Q

QBB Global Optimization Method

YUSHAN ZHU
Department of Chemical Engineering,
Tsinghua University, Beijing, China

MSC2000: 49M37, 65K10, 90C26, 90C30

Article Outline

Keywords and Phrases

Global optimization; Branch-and-bound; Simplicial partition; Quadratic underestimation function

Introduction

Most problems of process design, process control, process operations, and molecule design are determined by the optimal solutions; however, those problems are mainly characterized by the existence of multiple minima and maxima, as well as first-, second-, and higher-order saddle points. During the last decade we have experienced a rapid development of new methods for deterministic global optimization as well as the application of available global optimization algorithms in important engineering fields [1,2,9,10,11,12,13,14]. Recently, in order to locate the global solutions to the nonconvex phase stability analysis problems [3,4,5], a quadratic underestimation function based branch-and-bound algorithm, i. e., QBB, was developed for twice-differentiable nonlinear programs (NLPs) in terms of the simplicial partition of the constrained region [6,7].

Formulation

The nonconvex optimization problem considered in this section can be formulated as

$$(\mathrm{P}) \left| \begin{array}{ll} \min\limits_{\mathbf{x}} & f(\mathbf{x}) \\ \text{subject to} & g_i(\mathbf{x}) \le 0 \quad i = 1, \ldots, m, \\ & \mathbf{x} \in \mathbf{S}^0 \subset \Re^n, \end{array} \right.$$

where f and $\mathbf{g}_i$ belong to C^2, the set of twice-differentiable functions, and $\mathbf{S}^0$ is a simplex defined by

$$\mathbf{S}^0 = \left\{ \mathbf{x} \in \Re^n : \mathbf{x} = \sum_{i=1}^{n+1} \lambda_i \mathbf{V}^i,\ \lambda_i \ge 0,\ \sum_{i=1}^{n+1} \lambda_i = 1 \right\},$$

where $\mathbf{V}^i \in \mathbf{V} \subset \Re^n$, $i = 1, 2, \ldots, n+1$ are the $n+1$ vertices of the simplex $\mathbf{S}^0$, and $\mathbf{V}$ is the set of its vertices. Let D_g be a subset of $\Re^n$ defined by

$$D_g = \{ \mathbf{x} \in \Re^n : \mathbf{g}_i(\mathbf{x}) \le 0,\ i = 1, 2, \ldots, m \}.$$

In general, the set D_g is nonconvex and even disconnected. We assume throughout this section that problem (P) has an optimal solution, unless otherwise stated. For any nonconvex optimization problem, i. e., (P), the QBB algorithm proposed in this section belongs

to a branch-and-bound scheme. During each iteration of this framework, a branching step and a bounding step must be finished simultaneously.

Simplicial Partition

For the branching procedure, the simplex $\mathbf{S}^0$ will be divided into refined subregions by using simplicial partition. For such kind of branching, it is a simple matter to check that for every $i \in I$, where I is the vertex set of $\mathbf{S}^0$, the points $\mathbf{V}^1, \ldots, \mathbf{V}^{i-1}, \mathbf{U}, \mathbf{V}^{i+1}, \ldots, \mathbf{V}^{n+1}$ are vertices of a simplex $\mathbf{S}_i \subset \mathbf{S}$, $\mathbf{S}$ is the current simplex, and that

$$(\text{int } \mathbf{S}_i) \cap \left(\text{int } \mathbf{S}_j\right) = \phi \quad \forall j \neq i\,; \quad \bigcup_{i \in I} \mathbf{S}_i = \mathbf{S}\,.$$

Then, the simplexes $\mathbf{S}_i$, $i \in I$, form a subdivision of the simplex $\mathbf{S}$ via $\mathbf{U}$. Each $\mathbf{S}_i$ will be referred to as a subsimplex of $\mathbf{S}$. An important special case is the bisection where $\mathbf{U}$ is a point of the longest edge of the simplex $\mathbf{S}$, for example, $\mathbf{U} \in [\mathbf{V}^m, \mathbf{V}^n]$, i. e.

$$\|\mathbf{V}^m - \mathbf{V}^n\| = \max_{\substack{i<j \\ i,\, j=1,\ldots n+1}} \left\{\left\|\mathbf{V}^i - \mathbf{V}^j\right\|\right\}\,,$$

where $\|\cdot\|$ denotes any given norm in $\Re^n$, and $\mathbf{U} = a\mathbf{V}^m + (1-a)\mathbf{V}^n$ with $0 < a \leq 1/2$. Adjiman et al. [9] proved that this simplicial bisection is exhaustive since $\delta(\mathbf{S}_k) \to 0$ as $k \to +\infty$.

Quadratic Underestimation Function for General Non-convex Structures

In the bounding step of a branch-and-bound algorithm, a lower bound is always obtained by constructing a valid convex underestimation problem for the original one appearing in the problem (P), and solving the relaxed convex NLP to global optimality. For the current simplex given by

$$\mathbf{S} = \left\{ x \in \Re^n : x = \sum_{i=1}^{n+1} \lambda_i \mathbf{V}^i,\ \lambda_i \geq 0,\ \sum_{i=1}^{n+1} \lambda_i = 1 \right\}\,, \tag{1}$$

where $\mathbf{V}^i \in \mathbf{V} \subset \Re^n$, $i = 1, 2, \ldots, n+1$ are the $n+1$ vertices of the current simplex $\mathbf{S}$, and $\mathbf{V}$ is the set of these vertices. Then, we intend to compute a lower bound $\mu(S)$ of the objective function f on $\mathbf{S} \cap D_g$. In other words, we compute a lower bound for the optimal value of the problem

$$(P(S)) \left| \begin{array}{ll} \min\limits_{\mathbf{x}} & f(\mathbf{x}) \\ \text{subject to} & g_i(\mathbf{x}) \leq 0 \quad i = 1, \ldots, m\,, \\ \mathbf{x} \in \mathbf{S} \subset \Re^n\,. & \end{array} \right.$$

As mentioned above, f and g_i are generic nonconvex functions belonging to C^2, then the main idea for computing a lower bound $\mu(S)$ is to construct from problem (P(S)) a convex problem by replacing all those nonconvex functions with their respective convex underestimation functions, then solving the resulting relaxed convex problem. In order to achieve this, we see the following definition:

Definition 1 Given any nonconvex function $f(\mathbf{x})\colon S \to \Re$, $\mathbf{x} \in S \subseteq \Re^n$ belonging to C^2, the following quadratic function is defined by

$$F(\mathbf{x}) = \sum_{i=1}^{n} a_i \mathbf{x}_i^2 + \sum_{i=1}^{n} b_i \mathbf{x}_i + c\,, \tag{2}$$

where $\mathbf{x} \in \mathbf{S} \subseteq \Re^n$ and $F(\mathbf{x}) = f(\mathbf{x})$ holds at all vertices of $\mathbf{S}$. The a_i's are nonnegative scalars and are large enough such that $F(\mathbf{x}) \leq f(\mathbf{x})$, $\forall \mathbf{x} \in \mathbf{S}$.

It is trivial to see that $F(\mathbf{x})$ is convex since all quadratic coefficients, i. e., a_i's , are nonnegative. Theorem 2.2.1 in [7] ensures that $F(\mathbf{x})$ defined by Definition 1 is a convex underestimator of $f(\mathbf{x})$ if the difference function between them, i. e., $D(\mathbf{x}) = F(\mathbf{x}) - f(\mathbf{x})$, is a convex function. It is well known that $D(\mathbf{x})$ is convex if and only if its Hessian matrix $H_D(\mathbf{x})$ is positive semidefinite in the current simplex. A useful convexity condition is derived by noting that $H_D(\mathbf{x})$ is related directly to the Hessian matrix $H_f(\mathbf{x})$ of $f(\mathbf{x})$, $\mathbf{x} \in \mathbf{S}$ by the following equation:

$$\mathbf{H}_D(\mathbf{x}) = 2\Delta - \mathbf{H}_f(\mathbf{x})\,,$$

where Δ is a diagonal matrix whose diagonal elements are a_i's defined in Definition 1. Analogous to the "*diagonal shift matrix*" defined in [9], Δ here is referred to as the *diagonal underestimation matrix*, since these parameters guarantee that $F(\mathbf{x})$ defined by Eq. 2 is a rigorous underestimator of the generic nonconvex function $f(\mathbf{x})$. $D(x)$, as defined, is convex if and only if $2\Delta - H_f(\mathbf{x}) = 2\text{diag}(a_i) - H_f(\mathbf{x})$ is positive semidefinite for all $\mathbf{x} \in \mathbf{S}$.

In order to simplify the parameter calculation, the underestimator $F(\mathbf{x})$ is reformulated by using a single nonnegative a value, as follows:

$$F(\mathbf{x}) = a\sum_{i=1}^{n} x_i^2 + \sum_{i=1}^{n} b_i x_i + c\,. \tag{3}$$

Then, all diagonal elements of the diagonal underestimation matrix Δ are therefore equal to the uniform quadratic coefficient a defined by Eq. 3. Some interval arithmetic approaches are provided in [5,7] to estimate the quadratic coefficients with theoretical guarantee in the current simplex.

After the quadratic coefficients have been identified, the linear and constant coefficients of $F(\mathbf{x})$ defined by Eqs. 2 or 3, i. e., b_i's and c, can be given by the quadratic coefficients a_i's and the current simplex. In view of Definition 1, we know $F(\mathbf{x}) = f(\mathbf{x})$ holds at all vertices of $\mathbf{S}$, so the following linear equation group can be obtained as

$$\mathbf{V}^{k^T}\Delta\mathbf{V}^k + \mathbf{b}^T\mathbf{V}^k + c = f\left(\mathbf{V}^k\right) \quad k = 1, \ldots, n+1,$$

where $\Delta \in \Re^{n\times n}$ is the diagonal underestimation matrix whose diagonal elements are the quadratic term coefficients, a_i's defined in Eqs. 2 or 3. $\mathbf{b} \in \Re^n$ is the linear coefficient vector whose elements are b_i's defined in Eqs. 2 or 3, and c is a scalar:

$$\mathbf{b}^T\mathbf{V}^k + c = f\left(\mathbf{V}^k\right) - \mathbf{V}^{k^T}\Delta\mathbf{V} \quad k = 1, \ldots, n+1\,.$$

The vector $\mathbf{b} \in \Re^n$ is augmented as $(\mathbf{b},\, c) \in \Re^{n+1}$, in order to include the scalar c. In the same way, the matrix $\mathbf{V} \in \Re^{(n+1)\times n}$ is augmented as $(\mathbf{V},\, \mathbf{1}) \in \Re^{(n+1)\times(n+1)}$, where $\mathbf{1}$ is a column unity matrix of $\Re^n$. $(\mathbf{V},\, \mathbf{1}) \in \Re^{(n+1)\times(n+1)}$ is a regular square matrix since $\mathbf{V} \in \Re^{(n+1)\times n}$ is the coordinate matrix of the simplex which is linearly independent. Then we have

$$(\mathbf{b},\, c)^T = (\mathbf{V},\, \mathbf{1})^{-1}\left[f(\mathbf{V}) - \mathbf{V}^T\Delta\mathbf{V}\right]\,,$$

where $[f(\mathbf{V}) - \mathbf{V}^T\Delta\mathbf{V}] \in \Re^{n+1}$ is a column vector for the $n+1$ vertices of the current simplex. By virtue of this equation, it is obvious that the linear and constant coefficients defined by Eqs. 2 or 3 are determined uniquely by the quadratic coefficients and the current simplex.

By replacing all the nonconvex functions in problem (P(S)) with their corresponding quadratic function based convex underestimators described by Eq. 3, we have the following relaxed convex programming problem (QP(S)):

$$(QP(S)) \left| \begin{array}{ll} \min\limits_{\mathbf{x}} & F(\mathbf{x}) \\ \text{subject to} & G_i(\mathbf{x}) \le 0 \quad i = 1, \ldots, m\,, \\ & \mathbf{x} \in \mathbf{S} \subset \Re^n\,, \end{array} \right.$$

where

$$F(\mathbf{x}) = \sum_{i=1}^{n} a_i^f x_i^2 + \sum_{i=1}^{n} b_i^f x_i + c^f\,,$$

$$\mathbf{G}_j(\mathbf{x}) = \sum_{i=1}^{n} a_i^{g_j} x_i^2 + \sum_{i=1}^{n} b_i^{g_j} x_i + c^{g_j} \quad j = 1, 2, \ldots, m\,.$$

Let $D_{\mathbf{G}}$ be a subset of $\Re^n$ defined by

$$D_{\mathbf{G}} = \{x \in \Re^n : \mathbf{G}_i(\mathbf{x}) \le 0\,,\ i = 1, 2, \ldots, m\}\,.$$

Obviously, the set $D_{\mathbf{G}}$ is convex and compact. It should be noted that only additional $m+1$ quadratic parameters, i. e., a^f and a^{g_i} for $i = 1, 2, \ldots, m$, are introduced during the above transformation process if the uniform underestimation function is used, since all other linear and constant coefficients can be calculated by those quadratic parameters and the current simplex.

QBB Underestimators for Special Structures

For the concave function structure, denoted by $f^{CA}(\mathbf{x})$, whose eigenvalues are all nonpositive, i. e., $\lambda_{i,\,\mathbf{x}\in\mathbf{S}}(\mathbf{x}) \le 0$. Then, the quadratic coefficient of its underestimator defined by Eq. 2 is zero, so the valid lower bound of the concave function structure over the current simplex is a linear function. In fact, the valid bound constructed by Eq. 2 is equivalent to the convex envelope of the concave function over a simplex [7]. Let $\mathbf{S}$ be a simplex generated by the vertices $\mathbf{V}^1, \mathbf{V}^2, \ldots, \mathbf{V}^{n+1}$, i. e., $\mathbf{S} = \{\mathbf{x} \in \Re^n : \mathbf{x} = \sum_{i=1}^{n+1}\lambda_i\mathbf{V}^i,\ \lambda_i \ge 0,\ \sum_{i=1}^{n+1}\lambda_i = 1\}$, and let $f^{CA}(\mathbf{x})$ be a concave function defined on $\mathbf{S}$. Then the convex envelope of $f^{CA}(\mathbf{x})$ over $\mathbf{S}$ is the affine function $L^{CA}(\mathbf{x}) = \mathbf{b}^T\mathbf{x} + c$ that is uniquely determined by the system of linear equations $f^{CA}(\mathbf{V}^i) = \mathbf{b}^T\mathbf{V}^i + c$ for $i = 1, \ldots, n+1$.

For the general quadratic function presented by

$$f(\mathbf{x}) = \mathbf{x}^T\mathbf{Q}\mathbf{x} + \mathbf{q}^T\mathbf{x}$$

(note $\mathbf{H}_f(\mathbf{x}) = \mathbf{Q}$), we have the *diagonal underestimation matrix*, Δ, constructed on the basis of interval arithmetic [7], as

$$a = \max_i \left\{0,\ \frac{1}{2}\lambda_i^{\mathbf{Q}}\right\}$$

for the uniform case, and for the nonuniform case, we get

$$a_i = \max\left\{0,\ \frac{1}{2}\left(\mathbf{Q}_{ii} + \sum_{j\neq i}\left|\mathbf{Q}_{ij}\right|\right)\right\} .$$

Then, we have the quadratic underestimation function as

$$F(\mathbf{x}) = \mathbf{x}^T\Delta\mathbf{x} + \mathbf{b}^T\mathbf{x} + c\ ,$$

where the linear and constant coefficients, i. e., $(\mathbf{b}, c)$, can be determined uniquely by the quadratic coefficients calculated above and the current simplex.

Properties of the QBB Underestimator

For construction of the QBB underestimator, only quadratic coefficients need to be calculated since the linear and constant ones defined by Eqs. 2 or 3 can be determined uniquely by the quadratic coefficients and the current simplex. Another important property of the QBB algorithm is that the quadratic function based underestimator is always convex throughout the problem space. A potential benefit of this property is that it allows the convex solver applied to get the solution to the underestimator to have a feasible or an infeasible convergence path. Geometrically speaking, the QBB uses a convex quadratic function to approximate the convex part of a general nonconvex function directly, which can bypass the concave parts and avoid the overestimation for them.

