\varrho_2, \vartheta_2, \varrho_1, \vartheta_1, x, u, w) = f(x, u, w) + \frac{1}{2}\varrho_1 \|g(p, u, w) - z_1 + \vartheta_1\|^2 -$$
$$-\frac{1}{2}\varrho_1\|\vartheta_1\|^2 + \frac{1}{2}\varrho_2\|Mh(x) - w + \vartheta_2\|^2 - \frac{1}{2}\varrho_2\|\vartheta_2\|^2 \tag{40}$$

and this function has a saddle-point in $(\vartheta_2, \vartheta_1)$, (x, u, w), provided the second-order sufficient condition for a solution of the problem (39a, b, c) are satisfied and ϱ_1, ϱ_2 are sufficiently large — see [5]. Moreover, this function can be decomposed into local Lagrangians similarly, as in case *C*.

The algorithm A4 modified to solve large scale problems is fully utilized first for non-quadratic and non-linear problems. Consider as an example the coordination in ϑ_1 on the global level. The gradient $\frac{1}{\varrho_2}\Lambda_{2\vartheta 2}$ is equal to the violation of the global constraints

$$\frac{1}{\varrho_2}\Lambda_{2\vartheta 2} = Mh(x) - w \tag{41}$$

Since the Lagrange function is not quadratic in $y = (x, u, w)$, its gradient with respect to y is not equal to zero after the given number (say, $\tau + n_w$) of iterations. Since $b_y^i \neq 0$ in (10a), the step (10b) is actually the prediction of the violation of global constraints after a quasi-Newton change of y, and the compensation of this predicted violation by a quasi-Newton change of the coordinating variable $v = \vartheta_2$. Superlinear convergence in all variables is expected in such a case.

An interpretation of the algorithmic idea is important for possible extensions to on-line hierarchical control (as opposed to off-line multilevel optimization). After initial, given number of optimization iterations for subproblems, an estimate of the solutions for subproblems is found together with an estimate of the second-order derivatives for subproblems. Once this information is known, it is not necessary to repeat many optimization iterations for subproblems after each change of coordinating variable. One quasi-Newton iteration gives a good estimate of the solutions for subproblems for any given value of coordinating variable. To speed up the coordination, a prediction and compensation of the violation of the global constraints is used. This provides for a simultaneous coordination and local optimization which can be of great value for on-line hierarchical methods of control.

Clearly, the possible extension of this algorithmic idea to on-line hierarchical control must be studied more deeply because of the known special features of such problems (hard constraints are automatically satisfied in a controlled plant and the violation of these constraints does not actually occur, but the coordination errors result in additional deviations from the optimal solution — see [12]). Other extensions are also possible. The augmented Lagrange functions can be defined for problems with inequality constraints [5] and even for equality and inequality constraints in a Hilbert space [6] (practically speaking, for discretized dynamic optimization problems with a very large dimmensionality of the equivalent equality and inequality constraints). Moreover, these augmented Lagrange functions posses also saddle-points under appropriate, mild assumptions [5], [6]. For example, if the global constraints (39c) take the form

$$Y_{02} = \{(x, u, w) \in R^n : h_j(x_j) \quad w_{ij},\ j = 1, \dots k_1;\ h_j(x_j) = w_{ij},\ j = k_1+1, \dots k\} \tag{42a}$$

then the fully augmented Lagrange function is

$$\Lambda_2(\varrho_2, \vartheta_2, \varrho_1, \vartheta_1, x, u, w) = f(x, u, w) + \frac{1}{2}\varrho_1 \|g(x, u, w) - z_1 + \vartheta_1\|^2 -$$

$$-\frac{1}{2}\varrho_1\|\vartheta_1\|^2 + \frac{1}{2}\varrho_2\Big(\sum_{j=1}^{k_1}\|\max(O, h_j(x_j) - w_{ij} + \vartheta_{2j})\|^2 +$$

$$+\sum_{j=k_1+1}^{k_1}\|h_j(x_j) - w_{ij} + \vartheta_{2j}\|^2\Big) - \frac{1}{2}\varrho_2\|\vartheta_2\|^2 \tag{42b}$$

The generalization of the global constraints (41a) to a Hilbert space is

$$Y_{02} = \{(x, u, w) \in H_y : w - Mh(x) \in D \subset H_w\} \tag{43a}$$

where D is a given positive cone in the space of global constraints H_w, and H_y is the solution space (if $H_w = R^{nw}$ and $D = D_1 = \{w \in R^{ns}:\ w_{ij} \geqslant 0, j = 1, \dots k_1;$ $w_{ij} = 0, j = k_1+1, \dots k\}$, then the sets Y_{02} given by (42a) and (43a) are identical). The function $f : H_y \to R^1$ should be interpreted as a performance functional, and the local constraining functions $g_j : H_{uj} \to H_{xj}$ can, for example, express the state equations for local variables x_j with controls u_j, w_j and disturbances z_j. The fully augmented Lagrange functional takes the form