Function Decomposition

It should be noted that the relaxed convex programming problem (QP(S)) can contain not only the quadratic underestimation functions for the generic nonconvex terms, but also the convex function terms which are not necessarily transformed into the quadratic underestimators. Then, the final underestimation strategy of the relaxed problem (QP(S)) can be slightly decomposed into the following convex programming formulation, as

$$(\mathrm{QP(S)}')\quad \left|\begin{array}{ll} \min\limits_{\mathbf{x}} & F'(\mathbf{x}) \\ \text{subject to} & G_i'(\mathbf{x}) \le 0 \quad i = 1,\ \ldots\ ,\ m\ , \\ & \mathbf{x}\in\mathbf{S}\subset\Re^n\ , \end{array}\right.$$

where

$$\begin{aligned} F'(\mathbf{x}) &= f^L(\mathbf{x}) + f^C(\mathbf{x}) + L_f^{CA}(\mathbf{x}) + F^{NC}(\mathbf{x})\ , \\ G_i'(\mathbf{x}) &= g_i^L(\mathbf{x}) + g_i^C(\mathbf{x}) + L_{g_i}^{CA}(\mathbf{x}) + G_i^{NC}(\mathbf{x}) \\ & \quad i = 1,\ 2,\ \ldots\ ,\ m\ , \end{aligned}$$

and $f^L(\mathbf{x})$, $f^C(\mathbf{x}), L_f{}^{CA}(\mathbf{x}), g_i{}^L(\mathbf{x}), g_i{}^C(\mathbf{x})$, and $L_{g_i}^{CA}(\mathbf{x})$ represent the linear terms, convex terms, and the linear underestimation functions for the concave terms in the objective function and the constraints, respectively. $F^{NC}(\mathbf{x})$ and $G_i{}^{NC}(\mathbf{x})$ represent the quadratic convex underestimation functions for the generic nonconvex terms. Compared with the relaxed problem (QP(S)), the relaxed problem (QP(S)′) contains not only quadratic function terms, but also the generic convex terms of the original problem.

Algorithmic Procedure of QBB

At the start of this section, problem (P) is formulated over an initial simplex $\mathbf{S}^0$ which can be easily obtained by using an outer approximation approach. Now, we are in a position to present the proposed algorithm for solving problem (P) by using the basic operations described in previous sections.

Step 1 – Initialization. A convergence tolerance, ε_c, and a feasibility tolerance, ε_f, are selected and the iteration counter k is set to be zero. The global lower and upper bounds μ_0 and γ_0 of the global minimum of problem (P) are initialized and an initial current point $\mathbf{x}^{k,c}$ is randomly selected.

Step 2 – Local solution of problem (P) and update of upper bound. The nonconvex and nonlinear optimization problem (P) is solved locally within the current simplex $\mathbf{S}$. If the solution f_{local}^k of problem (P) is ε_f-feasible, the upper bound γ_k is updated as $\gamma_k = \min(\gamma_k,\ f_{\text{local}}^k)$.

Step 3 – Partitioning of the simplex. The current simplex, $\mathbf{S}^k$, is partitioned into the following two sim-

plexes ($r = 1, 2$):

$$\mathbf{S}^{k,1} = \left(\mathbf{V}^{k,0}, \ldots, \mathbf{V}^{k,m}, \ldots, \frac{\mathbf{V}^{k,m} + \mathbf{V}^{k,l}}{2}, \mathbf{V}^{k,n}\right),$$
$$\mathbf{S}^{k,2} = \left(\mathbf{V}^{k,0}, \ldots, \frac{\mathbf{V}^{k,m} + \mathbf{V}^{k,l}}{2}, \ldots, \mathbf{V}^{k,l}, \mathbf{V}^{k,n}\right),$$

where, k,m and k,l correspond to the vertices with the longest edge in the current simplex, i. e., $(k,\ m)$, $(k,\ l) = \arg\max_{i<j} \left\{\left\|\mathbf{V}^{k,j} - \mathbf{V}^{k,i}\right\|\right\}$.

Step 4 – Update of $a^r_{k,f}$, $b^r_{k,f}$, $c^r_{k,f}$ and a^r_{k,g_i}, b^r_{k,g_i}, c^r_{k,g_i} inside both subsimplexes $r = 1,\ 2$. The nonnegative parameters $a^r_{k,f}$ and a^r_{k,g_i} of the general nonconvex terms in the objective function and the constraints are updated inside both simplexes $r = 1,\ 2$ according to the methods presented in former sections, and the corresponding linear and constant coefficients, i. e., $b^r_{k,f}$, $c^r_{k,f}$ and b^r_{k,g_i}, c^r_{k,g_i}, are renewed accordingly.

Step 5 – Solutions inside both subsimplexes $r = 1,\ 2$. The convex programming problem (QP(S)′) is solved inside both subsimplexes ($r = 1,\ 2$) by using some nonlinear programming solver. If a solution $F^{k,r}_{\mathrm{sol}}$ is feasible and less than the current upper bound, γ_k, then it is stored along with the solution point $\mathbf{x}^{k,r}_{\mathrm{sol}}$.

Step 6 – Update iteration counter k and lower bound μ_k. The iteration counter increases by 1,

$$k \leftarrow k + 1\ ,$$

and the lower bound μ_k is updated to the minimum solution over the stored ones from the previous iterations. Furthermore, the selected solution is erased from the stored set:

$$\mu_k = F^{k',r'}_{\mathrm{sol}}\ ,$$

where, $F^{k',r'}_{\mathrm{sol}} = \min_{r,I}\{F^{I,r}_{\mathrm{sol}},\ r = 1,\ 2,\ I = 1, \ldots, k-1\}$. If the set I is empty, set $\mu_k = \gamma_k$ and go to step 8.

Step 7 – Update the current point $\mathbf{x}^{k,c}$ and the current simplex $\mathbf{S}^{k}$. The current point is selected to be the solution point of the previously found minimum solution in step 6,

$$\mathbf{x}^{k,c} = \mathbf{x}^{I',r'}_{\mathrm{sol}}\ ,$$

and the current simplex becomes the subsimplex containing the previously found solution,

$$\mathbf{S}^{k} = \left(\mathbf{V}^{k',0}, \ldots, \mathbf{V}^{k',m}, \ldots, \frac{\mathbf{V}^{k',m} + \mathbf{V}^{k',l}}{2}, \ldots, \mathbf{V}^{k',n}\right), \quad \text{if } r' = 1\ ,$$
$$\mathbf{S}^{k} = \left(\mathbf{V}^{k',0}, \ldots, \frac{\mathbf{V}^{k',m} + \mathbf{V}^{k',l}}{2}, \ldots, \mathbf{V}^{k',l}, \ldots, \mathbf{V}^{k',n}\right), \quad \text{otherwise.}$$

Step 8 – Check for convergence. If $(\gamma_k - \mu_k) > \varepsilon_c$, then return to step 2. Otherwise, ε_c-convergence has been reached. The global minimum solution and the solution point are given as

$$f^* \leftarrow f^{c,k''}\ ,$$
$$\mathbf{x}^* \leftarrow \mathbf{x}^{c,k''}\ ,$$

where, $k'' = \arg_I \left\{f^{c,I} = \gamma_k\right\},\ I = 1, \ldots, k$.

Conclusion

The QBB algorithm is guaranteed to identify the global optimum solution of problems belonging to the broad class of twice-differentiable NLPs. For any such problem, the ability to generate progressively tighter convex lower bounding problems in a branch-and-bound framework guarantees the convergence of this algorithm to within ε of the global optimum solution under the exhaustive simplicial partition of the initial simplex. Different methods [7] have been developed for the construction of the convex valid underestimators for special function structures and the generic nonconvex function structures, where the maximal eigenvalue analysis of the interval Hessian matrix provides the rigorous guarantee for the QBB algorithm to converge to the global solution.

References

1. Grossmann IE (ed) (1996) Global Optimization in Engineering Design, Kluwer book series in nonconvex optimization and its applications. Kluwer, Dordrecht
2. Floudas CA (2000) Deterministic Global Optimization: Theory, Methods and Applications. Kluwer, Dordrecht
3. Zhu Y (1999) Calculation of Phase Equilibria Based on Global Stability Analysis, Ph.D Dissertation. Institute of Process Engineering, Chinese Academy of Sciences, Beijing

4. Zhu Y, Xu Z (1999) Calculation of liquid-liquid equilibrium based on the global stability analysis for ternary mixtures by using a novel branch and bound algorithm: Application to UNIQUAC equation. Ind Eng Chem Res 38:3549–3556
5. Zhu Y, Inoue K (2001) Calculation of chemical and phase equilibrium based on stability analysis by QBB algorithm: Application to NRTL equation. Chem Eng Sci 56:6915–6931
6. Zhu Y, Kuno T (2003) Global optimization of nonconvex MINLP by a hybrid branch-and-bound and revised general Benders decomposition method. Ind Eng Chem Res 42:528–539
7. Zhu Y, Kuno T (2005) A global optimization method, QBB, for twice-differentiable nonconvex optimization problem. J Global Optim 33:435–464
8. Horst R, Tuy H (1990) Global optimization: deterministic approaches. Springer, Berlin
9. Adjiman CS, Dallwig S, Floudas CA, Neumaier A (1998) A global optimization method, aBB, for general twice-differentiable constrained NLPs - I. Theoretical advances. Comput Chem Eng 22:1137–1158
10. Adjiman CS, Androulakis IP, Floudas CA (1998) A global optimization method, aBB, for general twice-differentiable constrained NLPs - II. Implementation and computational results. Comput Chem Eng 22:1159–1179
11. Adjiman CS, Androulakis IP, Floudas CA (2000) Global optimization of mixed integer nonlinear problems. AIChE J 46:1769–1797
12. Akrotirianakis IG, Floudas CA (2004) A new class of improved convex underestimators for twice continuously differentiable constrained NLPs. J Global Optim 30:367–390
13. Akrotirianakis IG, Floudas CA (2004) Computational experience with a new class of convex underestimators: Box constrained NLP problems. J Global Optim 29:249–264
14. Meyer CA, Floudas CA (2005) Convex underestimation of twice continuously differentiable functions by piecewise quadratic perturbation: Spline aBB underestimators. J Global Optim 32:221–258

QR Factorization

Udaya Bhaskar Vemulapati
School of Computer Sci., University Central Florida, Orlando, USA

MSC2000: 65F25, 15A23, 65F05, 65F20, 65F22

Article Outline

Keywords

Orthogonal factorization; QR decomposition; Complete orthogonal factorization

QR factorization is a process of reducing a square (rectangular) matrix into upper triangular (upper trapezoidal) form by applying a series of *elementary orthogonal transformations*.

Properties of Orthogonal Transforms

Orthogonal transforms are where the transformation matrices are orthogonal. Orthogonal matrices are square matrices where each column is a unit vector and each column is mutually orthogonal to every other column. This implies that $Q \in \mathbf{R}^{n\times n}$ is orthogonal if and only if $Q^{\mathsf{T}} Q = QQ^{\mathsf{T}} = I$ (i. e. the transpose of an orthogonal matrix is its inverse). Orthogonal transformations are invariant under the 2-norm; i. e. $\| Qx \|_2 = \| x \|_2$. More details can be found in [8]. There are two popular orthogonal transformations: Householder and Givens.

Householder Transformations

These are named after A.S. Householder, who popularized their use in matrix computations. However, the properties of these matrices have been known for quite some time. For any nonzero $v \in \mathbf{R}^n$, a matrix H of the form

$$H = I - 2\frac{vv^{\top}}{v^{\top}v}$$

is called *Householder transformation*. It is easy to verify that H is symmetric, and orthogonal (which also means that it is its own inverse). Identity matrices are not Householder matrices. Geometrically, Householder matrices merely rotate a given vector in n-dimensions (without stretching or shrinking). Given any two vectors x and y such that $\| x \|_2 = \| y \|_2$, there exists

a Householder transformation H such that $Hx = y$ (it is easy to verify that $v = y - x$ satisfies this equation). Note that v completely characterizes H (in the sense that even though H is an $n \times n$ matrix, v is enough to reconstruct H, and to apply H). Also, scaling v by a scalar factor α will not change the transformation H.

QR Factorization Using Householder Transformations

Since Householder transformations rotate vectors in n-dimensions, they can be used to introduce zeroes selectively. Specifically, given any vector $x \neq 0 \in \mathbf{R}^n$, one can construct a Householder matrix H such that Hx is a multiple of e_1 (the first column of the identity matrix), i. e. make everything except the first row of Hx zero. Geometrically, this amounts rotating the vector such that it is parallel to the principal axis. It is easy to see that such H has the form $H = \mathrm{I} - 2vv^{\mathsf{T}}\ v^{\mathsf{T}}\ v$ where $v = x \pm \alpha\ e_1$ and $\alpha = \| \mathrm{x} \|_2$. In order to avoid subtracting close numbers (while dealing with floating point arithmetic), v is often chosen as $v = x + \mathrm{sign}(\xi_1)\ \alpha\ e_1$, where ξ_1 is the first element of x.

The following function House will compute the vector v, given x, that characterizes H so that $H = I - 2vv^{\mathsf{T}}/v^{\mathsf{T}}v$ and that $Hx = -\alpha\ e_1$. Also, v is scaled such that $v(1) = 1$, as the scaling does not affect H (using a notation similar to MATLAB [5]).

To apply H to a vector y, note that

$$Hy = \left(I - 2\frac{vv^{\top}}{v^{\top}v}\right) y = y - 2\frac{v(v^{\top}y)}{v^{\top}v} = y - 2\frac{v^{\top}y}{v^{\top}v}v,$$

and hence, one can compute Hy without explicitly computing H. The same idea can be extended to applying H to a set of columns $C \in \mathbf{R}^{n \times k}$. Let us call that function row. House(v, C).

```
function: v = House(x)
    n = length(x);
    v(1) = x(1) + sign(x(1)) * norm(x, 2);
    v(2 : n) = x(2 : n)/v(1);
    v(1) = 1;
end;
```

Suppose $H_1 = \mathrm{House}(x)$ with x taken as the first column of a matrix $A \in \mathbf{R}^{m \times n}$. Then H_1A will have zeros on the first column below the first row. Then one can find $H_2' = \mathrm{House}(A(2 : m, 2))$ such that everything below the second row of the second column is zeroed. Effectively, applying H_2' to the lower $(m-1) \times (n-1)$ matrix is the same as applying

$$H_2 = \begin{array}{c} \\ 1 \\ m-1 \end{array} \begin{array}{c} \begin{array}{cc} 1 & m-1 \end{array} \\ \begin{pmatrix} 1 & 0 \\ 0 & H_2' \end{pmatrix} \end{array}$$

to A. Note that H_2 does not affect the first row and column of H_1A. If this process is continued by applying a sequence of Householder transformations to A, it is reduced to an upper-trapezoidal matrix R; i. e.

$$H_{n-1}H_{n-2}\cdots H_1A = R\,. \tag{1}$$

Since each H_i is orthogonal, the product $Q^{\mathsf{T}} = H_{n-1}\ H_{n-2}\cdots\ H_1$ is also orthogonal. Then rearranging the equation,

$$A = QR\,, \tag{2}$$

where Q is orthogonal and R is upper-trapezoidal. This form of factorization is called *QR factorization* (or *orthogonal factorization*). The following algorithm computes the QR factorization of a matrix A.

```
function: QR(A, m, n)
    for i = 1 : min(m, n),
      v = House(A(i : m, i));
      A(i + 1 : m, i) = v(2 : m − i + 1);
      A(i+1 : m, i+1 : n) = row.House(v, A(+1 : m,
          i + 1 : n));
    end;
end;
```

In the above algorithm, the essential components of the Householder vectors are stored right where the zeros are going to be introduced. Here are the different parts of a matrix A after the algorithm is applied (superscripts indicate how many times an entry has been modified):

$$\begin{pmatrix} a_{11}^1 & a_{12}^1 & a_{13}^1 & \cdots & a_{1n}^1 \\ v_{21} & a_{22}^2 & a_{23}^2 & \cdots & a_{2n}^2 \\ v_{31} & v_{32} & a_{33}^3 & \cdots & a_{3n}^3 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ v_{m1} & v_{m2} & v_{m3} & \cdots & v_{mn} \end{pmatrix}.$$

v_{ij} is the ith component of vector v produced by the above algorithm during the jth iteration. Note that the matrix Q is available in in the lower triangular portion of the matrix in a factored form.

Givens Rotations

These rotations are named after W. Givens; they are also referred to as *Jacobi iterations*. C.G. Jacobi devised a symmetric eigenvalue algorithm based on these transformations in 1846. Consider the following 2×2 matrix of the form

$$G(\theta) = \begin{pmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{pmatrix}$$

applied to a vector $x \in \mathbf{R}^2$. It is easy to see that $G^\mathsf{T}x$ is a mere rotation of x by an angle of θ in counterclockwise direction. Such transformations are called rotations and as such are orthogonal. A straightforward extension that applies to an n-vector is given by matrices of the following form

$$G(i,j,\theta) = \begin{array}{c} \\ i \\ \\ j \\ \\ \end{array} \begin{array}{c} \begin{array}{ccccccc} & & i & & j & & \end{array} \\ \begin{pmatrix} 1 & \cdots & 0 & \cdots & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & & \vdots & & \vdots \\ 0 & \cdots & c & \cdots & s & \cdots & 0 \\ \vdots & & \vdots & \ddots & \vdots & & \vdots \\ 0 & \cdots & -s & \cdots & c & \cdots & 0 \\ \vdots & & \vdots & & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & \cdots & 0 & \cdots & 1 \end{pmatrix} \end{array}$$

with $c^2 + s^2 = 1$. Here $G^\mathsf{T}(i,j,\theta)\,x$ is a rotation of $x \in \mathbf{R}^n$ by an angle θ in counterclockwise direction in the (i,j)-plane. It is easy to verify that $G^\mathsf{T}(i,j,\theta)$ only modifies the rows i, j of the vector that is applied to and the remaining entries are unaffected; i. e.

$$G^\top(i,j,\theta)x = \begin{cases} cx_i - sx_j & i\text{th component}, \\ sx_i + cx_j & j\text{th component}, \\ \text{unchanged} & \text{otherwise}. \end{cases}$$

Given any vector $x \in \mathbf{R}^n$, $G^\mathsf{T}(i,j,\theta)$ can be constructed such that only the rows i, j are affected and that x_j is zeroed. Solving the following equations

$$sx_i + cx_j = 0 \quad \text{and} \quad c^2 + s^2 = 1$$

will yield

$$c = \frac{x_i}{\sqrt{x_i^2 + x_j^2}}, \quad s = \frac{-x_j}{\sqrt{x_i^2 + x_j^2}}. \tag{3}$$

Let $G_{ij}^{(k)}$ denote the application of a Givens rotation that uses rows i and j and zeros A_{jk} entry. The first column below the first row can be zeroed using a sequence of Givens rotations such as

$$Q_1 = G_{1m}^{(1)} \cdots G_{12}^{(1)}.$$

Similarly, the second column can be zeroed below the diagonal by

$$Q_2 = G_{2m}^{(2)} \cdots G_{23}^{(2)}.$$

Repeating this process for each column, A is reduced to upper-trapezoidal form, as in

$$Q_n \cdots Q_1 A = R.$$

The beauty about using Givens rotations is that there are various ways of applying these rotations and yet getting the same final QR factorization. In fact, this fact can be exploited in parallel processing very effectively. Detailed parallel QR factorization algorithms can be found in [6,7] and [3].

Fast Givens Transformations

Fast Givens Transformations involve half the number of multiplications compared to Givens rotations and they can be used to zero without an explicit square root computation. They are also referred to as *square-root-free Givens transformations*. Details can be found in [1,2,4].

Finally, it can be shown that if A has full rank, then it has a unique QR factorization if we make the diagonal elements of R positive [8].

See also

- ABS Algorithms for Linear Equations and Linear Least Squares
- Cholesky Factorization
- Interval Linear Systems
- Large Scale Trust Region Problems
- Large Scale Unconstrained Optimization
- Linear Programming
- Orthogonal Triangularization

- ► Overdetermined Systems of Linear Equations
- ► Solving Large Scale and Sparse Semidefinite Programs
- ► Symmetric Systems of Linear Equations

References

1. Anda AA, Park H (1994) Fast plane rotations with dynamic scaling. SIAM J Matrix Anal Appl 15:162–174
2. Gentleman WM (1973) Least squares computations by Givens rotations without square roots. J Inst Math Appl 12:329–336
3. Golub GH, Van Loan CF (1997) Matrix computations. third Johns Hopkins Univ. Press, Baltimore
4. Hammarling S (1974) A note on modifications to the Givens plane rotation. J Inst Math Appl 13:215–218
5. Hanselman D, Littlefield B (1997) The student edition of MATLAB 5 user's guide. MathWorks, Orchard Hill Place
6. Pothen A, Jha S, Vemulapati U (1987) Orthogonal factorization on the hypercube. In: Heath MT (ed) Hypercube Multiprocessors 1987. SIAM, Philadelphia, pp 587–596
7. Sameh AH, Kuck DJ (1978) On stable parallel linear system solvers. J ACM 25(1):81–91
8. Stewart GW (1973) Introduction to matrix computations. Acad. Press, New York

Quadratic Assignment Problem

QAP

LEONIDAS PITSOULIS[1], PANOS M. PARDALOS[2]

[1] Princeton University, Princeton, USA

[2] Center for Applied Optim. Department Industrial and Systems Engineering, University Florida, Gainesville, USA

MSC2000: 90C08, 90C11, 90C27, 90C57, 90C59

Article Outline

Keywords

Combinatorial optimization; Quadratic assignment problem; Nonlinear assignment problems; Facility location

The quadratic assignment problem (QAP) is a combinatorial optimization problem, that although there is a substantial amount of research devoted to it, it is still, up to this date, not well solvable in the sense that no exact algorithm can solve problems of size $n > 20$ in reasonable computational time. The QAP can be viewed as a natural extension of the linear assignment problem (LAP; cf. also ► Assignment and matching). Let $\mathcal{S}_n$ denote the set of all permutations $\phi{:}N \to N$, where $N = \{1, \ldots, n\} \in \mathbf{Z}^+$. Given a cost matrix $C = (c_{ij}) \in \mathbf{R}^{n \times n}$ we can formulate the LAP using permutations as:

$$\min_{\phi \in S_n} \sum_{i=1}^{n} \sum_{j=1}^{n} c_{\phi(i)\phi(j)} = \min_{\phi \in S_n} \sum_{i=1}^{n} c_{i\phi(i)} \,. \tag{1}$$

The general formulation of the QAP as introduced by E.L. Lawler in [88] is obtained by increasing the dimension of the cost array C:

$$\min_{\phi\in S_n} \sum_{i=1}^{n}\sum_{j=1}^{n}\sum_{k=1}^{n}\sum_{l=1}^{n} c_{\phi(i)\phi(j)\phi(k)\phi(l)} = \min_{\phi\in S_n} \sum_{i=1}^{n}\sum_{j=1}^{n} c_{ij\phi(i)\phi(j)} \,. \quad (2)$$

Formulation (2) will be referred to as the general QAP, while an instance will be denoted by QAP(C). The most widely used formulation of the QAP, and its first appearance in the literature, is that of T.C. Koopmans and M.J. Beckmann [85] which is a special case of (2). Used as a mathematical model for the location of a set of indivisible economical activities, the formulation of Koopmans and Beckmann involves three $n \times n$ input matrices with real elements $F = (f_{ij})$, $D = (d_{kl})$ and $B = (b_{ik})$, where f_{ij} is the flow between the facility i and facility j, d_{kl} is the distance between the location k and location l, and b_{ik} is the cost of placing facility i at location k. The objective is to assign each facility to a location such that the total cost is minimized. The *Koopmans–Beckmann* QAP formulation is given as follows:

$$\min_{\phi\in S_n} \sum_{i=1}^{n}\sum_{j=1}^{n} f_{ij}d_{\phi(i)\phi(j)} + \sum_{i=1}^{n} b_{i\phi(i)} \,. \quad (3)$$

In the context of facility location (cf. also ► Facilities layout problems) the matrices F and D are symmetric with zeros in the diagonal, and all the matrices are nonnegative. An instance of a QAP with input matrices F, D and B will be denoted by QAP(F, D, B), while we will denote an instance by QAP(F, D), if there is no linear term (i. e., $B = 0$). It can be seen that (3) is a special case of (2) by setting $c_{ijkl} = f_{ij}d_{kl}$ for all i, j, k, l with $i \neq j$ or $k \neq l$ and $c_{ikik} = f_{ii}d_{kk} + b_{ik}$, otherwise. In terms of computational complexity (cf. also ► Complexity theory; ► Computational complexity theory), S. Sahni and T. Gonzalez [129] have shown that the QAP is *NP*-hard and that even finding an approximate solution within some constant factor from the optimal solution cannot be done in polynomial time unless $P = NP$.

Formulations

The QAP can be formulated as the following 0-1 integer programming problem with quadratic objective function (hence the name 'quadratic assignment problem'):

$$\begin{cases} \min & \sum\limits_{\substack{i,j=1\\ i\neq k\, j\neq l}}^{n}\sum\limits_{k,l=1}^{n} c_{ijkl}x_{ik}x_{jl} + \sum\limits_{i,j=1}^{n} c_{ijij}x_{ij} \\ \text{s.t.} & \sum\limits_{i=1}^{n} x_{ij} = 1, \quad j = 1,\ldots,n, \\ & \sum\limits_{j=1}^{n} x_{ij} = 1, \quad i = 1,\ldots,n, \\ & x_{ij} \in \{0,1\}, \quad i,j = 1,\ldots,n. \end{cases} \quad (4)$$

The above formulation is a direct consequence of formulation (2), where the constraints imposed by the permutations are expressed algebraically. A QAP in Koopmans–Beckmann form can be formulated in a more compact way using the inner product between two matrices:

$$\begin{cases} \min & \langle F, XDX^{\top}\rangle + \langle B, X\rangle \\ \text{s.t.} & X \in \mathbf{X}_n, \end{cases} \quad (5)$$

where $\mathbf{X}_n$ is the set of all *permutation matrices* $X = (x_{ij})$ such that their elements satisfy the constraints in (4). In the objective function of (4), let the coefficients c_{ijkl} be the entries of an $n^2 \times n^2$ matrix S, such that c_{ijkl} is on row $(i-1)\,n + k$ and column $(j-1)n + l$. Now let $Q := S - \alpha I$, where I is the $(n^2 \times n^2)$ unit matrix and α is greater than the row norm $\|S\|_\infty$ of matrix S. The subtraction of a constant from the entries on the main diagonal of S does not change the optimal solutions of the corresponding QAP, it simply adds a constant to the objective function. Hence we can consider a QAP with coefficient array Q instead of S. Let $x = (x_{11}, \ldots, x_{1n}, x_{21}, \ldots, x_{nn})^{\mathsf{T}} = (x_1, \ldots, x_{nn})^{\mathsf{T}}$. Then we can rewrite the objective function of the QAP with array of coefficients Q as a quadratic form $x^{\mathsf{T}}\,Qx$, where it can be shown that Q is symmetric and negative definite. Therefore we have a quadratic concave minimization problem (cf. also ► Concave programming) and can formulate the

QAP as:

$$\begin{cases} \min & x^\top Q x \\ \text{s.t.} & \sum_{i=1}^{n} x_{ij} = 1, \quad j = 1, \ldots, n, \\ & \sum_{j=1}^{n} x_{ij} = 1, \quad i = 1, \ldots, n, \\ & x_{ij} \geq 0, \quad i, j = 1, \ldots, n, \end{cases} \tag{6}$$

The above formulation was introduced in [14], and was used to derive cutting plane procedures (cf. also ► Integer programming: Cutting plane algorithms). By adding the term αI to the matrix Q instead of subtracting it, we could always assume that the objective function of the QAP is convex. This leads to the formulation of the QAP as a quadratic convex minimization problem. The QAP can also be formulated using the trace of a matrix as:

$$\begin{cases} \min & \operatorname{tr}(FXD^\top + B)X^\top \\ \text{s.t.} & X \in \mathbf{X}_n. \end{cases} \tag{7}$$

The trace formulation of the QAP first appeared in [47], and was used in [51] to introduce eigenvalue lower bounding techniques for symmetric QAPs.

Let $\operatorname{vec}(X) \in \mathbf{R}^{n^2}$ be the vector formed by the columns of a permutation matrix X. The QAP can be formulated using the Kronecker product as

$$\begin{cases} \min & \operatorname{vec}(X)^\top (F \otimes D) \operatorname{vec}(X) \\ & \quad + \operatorname{vec}(B)^\top \operatorname{vec}(X) \\ \text{s.t.} & X \in \mathbf{X}_n. \end{cases} \tag{8}$$

Using the Kronecker product, Lawler [88] provided an alternative formulation for the QAP as an $n^2 \times n^2$ LAP. An $n^2 \times n^2$ matrix C is constructed from the n^4 costs c_{ijkl}, such that the $(ijkl)$th element corresponds to the $((i-1)\,n + k, (j-1)\,n + \mathrm{l})$th element of C. The QAP then is equivalent to an LAP of dimension n^2 with C as the cost matrix, and with the additional constraint that the $n^2 \times n^2$ permutation matrix which defines a feasible solution, must be the Kronecker product of two permutation matrices of dimension n. In other words the QAP is equivalent to

$$\begin{cases} \min & \langle C, Y \rangle \\ \text{s.t.} & Y = X \otimes X, \\ & X \in \mathbf{X}_n. \end{cases} \tag{9}$$

The resulting LAP however cannot be solved efficiently (i. e., in $O(n^6)$ time) because Y, although it is an $n^2 \times n^2$ permutation matrix, is constrained to have a special structure.

Linearizations

Linearization is a technique which involves the elimination of the nonlinear term in a given objective function, in order to make it linear, through the introduction of new variables and new linear (binary) constraints. The objective is to transform a 0–1 nonlinear integer program into a provably equivalent 0–1 linear integer program, such that existing methods for linear integer programs will provide a relaxed problem where lower bounds may be computed. Though there are several ways to linearize a given nonlinear integer program, it is desirable to have a linearization that will introduce the least amount of new variables and constraints. Moreover, the 'tightness' of the relaxation of the resulting linear integer program is very important.

The first attempt to devise solution techniques for solving the QAP had to do with the elimination of the quadratic term in the objective function of (4), in order to transform the problem into a 0–1 linear program. Four such linearizations of the QAP will be presented in this section. The first is due to Lawler [88], which is the first linearization suggested for the QAP, and the second by Kaufman and F. Broeckx [80] which is the smallest with regard to the number of new variables and constraints introduced. The third is a more recent (1983) one that is due to A.M. Frieze and J. Yadegar [56], which unifies most of the previous linearizations for the QAP, and is closely related to the fourth linearization presented in this section due to W.P. Adams and T.A. Johnson [2].

Lawler's Linearization

Lawler [88] replaces the quadratic terms $x_{ij}x_{kl}$ in the objective function of (4), with n^4 variables

$$y_{ijkl} := x_{ij}x_{kl}\,, \quad i, j, k, l = 1, \ldots, n\,,$$

which results in a 0–1 linear program of $n^4 + n^2$ binary variables and $n^4 + 2n^2 + 1$ constraints. More specifically, it is proved in [88] that the QAP is equivalent to the

following 0–1 linear program

$$\begin{cases} \min & \sum_{i,j=1}^{n}\sum_{k,l=1}^{n} c_{ijkl}\, y_{ijkl} \\ \text{s.t.} & (x_{ij}) \in \mathbf{X}_n, \\ & \sum_{i,j=1}^{n}\sum_{k,l=1}^{n} y_{ijkl} = n^2, \\ & x_{ij} + x_{kl} - 2y_{ijkl} \geq 0, \\ & y_{ijkl} \in \{0,1\}, \\ & \text{for } i,j,k,l = 1,\ldots,n. \end{cases}$$

Kaufman–Broeckx Linearization

Rearranging terms in the objective function (4) we obtain

$$\sum_{i,j=1}^{n} x_{ij} \sum_{k,l=1}^{n} c_{ijkl} x_{kl} \,.$$

Kaufman and Broeckx [80] defined n^2 new real variables

$$w_{ij} := x_{ij} \sum_{k,l=1}^{n} c_{ijkl} x_{kl} \,, \quad i,j = 1,\ldots,n \,,$$

resulting in an equivalent linear objective function

$$\sum_{i,j=1}^{n} w_{ij} \,.$$

Introducing n^2 constants $a_{ij} := \sum_{k,l=1}^{n} c_{ijkl}$ for $i, j = 1, \ldots, n$, the QAP becomes equivalent to the following mixed 0–1 linear program:

$$\begin{cases} \min & \sum_{i,j=1}^{n} w_{ij} \\ \text{s.t.} & (x_{ij}) \in \mathbf{X}_n, \\ & a_{ij} x_{ij} + \sum_{k,l=1}^{n} c_{ijkl} x_{kl} - w_{ij} \leq a_{ij}, \\ & w_{ij} \geq 0, \\ & i,j = 1,\ldots,n. \end{cases}$$

The above formulation employs n^2 new real variables, n^2 binary variables and $n^2 + 2n$ constraints. The elements c_{ijkl} are all assumed to be nonnegative, which is a valid assumption since the addition of a constant to each element will not affect the optimal solution. The proof of equivalence of the QAP to the above linear integer program can be found in [80].