$$\Lambda_2(\varrho_2, \vartheta_2, \varrho_1, \vartheta_1, x, u, w) = f(x, u, w) + \frac{1}{2} \varrho_1 \| g(x, u, w) - z_1 + \vartheta_1 \|^2 -$$

$$- \frac{1}{2} \varrho_1 \| \vartheta_1 \|^2 + \frac{1}{2} \varrho_2 \| (Mh(x) - w + \vartheta_2)^{D^*} \|^2 - \frac{1}{2} \varrho_2 \| \vartheta_2 \|^2 \tag{43b}$$

where $(\cdot)^{D^*}$ denotes the projection on the dual cone $D^* = \{w^* \in H_w : < w^*, w > \geqslant 0 \; \forall \; w \in D\}$. The analytical form of this projection is usually simple to determine (for example, if $H_w = R^{nw}$ with $D = D_1$ as above, then (42b) and (43b) are identical). If the local constraints correspond to state equations, it is often not necessary to penalize for these constraints, since they can be solved for x given u, w, z_1, and the augemented Lagrange functional (43b) takes a more simple form. The full form (43b) corresponds actually to an extension of the Balakrishnan ε-technique for differential constraints [13].

8. CONCLUSIONS

The algorithm A4 of saddle-point seeking can be modified to solve large scale problems. One of possible modifications of this algorithm corresponds to a primal-dual method of coordination. An important feature of coordination methods based on the algorithm A4 is that the local optimization and coordination are simultaneous. Thus, large scale problems of quadratic programming are solved by these methods in finite and small number of steps. The additional advantage of the primal-dual method is that the coordination problem can be made well-conditioned by a suitable choice of penalty coefficients. However, the most important advantage of coordination methods based on the algorithm A4 is that they can be easily extended to nonlinear and infinite-dimmensional problems.

BIBLIOGRAPHY

[1] WIERZBICKI, A. P.: Ein neuer Algorithmus zur Sattelpunktermittlung und ein universaller Strafverschiebungsalgorithmus für beschränkte nichtlineare Optimierungsprobleme in R_n und Hilbertraum. 21 Internationale Wissenschaftliche Kolloquium der Technischen Hochschule Ilmenau (1976).

[2] FLETCHER, R., M. J. D. POWELL: A rapidly convergent descent method for minimization. Comput. J. 6, 163—168 (1963).

[3] FLETCHER, R.: An ideal penalty function for constrained optimization. J. Inst. Maths Applics 15, 319—342 (1975).

[4] HESTENES, M. R.: Multiplier and gradient methods. JOTA 4, 303—320 (1969).
[5] ROCKAFELLAR, R. T.: Augmented Lagrange multiplier functions and duality in non-convex programming. SIAM J. Control 12, 268—285 (1974).
[6] WIERZBICKI A. P., St. KURCYUSZ: Projection on a cone, penalty functionals and duality theory for problems with inequality constraints in Hilbert space. SIAM J. Control 15, No 1 (1977).
[7] POWELL M. J. D.: A method for nonlinear constraints in minimization problems. In: Optimization, ed. R. Fletcher, Academic Press 1969.
[8] WIERZBICKI A. P.: A penalty shifting method in constrained optimization and its convergence properties. Archiwum Automatyki i Telemechaniki 16, 395—416 (1971).
[9] MANGASARIAN O. L. Unconstrained Lagrangians in nonlinear programming. SIAM J. Control 13, No 3 (1975).
[10] MARTENSSON K.: A new approach to constrained function optimization. JOTA 12, 531—555 (1972).
[11] BERTSEKAS D. P.: Multiplier methods — a survey. Automatica 12, 135—146 (1976).
[12] FINDEISEN W.: Wielopoziomowe układy sterowania (Multilevel control systems). PWN 1974.
[13] BALAKRISHNAN A. V.: On a new computing technique in optimal control. SIAM J. Control 6, 149—173 (1968).

SUMMARY

The developements of the theory of augmented Lagrange functions (or, equivalently, shifted penalty functions) resulted in powerful saddle-point theorems and, recently, in new single-loop iterative algorithms for saddle-point seeking and for solving optimization problems with constraints. A particularly strong new algorithm is based on two variable metric approximations and a constraint shift (or violation) prediction. For large scale optimization, this algorithm leads to a primal-dual coordination method. The method converges in a finite member of steps for quadratic problems with linear constraints and interactions, and generally converges rapidly for more complicated problems. The method has also other advantages of shifted penalty or augmented Lagrange funktions when applied to large scale optimization. It is also hoped that the method can be used for on-line coordination in hierarchical control systems.

Instytut Badań Systemowych PAN
Nakład 300 egz. Ark. wyd. 25,0. Ark. druk. 23,75. Papier druk. sat. kl. III 80 g 61×86. Oddano do składania 8 X 1976
Podpisano do druku w sierpniu 1978 r. Druk ukończono w sierpniu 1978 roku

CDW — Zakład nr 5 w Bielsku-Białej zam. 62/K/77 J-124