Frieze–Yadegar Linearization

In [56] the products of the binary variables are replaced by continuous variables (i. e. $y_{ijkl} := x_{ij}x_{kl}$), and the QAP(C) is proved to be equivalent to the following mixed 0–1 linear program:

$$\begin{cases} \min & \sum_{i,j=1}^{n}\sum_{k,l=1}^{n} c_{ijkl}\, y_{ijkl} \\ \text{s.t.} & (x_{ij}) \in \mathbf{X}_n, \\ & \sum_{i=1}^{n} y_{ijkl} = x_{kl}, \quad \forall j,k,l, \\ & \sum_{j=1}^{n} y_{ijkl} = x_{kl}, \quad \forall i,k,l, \\ & \sum_{k=1}^{n} y_{ijkl} = x_{ij}, \quad \forall i,j,l, \\ & \sum_{l=1}^{n} y_{ijkl} = x_{ij}, \quad \forall i,j,k, \\ & y_{ijij} = x_{ij}, \quad \forall i,j, \\ & 0 \geq y_{ijkl} \leq 1, \quad \forall i,j,k,l, \end{cases} \tag{10}$$

where $i, j, k, l = 1, \ldots, n$. The above program has n^4 new real variables, n^2 binary variables, and $n^4 + 4n^3 + n^2 + 2n$ constraints. Note that the constraint $y_{ijij} = x_{ij}$ is redundant since it follows from the definition of the y_{ijkl} variables. Frieze and Yadegar considered a Lagrangian relaxation of the above 0–1 linear program, and established a relationship between the lower bounds derived by the solution of the relaxation, and the lower bounds derived from decomposition techniques applied to the Gilmore–Lawler bound for the QAP.

Adams–Johnson Linearization

Adams and Johnson presented in [2] a new 0–1 linear integer formulation for the QAP, which resembles the one of Frieze and Yadegar described previously. It is based on the general linearization technique for general 0–1 polynomial programs introduced by Adams and

H.D. Sherali [3,4]. The QAP(C) is proved to be equivalent to the following mixed 0–1 linear program:

$$\begin{cases} \min & \sum_{i,j=1}^{n} \sum_{k,l=1}^{n} c_{ijkl} y_{ijkl} \\ \text{s.t.} & (x_{ij}) \in \mathbf{X}_n, \\ & \sum_{i=1}^{n} y_{ijkl} = x_{kl}, \quad \forall j,k,l, \\ & \sum_{j=1}^{n} y_{ijkl} = x_{kl}, \quad \forall i,k,l, \\ & y_{ijkl} = y_{klij}, \quad \forall i,j,k,l, \\ & y_{ijkl} \geq 0, \quad \forall i,j,k,l, \end{cases} \tag{11}$$

where $i, j, k, l = 1, \ldots, n$, and each y_{ijkl} represents the product $x_{ij}x_{kl}$. The above formulation contains n^2 binary variables x_{ij}, n^4 continuous variables y_{ijkl}, and $n^4 + 2n^3 + 2n$ constraints excluding the nonnegativity constraints on the continuous variables. Although as noted in [3] a significant smaller formulation in terms of both the number of variables and constraints could be obtained, the structure of the relaxation of the above formulation is favorable for solving it. As noted in [2], the constraint set of the above relaxation describes a solution matrix Y which is the Kronecker product of two permutation matrices (i. e. $Y = X \otimes X$ where $X \in \mathbf{X}_n$), showing clearly the equivalence of the above formulation with the QAP as formulated in (9). The theoretical strength of the above linearization of the QAP lies on the fact that, as shown in [2] and [73], the constraints of the relaxations derived from all previous linearizations, can be expressed as a linear combination of the constraints of the continuous relaxation of the above linearization. Moreover, many of the previously published lower-bounding techniques, can be explained based on the dual-space of this relaxation.

Complexity Issues

The first two parts of this section bring evidence to the fact that the QAP is a 'very hard' problem from the theoretical point of view. Not only the QAP cannot be solved to optimality efficiently but it even cannot be approximated efficiently within some constant approximation ratio. Furthermore, finding local optima is not a trivial task even for simply structured neighborhoods like the 2-opt neighborhood. The asymptotic behavior of the QAP and polynomially solvable special cases of the QAP are mentioned in the last two parts of this section.

Computational Complexity

Two early results obtained by Sahni and Gonzalez [129] in 1976 settled the complexity of solving and approximating the QAP. It was shown that the QAP is *NP*-hard and that even finding an ϵ-approximate solution for the QAP is a hard problem, in the sense that the existence of a polynomial ϵ-approximation algorithm implies $P = NP$.

Theorem 1 [129] *The quadratic assignment problem is strongly NP-hard. For an arbitrary $\epsilon > 0$, the existence of a polynomial time ϵ-approximation algorithm for the QAP implies $P = NP$.*

The proof is done by a reduction from the Hamiltonian cycle problem [58].

M. Queyranne [121] derives an even stronger result which further confirms the widely spread belief on the inherent difficulty of the QAP in comparison with other difficult combinatorial optimization problems. It is well known and very easy to see that the traveling salesman problem (TSP) is a special case of the QAP. The TSP on n cities can be formulated as a QAP(F, D) where F is the distance matrix of the TSP instance and D is the adjacency matrix of a Hamiltonian cycle on n vertices. In the case that the distance matrix is symmetric and satisfies the triangle inequality, the TSP is approximable in polynomial time within 3/2 as shown in [37]. Queyranne [121] showed that, unless $P = NP$, QAP(A, B) is not approximable in polynomial time within some finite approximation ratio, even if A is the distance matrix of some set of points on a line and B is a symmetric block diagonal matrix.

A more recent result of S. Arora, Frieze and H. Kaplan [6] answers partially one of the open questions stated by Queyranne [121]. What happens if matrix A is the distance matrix of n points which are regularly spaced on a line, that is, points with abscissae given by $x_p = p$, $p = 1, \ldots, n$? This special case of the QAP is termed *linear arrangement problem* and is a well studied *NP*-hard problem. In the linear arrangement problem the matrix B is not restricted to have the block diagonal structure mentioned above, but is simply a symmetric 0–1 matrix. Arora, Frieze and Kaplan [6] give

a polynomial time approximation scheme (PTAS) for the linear arrangement problem in the case that the 0–1 matrix B is dense, that is, the number of '1' entries in B is in $\Omega(n^2)$, where n is the size of the problem. They show that for each $\epsilon > 0$ there exists an ϵ-approximation algorithm for the dense linear arrangement problem with time complexity depending polynomially on n and exponentially on $1/\epsilon$, hence polynomial for each fixed $\epsilon > 0$.

PLS-Complexity

Assume that an optimization problem P is given by specifying a ground set $\mathcal{E}$, a set $\mathcal{F} \subseteq 2^{\mathcal{E}}$ of feasible solutions and an objective function $f : \mathcal{F} \to \mathbf{R}$. A globally optimal solution $S^* \in \mathcal{F}$ of the problem P is defined as:

$$f(S^*) := \min_{S \in \mathcal{F}} f(S) .$$

For any given $S \in \mathcal{F}$ denote the *neighborhood* of S by $\mathcal{N}(S) \subset \mathcal{F}$. The neighborhood of S consists of other feasible solutions which are topologically 'close' to S. A *locally optimal solution* or a *local minimum* $\overline{S} \in \mathcal{F}$ of the problem P, given the neighborhood $\mathcal{N}$ is defined as:

$$f(\overline{S}) = \min_{S \in \mathcal{N}(\overline{S})} f(S) .$$

Recently (as of 1999) it has been shown that even finding a locally optimal solution for the QAP can be prohibitively hard, that is, even local search is hard in the case of the QAP. Consider the following question 'How easy it is to find a locally optimal solution for the QAP?' Since local optimality is defined through a specific neighborhood structure, the answer depends on the involved neighborhood structure. If the neighborhood $\mathcal{N}$ is replaced by new neighborhood $\mathcal{N}'$, one would generally expect changes in the local optimality status of a solution. The theoretical basis for facing this kind of problems was introduced by D.S. Johnson, C.H. Papadimitriou and Yannakakis [72]. They define the so-called *polynomial time local search problems*, shortly *PLS problems*. A pair $(P, \mathcal{N})$, where P is a (combinatorial) optimization problem and $\mathcal{N}$ is an associated well defined neighborhood structure, defines a local search problem in which the objective is to find a locally optimal solution of P with respect to the neighborhood structure $\mathcal{N}$. Without going into technical details a problem in the PLS class is a local search problem for which local optimality can be checked in polynomial time. In analogy with decision problems, there exist complete problems in the class of PLS problems. The PLS-complete problems, are – in the usual complexity sense – the most difficult among the PLS problems.

K.A. Murthy, Pardalos and Y. Li [103] introduce a neighborhood structure for the QAP which is similar to the neighborhood structure proposed by B.W. Kernighan and S. Lin [81] for the graph partitioning problem. For this reason we will call it a *K-L type neighborhood structure for the QAP*. Murthy, Pardalos and Li [103] show that the corresponding local search problem is PLS-complete. Consider a permutation $\phi_0 \in \mathcal{S}_n$. A swap of ϕ_0 is a permutation $\phi \in \mathcal{S}_n$ obtained from ϕ_0 by applying a transposition (i, j) to it, $\phi = \phi_0 \circ (i, j)$. A *transposition* (i, j) is defined as a permutation which maps i to j, j to i, and k to k for all $k \notin \{i, j\}$. A *greedy swap* of permutation ϕ_0 is a swap ϕ which maximizes $Z(F, D, \phi_0) - Z(F, D, \phi)$ over all swaps ϕ of ϕ_0, where $Z(F, D, \phi)$ is the objective function value of QAP(F, D) with permutation ϕ (see formulation (3)). Let $\phi_0, \ldots, \phi_l$ be a sequence of permutations in $\mathcal{S}_n$, each of them being a greedy swap of the preceding one. Such a sequence is called *monotone* if for all $k = 0, \ldots, l$, in the pair (ϕ_k, ϕ_{k+1}) where $\phi_k = \phi_{k-1} \circ (i_k, j_k)$ and $\phi_{k+1} = \phi_k \circ (i_{k+1}, j_{k+1})$, we have $\{i_k, j_k\} \cap \{i_{k+1}, j_{k+1}\} = \emptyset$. The *neighborhood* of ϕ_0 consists of all permutations which occur in the (unique) maximal monotone sequence of greedy swaps starting with permutation ϕ_0. Let us denote this neighborhood structure for the QAP by $\mathcal{N}_{K-L}$. It is not difficult to see that, given a QAP(F, D) of size n and a permutation $\phi \in \mathcal{S}_n$, the cardinality of $\mathcal{N}_{K-L}(\phi)$ does not exceed $\lfloor n/2 \rfloor + 1$. It is easily seen that the local search problem (QAP, $\mathcal{N}_{K-L}$) is a PLS problem. This result can be found in [112], where the authors show that the graph partitioning problem with the neighborhood structure defined in [81] is PLS-reducible to (QAP, $\mathcal{N}_{K-L}$).

Theorem 1 [112] *The local search problem (QAP, $\mathcal{N}_{K-L}$), where $\mathcal{N}_{K-L}$ is the Kernighan–Lin type neighborhood structure for the QAP, is PLS-complete.*

The PLS-completeness of (QAP, $\mathcal{N}_{K-L}$) implies that, in the worst case, a general local search algorithm as described above involving the Kernighan–Lin type neighborhood finds a local minimum only after a time which is exponential on the problem size. Numerical results,

however, show that such local search algorithms perform quite well when applied to QAP test instances, as reported in [103].

Another simple and frequently used neighborhood structure is the so-called *pair-exchange* (or *2-opt*) neighborhood $\mathcal{N}_2$. The pair-exchange neighborhood of a permutation $\phi_0 \in \mathcal{S}_n$ consists of all permutations $\phi \in \mathcal{S}_n$ obtained from ϕ_0 by applying some transposition (i, j) to it. Specifically,

$$\mathcal{N}_2(\phi) := \{\phi \circ (i, j):\ i, j = 1, \ldots, n,\ i \neq j\} \ .$$

It can also be shown that (QAP, $\mathcal{N}_2$) is PLS-complete. A.A. Schräffer and Yannakakis [130] have proven that the graph partitioning problem with a neighborhood structure analogous to $\mathcal{N}_2$ is PLS-complete. A similar PLS-reduction as in [112] implies that the local search problem (QAP, $\mathcal{N}_2$) is PLS-complete.

Finally, let us mention that no local criteria are known for deciding how good a locally optimal solution is as compared to a global one. From the complexity point of view, deciding whether a given local optimum is a globally optimal solution to a given instance of the QAP is a hard problem, see [108].

Asymptotic Behavior

Under certain probabilistic conditions on the coefficient matrices of the QAP, the ratio between its 'best' and 'worst' values of the objective function approaches 1, as the size of the problem approaches infinity. R.E. Burkard and U. Fincke [29] identify a common combinatorial property of a number of problems which, under certain probabilistic conditions on the problem data, behave as described above.

In an early work Burkard and Fincke [28] investigate the relative difference between the worst and the best values of the objective function for the Koopmans–Beckmann QAP. They first consider the case where the coefficient matrix D is the matrix of pairwise distances of points chosen independently and uniformly from the unit square in the plane. Then the general case where entries of the flow and distance matrices F and D are independent random variables taken from a uniform distribution on [0, 1] is considered. In both cases it is shown that the relative difference mentioned above approaches 0 with probability tending to 1 as the size of the problem tends to infinity.

Later Burkard and Fincke [29] consider the ratio between the objective function values corresponding to an optimal (or best) and a worst solution of a generic combinatorial optimization problem. They find that for each $\epsilon > 0$, the ratio between the best and the worst values of the objective function lies on $(1 - \epsilon, 1 + \epsilon)$, with probability tending to 1, as the size of the problem approaches infinity. Under additional combinatorial conditions, W. Szpankowski [132] strengthens this result and improves the range of the convergence to almost surely. In the almost sure convergence the probability that the above mentioned ratio tends to 1 is equal to 1. The asymptotic behavior of the QAP can be stated in the following theorem:

Theorem 3 *Consider a sequence of problems QAP($A^{(n)}$, $B^{(n)}$) for $n \in \mathbf{N}$, with $n \times n$ coefficient matrices $A^{(n)} = (a_{ij}^{(n)})$ and $B^{(n)} = (b_{ij}^{(n)})$. Assume that $a_{ij}^{(n)}$ and $b_{ij}^{(n)}$, $n \in \mathbf{N}$, $1 \leq i, j \leq n$, are independently distributed random variables on [0, M], where M is a positive constant. Moreover, assume that the entries $a_{ij}^{(n)}$, $n \in \mathbf{N}$, $1 \leq i, j \leq n$, have the same distribution, and the entries $b_{ij}^{(n)}$, $n \in \mathbf{N}$, $1 \leq i, j \leq n$, have also the same distribution, which does not necessarily coincide with that of $a_{ij}^{(n)}$. Furthermore, assume that these variables have finite expected values, variances and third moments. Let $\phi_{opt}^{(n)}$ and $\phi_{wor}^{(n)}$ denote an optimal and a worst solution of QAP($A^{(n)}$, $B^{(n)}$), respectively, that is,*

$$Z\left(A^{(n)}, B^{(n)}, \phi_{opt}^{(n)}\right) = \min_{\phi \in S_n} Z\left(A^{(n)}, B^{(n)}, \phi\right) \ ,$$

and

$$Z\left(A^{(n)}, B^{(n)}, \phi_{wor}^{(n)}\right) = \max_{\phi \in S_n} Z\left(A^{(n)}, B^{(n)}, \phi\right) \ .$$

Then the following equality holds almost surely:

$$\lim_{n \to \infty} \frac{Z\left(A^{(n)}, B^{(n)}, \phi_{opt}^{(n)}\right)}{Z\left(A^{(n)}, B^{(n)}, \phi_{wor}^{(n)}\right)} = 1 \ .$$

The above result suggests that the value of the objective function of the problem QAP($A^{(n)}$, $B^{(n)}$) (corresponding to an arbitrary feasible solution) gets somehow close to its expected value $n^2 \mathrm{E}(A)\mathrm{E}(B)$, as the size of the problem increases, where $\mathrm{E}(A)$ and $\mathrm{E}(B)$ are the expected values of $a_{ij}^{(n)}$ and $b_{ij}^{(n)}$, $n \in \mathbf{N}$, $1 \leq i, j \leq n$, respectively. J.C.B. Frenk, M. van Houweninge, and A.G. Rinnooy

Kan [54] and W.T. Rhee [126,127] provide different analytical evaluations for this 'getting close', by imposing different probabilistic conditions on the data.

Results on the asymptotic behavior of the QAP have been exploited by M.E. Dyer, Frieze, and C.J.H. McDiarmid [46] to analyze the performance of branch and bound algorithms for QAPs with randomly generated coefficients as described above. They have shown that for such QAPs the optimal value of the continuous relaxation of Frieze–Yadegar linearization as stated in (10), is in $O(n)$ with probability tending to 1 as the size n of the QAP tends to infinity. Hence the gap between the optimal value of this continuous relaxation and the optimal value of the QAP grows like $O(n)$ with probability tending to 1 as n tends to infinity.

Polynomially Solvable Cases

Since the QAP is *NP*-hard, restricted versions which can be solved in polynomial time are an interesting aspect of the problem. A basic question arising with respect to polynomially solvable versions is the identification of those versions and the investigation of the border line between hard and easy versions of the problem. There are two ways to approach this topic: first, find structural conditions to be imposed on the coefficient matrices of the QAP so as to obtain polynomially solvable versions, and secondly, investigate other combinatorial optimization or graph-theoretical problems which can be formulated as QAPs, and embed the polynomially solvable versions of the former into special cases of the later. These two approaches yield two groups of restricted QAPs which are briefly reviewed in this section. For a detailed information on this topic, see [35].

Most of the restricted versions of the QAP with specially structured matrices involve Monge matrices or other matrices having analogous properties. A matrix $A = (a_{ij})$ is a *Monge matrix* if and only if the following *Monge inequalities* are fulfilled for each 4-tuples of indices $i, j, k, l, i < k, j < l$:

$$a_{ij} + a_{kl} \leq a_{il} + a_{kj} .$$

A matrix $A = (a_{ij})$ is an *anti-Monge matrix* if and only if the following *anti-Monge inequalities* are fulfilled for each 4-tuples of indices $i, j, k, l, i < k, j < l$:

$$a_{ij} + a_{kl} \geq a_{il} + a_{kj} .$$

A simple example of Monge and anti-Monge matrices are the *sum matrices*; the entries of a sum matrix matrix $A = (a_{ij})$ are given as $a_{ij} = \alpha_i + \beta_j$, where (α_i) and (β_j) are the generating row and column vector, respectively. A *product matrix A* is defined in an analogous way: its entries are given as $a_{ij} = \alpha_i \beta_j$, where (α_i), (β_j) are the generating vectors. If the row generating vector (α_i) and the column generating vectors (β_i) are sorted nondecreasingly, then the product matrix $(\alpha_i \beta_j)$ is an anti-Monge matrix.

In contrast with the traveling salesman problem, it turns out that the QAP with both coefficient matrices being Monge or anti-Monge is *NP*-hard, whereas the complexity of a QAP with one coefficient matrix being Monge and the other one being anti-Monge is still open, see [23] and [35]. However, some polynomially solvable special cases can be obtained by imposing additional conditions on the coefficient matrices. These special cases involve very simple matrices like product matrices or so-called *chess-board matrices*. A matrix $A = (a_{ij})$ is a chess-board matrix if its entries are given by $a_{ij} = (-1)^{i+j}$. These QAPs can either be formulated as equivalent LAPs, or they are *constant permutation QAPs* (see [23,35]), that is, their optimal solution can be given before hand, without knowing the entries of their coefficient matrices. A few other versions of the QAP involving Monge and anti-Monge matrices with additional structural properties can be solved by dynamic programming. Other restricted versions of the QAP involve matrices with a specific diagonal structure such as *circulant* and *Toeplitz matrices*. An $n \times n$ matrix $A = (a_{ij})$ is called a Toeplitz matrix if there exist numbers $c_{-n+1}, \ldots, c_{-1}, c_0, c_1, \ldots, c_{n-1}$ such that $a_{ij} = c_{j-i}$, for all i, j. A matrix A is called a circulant matrix if it is a Toeplitz matrix and the generating numbers c_i fulfill the conditions $c_i = c_{n-i}$, for $0 \leq i \leq n-1$. In other words, a Toeplitz matrix has constant entries along lines parallel to the diagonal, whereas a circulant is given by its first row and the entries of the $i-$ th row resembles the first row shifted by $i-1$ places to the right.

In general versions of the QAP with one anti-Monge (Monge) matrix and one Toeplitz (circulant) matrix, remain *NP*-hard unless additional conditions, such as monotonicity, are imposed on the coefficient matrices. A well studied problem is the so called anti-

Monge–Toeplitz QAP where the rows and columns of the anti-Monge matrix are nondecreasing, investigated in [26]. It has been shown that this problem is *NP*-hard and contains as a special case the so-called turbine balancing problem (TBP) introduced in [99] and formulated as a QAP in [87]. In the TBP we are given a number of blades to be welded in regular spacing around the cylinder of the turbine. Due to inaccuracies in the manufacturing process the weights of the blades differ slightly and consequently the gravity center of the system does not lie on the rotation axis of the cylinder, leading to instabilities. In an effort to make the system as stable as possible, it is desirable to locate the blades so as to minimize the distance between the center of gravity and the rotation axis. The mathematical formulation of this problem leads to an *NP*-hard anti-Monge–Toeplitz QAP. (For more details and for a proof of *NP*-hardness see [26].) It is probably interesting that the maximization version of this problem is polynomially solvable. Further polynomially solvable special cases of the anti-Monge–Toeplitz QAP arise if additional constraints such as benevolence or k-benevolence are imposed on the Toeplitz matrix. These conditions are expressed in terms of properties of the generating function of these matrices, see [26]. The polynomially solvable QAPs with one anti-Monge (Monge) matrix and the other one Toeplitz (circulant) matrix described above, are all constant permutation QAPs. The techniques used to prove this fact and to identify the optimal permutation is called reduction to extremal rays. This technique exploits two facts: first, the involved matrix classes form cones, and secondly, the objective function of the QAP is linear with respect to each of the coefficient matrices. These two facts allow us to restrict the investigations to instances of the QAP with 0–1 coefficient matrices which are extremal rays of the above mentioned cones. Such instances can then be handled by elementary means (exchange arguments, bounding techniques) more easily that the general given QAP. The identification of polynomially solvable special cases of the QAP which are not constant permutation QAPs and can be solved algorithmically remains a challenging open question.

Another class of matrices similar to the Monge matrices are the *Kalmanson matrices*. A matrix $A = (a_{ij})$ is a Kalmanson matrix if it is symmetric and its elements satisfy the following inequalities for all indices i, j, k, l, $i < j < k < l$:

$$a_{ij} + a_{kl} \leq a_{ik} + a_{jl}, \quad a_{il} + a_{jk} \leq a_{ik} + a_{jl}\,.$$

For more information on Monge, anti-Monge and Kalmanson matrices, and their properties, see [32]. The Koopmans–Beckmann QAP with one coefficient matrix being a Kalmanson matrix and the other one a Toeplitz matrix, has been investigated in [44]. The computational complexity of this problem is an open question, but analogously as in the case of the anti-Monge–Toeplitz QAP, polynomially solvable versions of the problem are obtained by imposing additional constraints to the Toeplitz matrix.

Further polynomially solvable cases arise as QAP formulations of other problems, like the linear arrangement problem, minimum feedback arc set problem, packing problems in graphs and subgraph isomorphism, see [23,35]. Polynomially solvable versions of these problems lead to polynomially solvable cases of the QAP. The coefficient matrices of these QAPs are the (weighted) adjacency matrices of the underlying graphs, and the special structure of these matrices is imposed by properties of these graphs. The methods used to solve these QAPs range from graph theoretical algorithms (in the case of the linear arrangement problem and the feedback arc set problem), to dynamic programming (in the case of subgraph isomorphism).

Lower Bounds

Lower bounding techniques are primarily used with implicit enumeration algorithms, such as branch and bound, to perform a limited search of the feasible region of a minimization problem, until an optimal solution is found. A more limited use of lower bounding techniques, is also for evaluating the performance of heuristic algorithms, by providing a relative measure of proximity of the suboptimal solution to the optimum. In comparing lower bounding techniques, the following criteria should be taken into consideration:

- complexity of computing the lower bound;
- tightness of the lower bound (i. e. closest to the optimum solution);
- efficiency in computing lower bounds for subsets of the primal feasible set.

Since there is no clear ranking of the performance of the lower bounds that will be discussed below, all of the above criteria should be kept in mind while reading the following paragraphs. Considering the asymptotic behavior of the QAP, it should be fair to assume that the tightness of the lower bound probably dominates all of the above criteria. That is, if there is a large number of feasible solutions close to the optimum, then a lower bound which is not tight enough, will fail to eliminate a large number of subproblems in the branching process.

Gilmore–Lawler Type Lower Bounds

Based on the formulation of the general QAP as a LAP of dimension n^2 stated in formulation (9), Lawler [88] derived lower bounds for the QAP, by constructing a solution matrix Y in the process of solving a series of LAPs. If the resulting matrix Y is a permutation matrix, then the objective function value is optimal, otherwise it is bounded from below by $\langle C, Y\rangle$. Specifically, consider an instance of QAP(C), where the matrix C is partitioned into n^2 minors, $C^{(i,j)} = (c_{ijkl})_{n\times n}$ for $i, j = 1, \ldots, n$. Each minor $C^{(i,j)}$ essentially contains the costs associated with the assignment $x_{ij} = 1$. Partition the solution matrix Y also into n^2 minors, $Y^{(i,j)} = (y_{ijkl})_{n\times n}$ for $i, j = 1, \ldots, n$, whose actual values are to be determined in the process. Solve the n^2 LAPs associated with each minor $C^{(i,j)}$,

$$\begin{cases} l_{ij} = \min \sum_{k=1}^{n}\sum_{l=1}^{n} c_{ijkl}\, y_{ijkl} \\ \text{s.t.} \quad \sum_{k=1}^{n} y_{ijkl} = 1, \quad l = 1, \ldots, n, \\ \qquad \sum_{l=1}^{n} y_{ijkl} = 1, \quad k = 1, \ldots, n, \\ \qquad y_{ijij} = 1, \\ \qquad y_{ijkl} \in \{0, 1\}, \quad i, j = 1, \ldots, n. \end{cases} \tag{12}$$

Observe that the last constraint essentially reduces the problem into an LAP of dimension $(n-1)$, obtained by deleting the ith row and jth column of the matrix $C^{(i,j)}$. Denote the solution matrix for each of the above LAPs by $Y^{(i,j)}$. Using the values l_{ij} from above, solve the LAP to obtain the *Gilmore–Lawler lower bound* for general QAPs:

$$\begin{cases} \text{GLB}(C) = \min & \sum_{i=1}^{n}\sum_{j=1}^{n} l_{ij}x_{ij} \\ \text{s.t.} & (x_{ij}) \in \mathbf{X}_n, \end{cases} \tag{13}$$

and denote its solution matrix by $X^* = (x^*_{ij})$. If

$$\frac{1}{n}\sum_{ij} x^*_{ij} Y^{(ij)} \in \mathbf{X}_n \,,$$

then $Y^* = (x^*_{ij}\, Y^{(ij)})_{n^2\times n^2}$ is a Kronecker product of two permutation matrices of dimension n, and then it is also an optimal solution. Otherwise the optimal solution to the QAP is bounded below by $\text{GLB}(C) = \langle C, Y^*\rangle$. Considering that each LAP can be solved in $O(n^3)$ time, the above lower bound for the general QAP of dimension n can be computed in $O(n^5)$ time.

For the more special Koopmans–Beckmann formulation of the QAP (cf. formulation (3)) where the quadratic costs are derived by the pairwise product of two matrices F and D, the structure of the problem can be used to reduce the computational effort. Before we proceed, let us make some definitions. For vectors $a, b \in \mathbf{R}^n$, define the following extremal variations of the usual inner product between vectors, by imposing an ordering in the elements of the vectors:

$$\langle a, b\rangle^- := \sum_{i=1}^{n} a_i b_i \,, \tag{14}$$

where $a_i \geq a_{i+1}$, $b_i \leq b_{i+1}$, $\forall i$, and

$$\langle a, b\rangle^+ := \sum_{i=1}^{n} a_i b_i \,, \tag{15}$$

where $a_i \geq a_{i+1}$, $b_i \geq b_{i+1}$, $\forall i$. The following is a well known result:

Proposition 4 [69] *For $a, b \in \mathbf{R}^n$ the following inequalities hold for any $\phi \in \mathcal{S}_n$:*

$$\langle a, b\rangle^- \leq \sum_{i=1}^{n} a_i b_{\phi(i)} \leq \langle a, b\rangle^+ .$$

Consider an instance QAP(F, D, B), and recall that this can be transformed into an instance of QAP(C) by assigning the values

$$c_{ijkl} = \begin{cases} f_{ik}d_{jl}, & \text{for } i \neq k,\ j \neq l, \\ f_{ii}d_{jj} + b_{ij}, & \text{for } i = k,\ j = l. \end{cases}$$

Each minor $C^{(i,j)}$ in the partitioned matrix C, is now $C^{(i,j)} = f_{(i\cdot)}\, d_{(j\cdot)}^{\top}$, where $f_{(i\cdot)}$ and $d_{(j\cdot)}$ is the ith and jth row of matrix F and D respectively. Therefore, using the result of Proposition 4, instead of solving n^2 LAPs we can easily compute the values l_{ij} as

$$l_{ij} = f_{ii}d_{jj} + b_{ij} + \left\langle \widehat{f}_{(i\cdot)}, \widehat{d}_{(j\cdot)} \right\rangle^{-} , \tag{16}$$

where the vectors $\widehat{f}_{(i\cdot)}, \widehat{d}_{(j\cdot)} \in \mathbf{R}^{n-1}$ are obtained by removing the ith and jth element of the vectors $f_{(i\cdot)}$ and $d_{(j\cdot)}$ respectively. Finally by solving the resulting LAP as in (12), we obtain the Gilmore–Lawler lower bound for the Koopmans–Beckmann QAPs, denoted by GLB(F, D), in $O(n^3)$ time. Its name is due to the fact that Lawler [88] and P.C. Gilmore [60] independently derived this lower bound, while the first author considered the case for general QAPs also. The simplicity of the Gilmore–Lawler lower bound makes it one of the most efficient to compute, although it deteriorates fast as n increases. The quality of the lower bound can be improved if the contribution of the quadratic term in the objective function is made to be smaller than that of the linear term. Consider the formulation of the general QAP where the linear and the quadratic terms are separated for clarity. By the above discussion the lower bound will be the solution to the LAP

$$\begin{cases} \min & \displaystyle\sum_{i=1}^{n}\sum_{j=1}^{n}(l_{ij} + c_{ijij})x_{ij} \\ \text{s.t.} & (x_{ij}) \in \mathbf{X}_n. \end{cases}$$

We want to decompose the cost coefficients in the quadratic term of (4) and transfer some of their value into the linear term such that $c_{ijij} \gg l_{ij}$, which will result in a tighter lower bound since the LAP can be solved exactly. This procedure known as *reduction* was introduced in [41], and it has been investigated by many researchers (see [18,47,56,128]). The general idea is to decompose each quadratic cost coefficient into several terms, which in turn will end up being linear cost coefficients and will be moved in the linear term of the objective function. Consider the following general decomposition for each quadratic cost coefficient in the objective function in (4):

D1) $c_{ijkl} = \overline{c}_{ijkl} + e_{ijk} + g_{ijl} + h_{ikl} + t_{jkl}, i \neq k, j \neq l.$

Here $e, g, h, t \in \mathbf{R}^{n^3}$. Substituting the above and separating terms, the objective function in (4) becomes

$$\sum_{\substack{i=1\\ i\neq k}}^{n}\sum_{\substack{j=1\\ j\neq l}}^{n}\sum_{k,l=1}^{n}\overline{c}_{ijkl}x_{ij}x_{kl} + \sum_{\substack{i=1\\ i\neq k}}^{n}\sum_{\substack{j=1\\ j\neq l}}^{n}\sum_{k,l=1}^{n}(e_{ijk} + g_{ijl} + h_{ikl} + t_{jkl})x_{ij}x_{kl} + \sum_{i,j=1}^{n} c_{ijij}x_{ij} .$$

Consider now the term associated with the e_{ijk}:

$$\sum_{i,j=1}^{n}\sum_{\substack{k=1\\ k\neq i}}^{n}\sum_{\substack{l=1\\ l\neq j}}^{n} e_{ijk}x_{ij}x_{kl} = \sum_{i,j=1}^{n} x_{ij}\left[\sum_{\substack{k=1\\ k\neq i}}^{n} e_{ijk}\left(\sum_{\substack{l=1\\ l\neq j}}^{n} x_{kl}\right)\right] .$$

We can add the term

$$\sum_{\substack{k=1\\ k\neq i}}^{n} e_{ijk}$$

to the (i, j)th element of the LAP that composes the linear term of the objective function, since $x_{ij} = 1 \Rightarrow x_{kj} = 0 \Rightarrow \sum_{l\neq j} x_{kl} = 1, \forall k$. Using similar arguments for the vectors g, h and t, their costs become linear and the objective function with decomposed quadratic costs become

$$\sum_{\substack{i=1\\ i\neq k}}^{n}\sum_{\substack{j=1\\ j\neq l}}^{n}\sum_{k=1}^{n}\sum_{l=1}^{n}\overline{c}_{ijkl}x_{ij}x_{kl} + \sum_{i=1}^{n}\sum_{j=1}^{n}\widehat{c}_{ij}x_{ij} ,$$

where

$$\widehat{c}_{ij} = c_{ijij} + \sum_{\substack{k=1\\ k\neq i}}^{n} e_{ijk} + \sum_{\substack{l=1\\ l\neq j}}^{n} g_{ijl} + \sum_{\substack{k=1\\ k\neq i}}^{n} h_{kij} + \sum_{\substack{l=1\\ l\neq j}}^{n} t_{lij} .$$

Therefore we can apply the Gilmore–Lawler bound in the quadratic term of the decomposed objective function, whereas we can get an exact solution to the LAP that composes the linear term, and the sum of these two values will constitute a lower bound for the QAP. In the case of the Koopmans–Beckmann formulation of the QAP were we have two matrices F and D, the general decomposition scheme is:

D2)

$$\begin{cases} f_{ij} = \overline{f}_{ij} + \lambda_i + \mu_j, & i \neq j, \\ d_{kl} = \overline{d}_{kl} + \nu_k + p_l, & k \neq l. \end{cases}$$

Here $\lambda, \mu, \nu, p \in \mathbf{R}^n$. Substituting to the product $f_{ij}d_{kl}$ it is easily seen that D2) reduces to the general decomposition D1) with vectors $e, g, h, t \in \mathbf{R}^{n^3}$. Frieze and Yadegar [56] showed that the inclusion of the vectors h and t in D1) does not affect the value of the lower bound, and therefore are redundant (similarly the vectors μ and p for the Koopmans–Beckmann QAP are redundant in D2)). The same authors in [56] derived lower bounds for the QAP based on a Lagrangian relaxation (cf. also ► Integer programming: Lagrangian relaxation). Specifically, consider the Lagrangian relaxation of the 0–1 linear programming formulation of the QAP (see (10)), where the second and third constraints are included in the objective function, using as Lagrangian multipliers the elements of the vectors e and g. The Lagrangian function is thus defined as

$$\mathcal{L}(e,g) := \begin{cases} \min & \sum_{ijkl} c_{ijkl} y_{ijkl} \\ & \quad + \sum_{jkl} e_{jkl} \left(x_{kl} - \sum_i y_{ijkl} \right) \\ & \quad + \sum_{ikl} g_{ikl} \left(x_{kl} - \sum_j y_{ijkl} \right) \\ & \quad = \sum_{ijkl} (c_{ijkl} - e_{jkl} - g_{ikl}) y_{ijkl} \\ & \quad + \sum_{ij} \left(\sum_k e_{kij} + \sum_l g_{lij} \right) x_{ij} \\ \text{s.t.} & \text{first constraint in (10),} \\ & \text{fourth constraint in (10)} \\ & \ldots \\ & \text{last constraint in (10).} \end{cases}$$

The authors prove in [56] that for any choice of e and g, the solution to the above minimization problem will equal the value of the Gilmore–Lawler lower bound as applied to the QAP, with decomposed quadratic cost coefficients, as dictated by using the vectors e and g only in D1). Therefore, $\max_{e,g} \mathcal{L}(e,g)$ constitutes an upper bound on the lower bounds for the QAP, obtained by using the Gilmore–Lawler bound with decomposed quadratic cost coefficients. Using subgradient algorithms (cf. also ► Nondifferentiable optimization: Subgradient optimization methods) the authors derive near optimal solutions for $\max_{e,g} \mathcal{L}(e,g)$ resulting in two lower bounds, denoted by $FY1$ and $FY2$, corresponding to the two different solution approaches proposed. As suggested by the experimental results in [56], these bounds seem to be sharper than previously reported Gilmore–Lawler based lower bounds using reduction techniques. Almost all of the other approaches for obtaining lower bounds for the QAP with reduction techniques, are special cases of the general decomposition scheme D2) (see [18,47,128]).

Variance Reduction Lower Bounds

The variance reduction lower bounds were introduced in [93]. Consider an instance of the Koopmans–Beckmann formulation of the QAP, with input matrices $F = (f_{ij}), D = (d_{ij}) \in \mathbf{R}^{n \times n}$. Now partition both matrices into a sum of two matrices, $F = F_1 + F_2$ and $D = D_1 + D_2$, were $F_1 = (f^{(1)}_{ij})$, $F_2 = (f^{(2)}_{ij})$ and $D_1 = (d^{(1)}_{ij})$, $D_2 = (d^{(2)}_{ij})$. Construct an $n \times n$ matrix $L = (l_{ij})$, by solving the following n^2 LAPs:

$$l_{ij} = \min_{\substack{\phi \in S_n \\ \phi(i) = j}} \sum_{k=1}^{n} f^{(1)}_{ik} d^{(1)}_{j\phi(k)} + f^{(1)}_{ki} d_{\phi(k)j} + f_{ki} d^{(2)}_{\phi(k)j} - f^{(2)}_{ki} d^{(2)}_{\phi(k)j} \quad (17)$$

It is proved in [93] that the solution of the LAP with cost matrix L as constructed above, constitutes a lower bound for QAP(F, D). The problem of concern now, is to find a way to partition the matrices F and D such that the resulting lower bound is maximized. Observe that when $F_1 = F$ and $D_1 = D$ (i. e. no partitioning), we essentially derive the GLB(F, D).

Let $M \in \mathbf{R}^{m \times n}$, and denote its rows and columns respectively as $m_{(i\cdot)}$, and $m_{(\cdot,j)}$, $i, j = 1, \ldots, n$. Think of M

as a data set of mn elements m_{ij}, and define an average $\gamma(M)$ and a variance $V(M)$ as,

$$\gamma(M) := \frac{1}{mn}\sum_{i=1}^{m}\sum_{j=1}^{n} m_{ij}\,,$$

$$V(M) := \sum_{i=1}^{m}\sum_{j=1}^{n}(\gamma(M) - m_{ij})^2\,.$$

Also define the total variance,

$$T(M,\lambda) := \lambda\sum_{i=1}^{m} V(m_{(i\cdot)}) + (1-\lambda)V(M)\,,$$

where $\lambda \in [0,1]$. The term $V(m_{(i\cdot)})$ stands for the variance of $m_{(i\cdot)}$, treated as a $1 \times n$ matrix. The authors observed that, as the variances of the matrices F and D decrease, the GLB(F, D) increases, while it is optimal if the variances of the rows of the matrices are zero. The partition scheme considered is of the form, $F_1 = F + \Delta_F$, $F_2 = -\Delta_F$, and $D_1 = D + \Delta_D$, $D_2 = -\Delta_D$. Considering only the matrix F, the problem is to find a matrix Δ_F, such that the variances of F_1 and F_2 and the sum of the variances of the rows for each F_1 and F_2 are minimized. We will only describe how Δ_F is obtained since Δ_D is obtained in the same way. The problem of minimizing the variances can be stated mathematically as

$$\min\, \theta T(F+\Delta_F,\lambda) + (1-\theta)T(-\Delta_F^{\top},\lambda)\,, \qquad (18)$$

where $\Delta_F \in \mathbf{R}^{n\times n}$ and $\theta \in [0,1]$ is a parameter. Two approximate solutions were proposed in [93], corresponding to the two reduction schemes

R1) $\delta_{ij} = \theta(f_{nn} - f_{ij}) + \delta_{nn}$,

R2) $\delta_{ij} = \theta(\gamma(f_{(\cdot,n)}) - \gamma(f_{(\cdot,j)})$.

Here $i, j = 1, \ldots, n$, $\Delta_F = (\delta_{ij})$, and with δ_{nn} being free to take any value (it was given a value of zero in the experiments conducted in [93]). In obtaining R2), the problem of minimizing the variances such that the matrix Δ_F is constrained to have constant columns, is considered. The matrix Δ_D is constructed in the same way. Based on the two reductions schemes above, the resulting lower bounds from the solution of (17) are denoted by $LB1(\theta)$, and $LB2(\theta)$. The above procedure for computing Δ_F, Δ_D has $O(n^2)$ computational complexity.

After the partitioning of the matrices F and D, the solution to the LAP with cost matrix $L = (l_{ij})$, were l_{ij} are defined in (17), will yield $LB1(\theta)$ or $LB2(\theta)$ according to what reduction scheme used. If $LB2(\theta)$ is used, the fact that the matrices F_2 and D_2 have constant columns can be exploited to compute the l_{ij}, $i, j = 1, \ldots, n$, efficiently as

$$l_{ij} = \left\langle \widehat{f}^{(1)}_{(i\cdot)}, \widehat{d}^{(1)}_{(j\cdot)} \right\rangle^{-} + f^{(2)}_{1i} \sum_{\substack{k=1\\k\neq j}}^{n} d_{kj} + d^{(2)}_{1j} \sum_{\substack{k=1\\k\neq i}}^{n} f_{kj} - (n-1) f^{(2)}_{1i} d^{(2)}_{1j} + f_{ii} d_{jj}\,,$$

where $\widehat{f}^{(1)}_{(i\cdot)}, \widehat{d}^{(1)}_{(j\cdot)} \in \mathbf{R}^{n-1}$ are the ith and jth row of F_1 and D_1 respectively, with the ith and jth elements removed from each, and $\langle\cdot,\cdot\rangle^{-}$ is defined in (14). In the case that $LB1(\theta)$ is used, the direct approach would be to solve the n^2 LAPs defined in (17), which will require $O(n^5)$ computational effort. A different approach is to calculate lower bounds for the values l_{ij}, $i, j = 1, \ldots, n$, as follows:

$$l_{ij} = \left\langle \widehat{f}^{(1)}_{(i\cdot)}, \widehat{d}^{(1)}_{(j\cdot)} \right\rangle^{-} + \left\langle \widehat{f}^{(2)}_{(\cdot,i)}, \widehat{d}_{(\cdot,j)} \right\rangle^{-} + \left\langle \widehat{f}_{(\cdot,i)}, \widehat{d}^{(2)}_{(\cdot,j)} \right\rangle^{-} + \left\langle \widehat{f}^{(2)}_{(\cdot,i)}, \widehat{d}^{(2)}_{(\cdot,j)} \right\rangle^{+}\,,$$

where each vector in the above extremal inner products, is of dimension n − 1, and corresponds to the ith row or column of the indicated matrix, upon removal of the ith element. Similarly as before the extremal inner products $\langle\cdot,\cdot\rangle^{-}$ and $\langle\cdot,\cdot\rangle^{+}$ are defined in (14) and (15). Using the above approach would require $O(n^3)$ time to compute lower bounds for the l_{ij}, $i, j = 1, \ldots, n$, thus the total computational complexity of the variance reduction lower bounds is $O(n^3)$. It is worth noting that there is also a closed form solution to problem (18) given in [71] which is

$$\delta_{ij} = \theta\lambda\frac{1-\theta}{1-\theta\lambda}\gamma(f_{(i\cdot)}) + \frac{\theta(1-\lambda)+\theta\lambda^2(1-\theta)-\theta^2\lambda^2(1-\theta)}{(1-\theta\lambda)(1-\lambda+\theta\lambda)}\gamma(F) - \frac{\theta\lambda(1-\theta)}{1-\lambda+\theta\lambda}\gamma(f_{(\cdot j)}) - \theta f_{ij}\,,$$

for $i, j = 1, \ldots, n$. However it was reported in [93] that using the above closed form in the computation of the lower bounds, poses implementation obstacles.

The experimental results conducted in [93], also suggest that the settings of $\theta = 0.5$ for $LB1(\theta)$, and $\theta = 1$ for $LB2(\theta)$ as best choices. Finally, these lower bounds perform well on QAPs with input matrices that have high variances, but their performance reduces to that of the Gilmore–Lawler bounds when the variance of the matrices is small.

Eigenvalue Based Lower Bounds

These bounds were introduced in [50,51], and are applied to the Koopmans–Beckmann formulation of the QAP. This approach utilizes known results on permutation matrices and eigenvalues, and exploits the special structure of QAP(F, D). Upon the introduction of the method in [50,51], many improvements and generalizations have appeared (see for example [65,66,67,68,123,124]). There is a resemblance with the Gilmore–Lawler based lower bounds, in the sense that, based upon a general lower bound, reduction techniques are applied to the quadratic terms of the objective function in order to improve its quality. The reduction techniques that applied to eigenvalue based lower bounds however, yield a significant improvement, which is not really the case with the Gilmore–Lawler bounds under certain reductions.

Considering the trace formulation of the QAP in (7), with F and D being real symmetric matrices, therefore with all their eigenvalues being real, the following result can be stated for the quadratic term [51]:

Theorem 5 [51] *Let $F, D \in \boldsymbol{R}^{n\times n}$ be symmetric matrices, and denote by $\lambda = (\lambda_1, \ldots, \lambda_n)^\intercal$ and $x_1, \ldots, x_n$ the eigenvalues and eigenvectors of F, and by $\mu = (\mu_1, \ldots, \mu_n)^\intercal$ and $y_1, \ldots, y_n$ the the eigenvalues and eigenvectors of D. Then the following two relations are true for all $X \in \boldsymbol{X}_n$,*

i) $tr\, FXDX^\intercal = \sum_{i=1}^n \sum_{j=1}^n \lambda_i \mu_j \langle x_i, Xy_j\rangle^2 = \lambda^\intercal S(X)\mu$. Here $S(X) = (\langle x_i, Xy_j\rangle^2)$ is a doubly stochastic matrix,

ii) $\langle \lambda, \mu\rangle^- \le tr\, FXDX^\intercal \le \langle \lambda, \mu\rangle^+$.

Using Theorem 5ii) a lower bound for QAP(F, D) based on the eigenvalues of F and D is then

$$EVB = \langle \lambda, \mu\rangle^- + \min_{X\in\mathbf{X}_n} \operatorname{tr} BX^\top ,$$

where the second term is an ordinary LAP that can be solved exactly. Observe that in Theorem 5, the smaller the interval $[\langle \lambda, \mu\rangle^-, \langle \lambda, \mu\rangle^+]$ is, the closest $\langle \lambda, \mu\rangle^-$ is to tr $FXDX^\intercal$. A possible way of making the interval smaller, is to decompose the matrices F and D such that some of their value will be transfered in the linear term, and the eigenvalues of the resulting matrices that compose the quadratic term, are as uniform in value as possible. Define the *spread* of the matrix F as

$$\text{spread}(F) := \max\left\{\left|\lambda_i - \lambda_j\right| :\ i, j = 1, \ldots, n\right\} .$$

Based on the above discussion, we want to minimize the spreads of the matrices that compose the quadratic term. There is no simple closed form for expressing spread(F) in terms of f_{ij}, however we can minimize instead a formula for the upper bound given in [98]:

$$\text{spread}(F) \le m(F) = \left[2\sum_{i=1}^n \sum_{j=1}^n f_{ij}^2 - \frac{2}{n}(\operatorname{tr} F)^2\right]^{\frac{1}{2}} . \tag{19}$$

The decomposition scheme that the authors use in [51], is the following:

$$f_{ij} = \overline{f}_{ij} + e_i + e_j + r_{ij} , \tag{20}$$

$$d_{kl} = \overline{d}_{kl} + g_k + g_l + s_{kl} , \tag{21}$$

where $r_{ij} = s_{ij} = 0$, for $i \ne j$.

Consider the decomposition for matrix F and let $\bar{F} = (\overline{f}_{ij})$. Minimizing the function $f(e, r) = m(\overline{F})$ obtained by substituting the values of $\overline{f}_{ij}$ in (19), the following values are obtained [51]:

$$z = \frac{1}{2(n-1)}\left(\sum_{i=1}^n \sum_{j=1}^n f_{ij} - \operatorname{tr} F\right) , \tag{22}$$

$$e_i = \frac{1}{n-2}\left(\sum_{j=1}^n f_{ij} - f_{ii} - z\right) , \tag{23}$$

$$r_{ii} = f_{ii} - 2e_i , \tag{24}$$

for $i = 1, \ldots, n$. Analogously we obtain the values for g and s for the decomposition of D. Replacing F and D in (7) we obtain

$$\operatorname{tr}(FXD + B)X^\top = \operatorname{tr}(\overline{F}X\overline{D} + \overline{B})X^\top ,$$

where $\overline{b}_{ij} = b_{ij} + f_{ii}d_{jj} + 2e_i \sum_{\substack{k=1 \\ k\neq j}}^{n} d_{jk}$, and matrices $\overline{F}$ and $\overline{D}$ have respective eigenvalues $\overline{\lambda} = (\overline{\lambda}_1, \ldots, \overline{\lambda}_n)$ and $\overline{\mu} = (\overline{\mu}_1, \ldots, \overline{\mu}_n)$. The corresponding eigenvalue lower bound is then

$$EVB1 = \left\langle \overline{\lambda}, \overline{\mu} \right\rangle^{-} + \min_{X \in \mathbf{X}_n} \operatorname{tr} \overline{B} X^{\top} .$$

If we restrict ourselves only to pure quadratic ($f_{ii} = d_{ii} = 0, \forall i, B = 0$) symmetric QAPs, the matrix $\overline{B}$ in the above decomposition becomes $\overline{B} = cw^{\top}$, where $c = 2(e_1, \ldots, e_n)^{\top}$ and $w = (\sum_j d_{1j}, \ldots, \sum_j d_{nj})^{\intercal}$. Therefore $\min_{X \in \mathbf{X}_n} \operatorname{tr} \overline{B} X^{\top} = \langle c, w \rangle^{-}$, and

$$EVB1 = \left\langle \overline{\lambda}, \overline{\mu} \right\rangle^{-} + \langle c, w \rangle^{-} \leq \min_{X \in \mathbf{X}_n} \operatorname{tr}(\overline{F} X \overline{D} + \overline{B}) X^{\top} .$$

We can however obtain further improvement as suggested by F. Rendl [123], who examined the linear term $\langle c, w \rangle^{-}$, and proposed a method where $EVB1$ is iteratively improved, until some specified number of iterations is reached, or we have satisfied an optimality condition. More specifically, let $S_k := \{X_1, \ldots, X_k\} \subseteq \mathbf{X}_n$, and

$$L(X_i) := \min \{ \langle c, X_i w \rangle : \ X_i \in \mathbf{X}_n \setminus S_{i-1} \} ,$$

so for any integer $k \geq 1$, $L(X_1) \leq L(X_2) \leq \cdots \leq L(X_k)$. In other words the set S_k contains the k first solutions (permutation matrices) of the problem $\min_{X \in X_n} \langle c, X_i w \rangle$, where the first solution $X_1 \ni L(X_1) = \langle c, w \rangle^{-}$. Let

$$\mathrm{QAP}(\overline{F}, \overline{D}, X_i) = \operatorname{tr}(\overline{F} X_i \overline{D} + \overline{B}) X_i^{\top} ,$$

and also define the following

$$Z(k) := \min \left\{ \mathrm{QAP}(\overline{F}, \overline{D}, X_i) : \ i = 1, \ldots, k \right\} .$$

The following inequalities [123] result:

$$Z(1) \geq \cdots \geq Z(k) \geq \left\langle \overline{\lambda}, \overline{\mu} \right\rangle^{-} + L(X_k) \geq \cdots \geq \left\langle \overline{\lambda}, \overline{\mu} \right\rangle^{-} + L(X_1) ,$$

where if $Z(i) = \left\langle \overline{\lambda}, \overline{\mu} \right\rangle^{-} + L(X_i)$ for some i, then X_i is the optimal solution to the problem. So essentially, we try to close or reduce the gap between the optimal solution of the QAP and the lower bound $EVB1$, by increasing the value of the linear term $\langle c, w \rangle^{-}$ in the bound in k steps, where k is specified as a parameter. The generation of the set S_k or ranking as it is called, is a special case of the problem of ranking the k first solutions of an assignment problem with cost matrix $(c_i w_j)$ where, as shown in [104], has time complexity $O(kn^3)$. Rendl [123] presents an $O(n \log n + (n + \log k)k)$ for this special case. There are two issues regarding the effectiveness of the above ranking procedure, in improving the lower bound, addressed in [123]. First, observe that if the vectors c and w have $m \leq n$ equal elements, then there are at least $m!$ permutation matrices $\{X_i\}$ such that the values $\langle c, X_i w \rangle$ are equal. This in turn, implies that there will be none or small improvement in the lower bound while generating S_k for quite some number of iterations. As dictated by the decomposition in (22), (23), c and w will have equal elements if the row sums of F and D are equal. One condition then for applying the ranking procedure, is that most of the row sums of F and D are not equal. Secondly, Rendl [123] also defines a ratio called the *degree of linearity* based on the ranges of the quadratic and linear terms that compose the lower bound

$$L := \frac{\left\langle \overline{\lambda}, \overline{\mu} \right\rangle^{+} - \left\langle \overline{\lambda}, \overline{\mu} \right\rangle^{-}}{\langle c, w \rangle^{+} - \langle c, w \rangle^{-}} .$$

The influence of the linear term on the lower bound then is inversely proportional to the value of L. A small value of L suggests that the ranking procedure would be beneficial for the improvement of $EVB1$ for symmetric, pure quadratic QAPs. For large values of L, we can expect that the quadratic term dominates the linear term in the objective function, and [51] suggest the following improvement on $EVB1$. Considering Theorem 5i) as applied to the reduced matrices $\overline{F}$ and $\overline{D}$, denote the elements of the matrix $S(X)$ by $s_{ij} = \langle x_i, X y_j \rangle^2$. We can apply the bounds $l_{ij} \leq s_{ij} \leq u_{ij}$ where

$$u_{ij} = \max\{(\langle x_i, y_j \rangle^{-})^2, (\langle x_i, y_j \rangle^{+})^2\},$$

$$l_{ij} = \begin{cases} 0 & \text{if } \langle x_i, y_j \rangle^{-}, \langle x_i, y_j \rangle^{+} \text{ differ in sign,} \\ \min\{(\langle x_i, y_j \rangle^{-})^2, (\langle x_i, y_j \rangle^{+})^2\} & \text{otherwise.} \end{cases}$$

Recalling the fact that the s_{ij} are the elements of a doubly stochastic matrix, we can then form the *capacitated*

transportation problem

$$\text{CTP}^* \begin{cases} \min & \displaystyle\sum_{i=1}^{n}\sum_{j=1}^{n} \overline{\lambda}_i \overline{\mu}_j s_{ij} \\ \text{s.t.} & \displaystyle\sum_{i=1}^{n} s_{ij} = 1, \quad j = 1, \ldots, n, \\ & \displaystyle\sum_{j=1}^{n} s_{ij} = 1, \quad i = 1, \ldots, n, \\ & l_{ij} \le s_{ij} \le u_{ij}. \end{cases}$$

The new lower bound then would be

$$EVB2 = \text{CTP}^* + \langle c, w \rangle^- \ .$$

A more generalized approach to eigenvalue based lower bounding techniques, was employed in [66], that led to new lower bounds. Consider the following sets of $n \times n$ matrices, where $I \in \mathbf{R}^{n\times n}$ is the identity matrix, and $u := (1, \ldots, 1)^\intercal \in \mathbf{R}^n$ is the vector of ones,

$$\begin{aligned} \mathcal{O} &:= \left\{X\colon\ X^\top X = I\right\} \ , \\ \mathcal{E} &:= \left\{X\colon\ Xu = X^\top u = u\right\} \ , \\ \mathcal{N} &:= \{X\colon\ X \ge 0\} \ . \end{aligned}$$

It is a well known result that $\mathbf{X}_n = \mathcal{O} \cap \mathcal{E} \cap \mathcal{N}$, while the set of doubly stochastic matrices $\Omega = \mathcal{E} \cap \mathcal{N}$. Moreover, by Birkhoff's theorem [15] we know that Ω is a convex polyhedron with a vertex set $\mathbf{X}_n$, that is, $\Omega = \text{conv}\{X : X \in \mathbf{X}_n\}$. Considering the above characterization of $\mathbf{X}_n$, we can see that any solution to a relaxation of the QAP obtained from excluding one or two of the matrix sets $\mathcal{O}$, $\mathcal{E}$ and $\mathcal{N}$, will yield a lower bound. Naturally the relaxation, and therefore the lower bound, will be tighter if only one of the matrix sets is excluded. In relation to Theorem 5, Rendl and Wolkowicz [124] showed that

$$\begin{aligned} \min_{X\in\mathcal{O}} \operatorname{tr} FXDX^\top &= \operatorname{tr} F\Lambda_F \Lambda_D^\top D \Lambda_D \Lambda_F^\top = \langle \lambda, \mu \rangle^- \ , \\ \max_{X\in\mathcal{O}} \operatorname{tr} FXDX^\top &= \operatorname{tr} F\Lambda_F \Lambda_D^\top D \Lambda_D \Lambda_F^\top = \langle \lambda, \mu \rangle^+ \ , \end{aligned}$$

where Λ_F, Λ_D are the matrices with columns the eigenvectors of F and D respectively, in the order specified by the minimal (maximal) inner product of the eigenvalues. In other words, the lower bound on the quadratic part of the QAP as obtained from the *EVB*, is derived by relaxing the feasible set to that of orthogonal matrices. In [124] a new lower bound is derived, similar to *EVB*2 but using a different approach to decompose the matrices F and D. More specifically, denote the decomposition scheme in (20) and (21) by the vector $d := (e^\intercal, g^\intercal, r^\intercal, s^\intercal) \in \mathbf{R}^{4n}$, where $r = (r_{11}, \ldots, r_{nn})^\intercal$ and $s = (s_{11}, \ldots, s_{nn})^\intercal$, and consider *EVB*1 as a function of d. Maximizing this function with respect to d will result in a lower bound with the best possible decomposition that involves both the linear and quadratic terms. This leads to a nonlinear, nonsmooth, nonconcave function which is hard to solve, and a steepest ascent algorithm is proposed for maximizing it in [124]. The new bound, denoted *EVB*3, produces some of the best lower bounds for the QAP, with the expense however of high computational requirements.

All of the above discussed lower bounds, relax the set of permutation matrices to $\mathcal{O}$. A tighter relaxation was proposed in [67], where the set of permutation matrices was relaxed to $\mathcal{O} \cap \mathcal{E}$, by incorporating $\mathcal{E}$ in the objective function, by exploiting the fact that the vector of ones u is both a left and right eigenvector with eigenvalue 1, for any $X \in \mathbf{X}_n$. More specifically define

$$P := [u/\|u\| \vdots V] \ ,$$

where $V^\intercal\ u = 0$, $V^\intercal\ V = I_{n-1}$. therefore V is an orthonormal basis for $\{u\}^\perp$, while $Q := VV^\intercal$ is the orthogonal projection on $\{u\}^\perp$. The following characterization of the permutation matrices is given in [67]

Lemma 6 [67] *Let $X \in \boldsymbol{R}^{n\times n}$ and $Y \in \boldsymbol{R}^{n-1\times n-1}$. If*

$$X = P \begin{bmatrix} 1 & 0 \\ 0 & Y \end{bmatrix} P^\top \ , \tag{25}$$

then

$$\begin{aligned} & X \in \mathcal{E} \ , \\ & X \in \mathcal{N} \Leftrightarrow VYV^\top \ge -\frac{uu^\top}{\|u\|}^2 \ , \\ & X \in \mathcal{O} \Leftrightarrow Y \in \mathcal{O}_{n-1} \ . \end{aligned}$$

Conversely if $X \in \mathcal{E}$ then there exists a Y such that (25) holds.

Note that the above characterization of the permutation matrices, preserves the orthogonality and the trace structure of the problem. Substituting $X = -uu^\intercal/\|u\|^2 + VYV^\intercal$ as suggested by (25), in the trace formulation of the QAP in (7), we have an equivalent projected problem (PQAP) of dimension $n-1$ on the variable

matrix Y. The new lower bound *IVB* is obtained by relaxing Y to O_{n-1}, therefore deriving a lower bound for the quadratic part of PQAP, while the linear part can be solved exactly as an LAP. Decompositions for improving the *IVB* are also considered in [67], where it is shown that the quadratic term in the projected problem is unaffected by e and g in the decomposition scheme in (20), (21). Obtaining a lower bound by considering both the quadratic and linear term is also considered in [78].

The symmetry assumption on the QAP is required by any of the eigenvalue based lower bounding techniques described above. Hadley, Rendl and Wolkowicz [68] show that any real QAP can be transformed into an equivalent QAP where the matrices F and D are Hermitian, which allows the application of eigenvalue based bounds.

Bounds Based on Semidefinite Relaxations

Recently (as of 1999), semidefinite programming (SDP) relaxations for the QAP were considered [76,77,137]. The SDP relaxations considered in these papers are solved by interior point methods or cutting plane methods (cf. also ► Linear programming: Interior point methods; ► Extended cutting plane algorithm), and the obtained solutions are valid lower bounds for the QAP. In terms of quality the bounds obtained in this way are competitive with the best existing lower bounds for the QAP. For many test instances from QAPLIB [31] such as some instances of Hadley [26], Roucairol [128], Nugent et al. [105], and Taillard [133], they are the best existing bounds. However, due to prohibitively high computation time requirements, the use of such approaches as basic bounding procedures within branch and bound algorithms is up to now not feasible. See [77,78] for a detailed description of SDP approaches to the QAP and illustrate the idea by describing just one semidefinite programming relaxation for the QAP.

The set of $n \times n$ permutation matrices $\mathbf{X}_n$ is the intersection of the set of $n \times n$ 0–1 matrices, denoted by $\mathcal{Z}_n$, and the set $\mathcal{E}_n$ of $n \times n$ matrices with row and column sums equal to 1. Moreover, $\mathbf{X}_n$ is also the intersection of $\mathcal{Z}_n$ with the set of $n \times n$ orthogonal matrices, denoted by $\mathcal{O}_n$. Hence

$$\mathbf{X}_n = \mathcal{Z}_n \cap \mathcal{E}_n = \mathcal{Z}_n \cap \mathcal{O}_n \ .$$

Recall that

$$\mathcal{O}_n = \left\{X \in \mathbf{R}^{n\times n}\colon\ XX^\top = X^\top X = I\right\}$$

and

$$\mathcal{E}_n = \left\{X \in \mathbf{R}^{n\times n}\colon\ Xu = X^\top u = u\right\}\ ,$$

where I is the $n \times n$ identity matrix and u is the n-dimensional vector of all ones. Then, the trace formulation of the QAP (7) with the additional linear term

$$-2\sum_{i=1}^{n}\sum_{j=1}^{n} b_{ij}x_{ij},$$

can be represented equivalently as follows:

$$\mathrm{QAP}_E \begin{cases} \min & \mathrm{tr}(FXDX^\top - 2BX^\top) \\ \text{s.t.} & XX^\top = X^\top X = I, \\ & Xu = X^\top u = u, \\ & x_{ij}^2 - x_{ij} = 0. \end{cases}$$

In order to obtain a semidefinite relaxation for the QAP from the formulation $\mathrm{QAP}_{\mathcal{E}}$ above, we introduce first an n^2-dimensional vector $\mathrm{vec}(X)$. $\mathrm{vec}(X)$ is obtained as a columnwise ordering of the entries of matrix X. Then the vector $\mathrm{vec}(X)$ is lifted into the space of $(n^2+1)\times(n^2+1)$ matrices by introducing a matrix Y_X,

$$Y_X = \begin{pmatrix} x_0 & \mathrm{vec}(X)^\top \\ \mathrm{vec}(X) & \mathrm{vec}(X)\,\mathrm{vec}(X)^\top \end{pmatrix}.$$

Thus, Y_X has some entry x_0 in the left-upper corner followed by the vector $\mathrm{vec}(X)$ in its first row (column). The remaining terms are those of the matrix $\mathrm{vec}(X)\ \mathrm{vec}(X)^\intercal$ sitting on the right lower $n^2 \times n^2$ block of Y_X. Secondly, the coefficients of the problem are collected in an $(n^2+1)\times(n^2+1)$ matrix K given as

$$K = \begin{pmatrix} 0 & -\mathrm{vec}(B)^\top \\ \mathrm{vec}(B) & D \otimes F \end{pmatrix},$$

where the operator vec is defined as above and $D \otimes F$ is the Kronecker product of D and F.

It is easy to see that with these notations the objective function of $\mathrm{QAP}_{\mathcal{E}}$ equals $\mathrm{tr}(KY_X)$. By setting $y_{00} := x_0 = 1$ as done in [77], one obtains two additional

constraints to be fulfilled by the matrix Y_X: Y_X is positive semidefinite and matrix Y_X is a rank-one matrix. Whereas the semidefiniteness and the equality $y_{00} = 1$ can be immediately included in an SDP relaxation, the rank-one condition is hard to handle and is discarded in an SDP relaxation. In order to assure that the rank-one positive semidefinite matrix Y_X is obtained by an $n \times n$ permutation matrix as described above, other constraints should be imposed to Y_X. Such conditions can be formulated as valid constraints of an SDP formulation for the QAP by means of some new operators, acting on matrices or vectors as introduced below. Given a matrix $A \in \mathbf{R}^{n\times n}$, the operator $\mathrm{diag}(A) \in \mathbf{R}^n$ produces a vector containing the diagonal entries of matrix A in their natural order, that is, from top-left to bottom-right. The adjoint operator Diag acts on a vector $V \in \mathbf{R}^n$ and produces a matrix $\mathrm{Diag}(V) \in \mathbf{R}^{n\times n}$ with off-diagonal entries equal to 0 and the components of V on the main diagonal. For some matrix $Y \in \mathbf{R}^{(n^2+1)\times(n^2+1)}$, operator $\mathrm{arrow}(Y) \in \mathbf{R}^{(n^2+1)}$, is defined as $\mathrm{arrow}(Y) := \mathrm{diag}(Y) - (0, Y_{0,1:n^2})$, where $(0, Y_{(0,1:n^2)})$ is an n^2+1-dimensional vector with first entry equal to 0 and other entries coinciding with the entries of Y lying on the 0th row and in columns between 1 and n^2, in their natural order. The adjoint operator Arrow acts on an n^2+1-dimensional vector W and produces an $(n^2+1)\times(n^2+1)$ matrix $\mathrm{Arrow}(W)$

$$\mathrm{Arrow}(W) = \begin{pmatrix} w_0 & \frac{1}{2}W_{1:n^2}^{\top} \\ \frac{1}{2}W_{(1:n^2)} & \mathrm{Diag}(W_{1:n^2}) \end{pmatrix},$$

where $W_{(1:n^2)}$ is the n^2-dimensional vector obtained from W by removing its first entry w_0. Furthermore, we are going to consider an $(n^2+1)\times(n^2+1)$ matrix Y as composed of its first row $Y_{(0\cdot)}$, of its first column $Y_{(\cdot,0)}$, and of n^2 submatrices of size $n \times n$ each, which are arranged in an $n \times n$ array of $n \times n$ matrices and produce its remaining $n^2 \times n^2$ block (this is similar to the structure of a Kronecker product of two $n \times n$ matrices. The entry $y_{\alpha\beta}$, $1 \le \alpha, \beta \le n^2$, will be also denoted by $y_{(ij)(kl)}$, with $1 \le i, j, k, l \le n$, where $\alpha = (i-1)\,n + j$ and $\beta = (k-1)\,n + l$. Hence, $y_{(ij)(kl)}$ is the element with coordinates (j, l) within the $n \times n$ block with coordinates (i, k).

With these formal conventions let us define the so-called *block-0-diagonal* and *off-0-diagonal* operators, acting on an $(n^2+1)\times(n^2+1)$ matrix Y, and denoted by b^0 diag and o^0 diag, respectively. $b^0\,\mathrm{diag}(Y)$ and $o^0\,\mathrm{diag}(Y)$ are $n \times n$ matrices given as follows:

$$b^0\,\mathrm{diag}(Y) = \sum_{k=1}^{n} Y_{(k\cdot)(k\cdot)}\,,$$

$$o^0\,\mathrm{diag}(Y) = \sum_{k=1}^{n} Y_{(\cdot,k),(\cdot,k)}\,,$$

where, for $1 \le k \le n$, $Y_{(k\cdot)(k\cdot)}$ is the kth $n \times n$ matrix on the diagonal of the $n \times n$ array of matrices, defined as described above. Analogously, $Y_{(\cdot,k),(\cdot,k)}$ is an $n \times n$ matrix consisting of the diagonal elements sitting on the position (k, k) of the $n \times n$ matrices (n^2 matrices altogether) which form the $n^2 \times n^2$ lower right block of matrix Y. The corresponding adjoint operators B^0 Diag and O^0 Diag act on an $n \times n$ matrix S and produce $(n^2+1)\times(n^2+1)$ matrices as follows:

$$B^0\,\mathrm{Diag} = \begin{pmatrix} 0 & 0 \\ 0 & I \otimes S \end{pmatrix},$$

$$O^0\,\mathrm{Diag} = \begin{pmatrix} 0 & 0 \\ 0 & S \otimes I \end{pmatrix}.$$

Finally, let us denote by e_0 the n^2+1-dimensional unit vector with first component equal to 1 and all other components equal to 0, and let R be the $(n^2+1)\times(n^2+1)$ matrix given by

$$R = \begin{pmatrix} n & -u^{\top}\otimes u^{\top} \\ -u\otimes u & I \otimes E \end{pmatrix} + \begin{pmatrix} n & -u^{\top}\otimes u^{\top} \\ -u\otimes u & E \otimes I \end{pmatrix},$$

where E is the $n \times n$ matrix of all ones. With these notations, a semidefinite relaxation for $\mathrm{QAP}_{\mathcal{I}}$ is given as follows

$$\mathrm{QAP}_{\mathrm{R0}} \begin{cases} \min & \mathrm{tr}(KY) \\ \text{s.t.} & b^0\,\mathrm{diag}(Y) = I, \\ & o^0\,\mathrm{diag}(Y) = I, \\ & \mathrm{arrow}(Y) = e_0, \\ & \mathrm{tr}(RY) = 0, \\ & Y \succeq 0, \end{cases}$$

where $\preccurlyeq$ is the so-called *Löwner partial order*, that is, $A \preccurlyeq B$ if and only if $B - A \succcurlyeq 0$, that is $B - A$ is positive semidefinite. In [77] it was shown that an equivalent formulation for the considered QAP is obtained

from $\mathrm{QAP}_{\mathrm{R0}}$ by imposing one additional condition on the matrix Y, namely, the rank-one condition.

Exact Solution Methods

Several exact solution approaches for solving the QAP will be presented in this section. Specifically the exact algorithms that have been used for the QAP are dynamic programming, cutting plane algorithms, and branch and bound which appears to be the most successful one.

Branch and Bound

Branch and bound algorithms appear to be the most efficient exact algorithms for solving the QAP. For the QAP there are three types of branch and bound algorithms, namely:

- *Single assignment algorithms* ([60,88]).
- *Pair assignment algorithms* ([59,86,105]).
- *Relative positioning algorithm* ([97]).

All of the above algorithms work by iterative constructing an optimal permutation starting from an empty permutation. The single assignment algorithms seem to be the most efficient and the pair assignment algorithms do not have favorable computational results.

We will now describe a recent branch and bound algorithm for the QAP, that was proposed in [111]. In the description that follows we will consider the Koopmans–Beckmann formulation of the QAP. First let us define the necessary notation used in describing the branch and bound algorithm. A partial permutation for the set of integers $\mathcal{S}_n = \{1, \ldots, n\}$ is denoted by

$$\phi_k := \begin{pmatrix} 1 & 2 & \cdots & k \\ \phi_k(1) & \phi_k(2) & \cdots & \phi_k(k) \end{pmatrix}$$

where $k \leq n$. From now we will write $\phi_k = (\phi_k(1), \phi_k(2), \ldots, \phi_k(k))$ for short. An assignment of a facility i to a location j will be denoted by $i \to j$, while if i must never be assigned to j we will write $i \nrightarrow j$. Note that ϕ_k is essentially a partial assignment of facilities to locations. If we want to add an extra assignment to some ϕ_k, say $k+1 \to j$, we will write $\phi_{k+1} = \phi_k \cup k+1 \to j$, thereby $\phi_{k+1}(i) = \phi_k(i)$ for $i = 1, \ldots, k$, and $\phi_{k+1}(k+1) = j$. Given some ϕ_k let its range be $Q_k := \{\phi_k(i) : i = 1, \ldots, k\}$, and define the sets of nonpermissible assignments to be $E_{k+1} := \{j \in \mathcal{S}_n \setminus Q_k : k+1 \nrightarrow j\}$. Given an instance QAP$(F, D, B)$, a pair of ϕ_k and E_{k+1} completely defines a subproblem P_i as

$$\mathrm{P}_i \begin{cases} \min\limits_{\phi \in S_n} & \sum\limits_{i}^{n}\sum\limits_{j}^{n} f_{ij} d_{\phi(i)\phi(j)} + \sum\limits_{i}^{n} b_{i\phi(i)} \\ \text{s.t.} & \phi(i) = \phi_k(i), \quad i = 1, \ldots, k, \\ & \phi(k+1) \notin E_{k+1}. \end{cases}$$

The original problem P_0 is obtained for an empty partial permutation ϕ_0 and $E_1 = \emptyset$. For each P_i a lower bound $g(\mathrm{P}_i)$ can be computed, using any of the lower bounds described previously, and let the optimal solution to P_i be denoted by $f(\mathrm{P}_i)$. In the branch and bound algorithm, a forest of n binary trees is constructed, where each node of the tree corresponds to a partial subproblem P_i. The *branching* process is as follows. Given a node P_i (i. e. a subproblem) defined by some ϕ_k and E_{k+1}, two descendant nodes are created, the left child P_i^l and the right child P_i^r. For P_i^l we set $\phi_{k+1}^l = \phi_k \cup k+1 \to j$ for some $j \notin E_k$ and $E_{k+2}^l = \emptyset$, while for P_i^r we set $\phi_k^r = \phi_k$ and $E_{k+1}^r = E_{k+1} \cup j$. A node which has ϕ_k with $k = n-1$ cannot decomposed further, and it is called a terminal node. Immediately we can identify the following properties

- $g(\mathrm{P}_i) \leq f(\mathrm{P}_i)$ for any node P_i,
- $g(\mathrm{P}_i) = f(\mathrm{P}_i)$ if P_i is a terminal node,
- $g(\mathrm{P}_j) \geq g(\mathrm{P}_i)$ if P_j has descended from P_i.

A node defined by some ϕ_k and E_{k+1} will have two terminal nodes as children if $k = n-2$. Moreover, for any node $|E_{k+1}| + k \leq n-1$, while if equality holds then there is only one $j \notin E_{k+1}$ and only one left child is generated with $\phi_{k+1} = \phi_k \cup k+1 \to j$ and $E_{k+2} = \emptyset$.

The branch and bound algorithm in [111] starts by computing an upper bound solution to the original subproblem P_0 by means of a heuristic (cf. also ▶ Heuristic search). Let the corresponding permutation be $\phi = (\phi(1), \phi(2), \ldots, \phi(\mathrm{n}))$. Note that during the process of the algorithm the upper bound is continuously updated whenever a better feasible solution is found. Then n nodes are created, where for each P_i for $i = 1, \ldots, n$, we set $\phi_1 = \phi_0 \cup 1 \to \phi(i)$. $E_2 = \emptyset$, and $g(\mathrm{P}\}_i) = 0$. Then the following steps are performed at each iteration

1) Selection: Here we choose which node to examine next, and we choose the node with the maximum $g(\mathrm{P}_i)$.
2) Branching: Given the chosen node P_i from step 1, we create two new nodes P_i^l and P_i^r, based on

the branching scheme described previously. We set $g(P_i^r) = g(P_i)$ and we compute $g(P_i^l)$.

3) Elimination: If $g(P_i^l)$ is less than or equal to the current upper bound, then the node P_i^l is pruned, that is, marked not to be considered in step 1) in the future.
4) Termination: The algorithm stops if, and only if, there are no more nodes to be considered in step 1).

The authors in [111] applied the above described branch and bound algorithm for the QAP in conjunction with the variance reduction lower bounds described previously.

Traditional Cutting Plane Methods

Traditional cutting plane algorithms for the QAP have been developed by a different authors, [7,8,9,13,14], and [80]. These algorithms make use of mixed integer linear programming (MILP) formulations for the QAP which are suitable for Benders decomposition. In the vein of Benders, the MILP formulation is decomposed into a master problem and a subproblem, called also slave problem, where the master problem contains the original assignment variables and constraints. For a fixed assignment the slave problem is usually a linear program and hence, solvable in polynomial time. The master problem is a linear program formulated in terms of the original assignment variables and of the dual variables of the slave problem, and is solvable in polynomial time for fixed values of those dual variables. The algorithms work typically as follows. First, a heuristic is applied to generate a starting assignment. Then the slave problem is solved for fixed values of the assignment variables implied by that assignment, and optimal values of the primal and dual variables are computed. If the dual solution of the slave problem satisfies all constraints of the master problem, we have an optimal solution for the original MILP formulation of the QAP. Otherwise, at least one of the constraints of the master problem is violated. In this case, the master problem is solved with fixed values for the dual variables of the slave problem and the obtained solution is given as input to the slave problem. The procedure is then repeated until the solution of the slave problem fulfills all constraints of the master problem.

Clearly any solution of the master problem obtained by fixing the dual variables of the slave problem to some feasible values, is a lower bound for the considered QAP. On the other side, the objective function value of the QAP corresponding to any feasible setting of the assignment variables is an upper bound. The algorithm terminates when the lower and the upper bounds coincide. Generally, the time needed for the upper and the lower bounds to converge to a common value is too large, and hence these methods may solve to optimality only very small QAPs. However, heuristics derived from cutting plane approaches produce good suboptimal solutions in early stages of the search, see for example, [21] and [14].

Polyhedral Cutting Planes

Similarly to traditional cutting plane methods also polyhedral cutting planes or branch and cut algorithms (cf. also ► Integer programming: Branch and cut algorithms) make use of an LP or MILP relaxation of the combinatorial optimization problem to be solved, in our case the QAP. Additionally, polyhedral cutting plane methods make use of a class of (nontrivial) valid or facet defining inequalities known to be fulfilled by all feasible solutions of the original problem. If the solution of the relaxation is feasible for the original problem, we are done. Otherwise, some of the above mentioned valid inequalities are probably violated. In this case a 'cut' is performed, that is, one or more of the violated inequalities are added to the LP or MILP relaxation of our problem. The latter is resolved and the whole process is repeated. In the case that none of the valid inequalities is violated, but some integrality constraint is violated, the algorithm performs a branching step by fixing (feasible) integer values for the corresponding variable. The branching steps produce the search tree like in branch and bound algorithms. Each node of this tree is processed as described above by performing cuts and then by branching it, if necessary. Clearly, related elements of branch and bound algorithms like upper bounds, selection and branching rules play a role in branch and cut algorithms. Hence, such an approach combines elements of cutting plane and branch and bound methods. The main advantage of polyhedral cutting plane algorithms with respect to traditional cutting planes relies on the use of cuts which are valid for the whole polytope of the feasible solutions, and possibly facet defining. Traditional cutting planes

instead rely frequently on cuts which are not valid for the whole polytope of the feasible solutions. In this case the whole computation has to be done from scratch for different variable fixings. This requires additional running time and additional amounts of memory. Another and not less important drawback of traditional cutting plane algorithms is due to the 'weakness' of the cuts they involve. In contrast with cuts produced by facet defining inequalities, the weak cuts cannot avoid the slow convergence.

Polyhedral cutting plane methods for the QAP are not yet backed by a strong theory. However, some efforts to design branch and cut algorithms for the QAP have been made in [106] and [75]. M.W. Padberg and M.P. Rijal [106] have tested their algorithm on sparse QAP instances. The numerical results are encouraging, although the developed software is of preliminary nature, as claimed by the authors. V. Kaibel [75] has used branch and cut to compute lower bounds for QAP instances. His results are promising especially in the case where box inequalities are involved.

Heuristics

There is a large amount of research directed toward heuristic algorithms for solving the QAP. This is partially due to the fact that, although substantial improvements have been done in the development of exact algorithms for the QAP, problems of dimension $n > 20$ are still not practical to solve because of very high computer time requirements. The following types of heuristic algorithmic approaches have been applied towards the QAP:

- construction methods (CM);
- limited enumeration methods (LEM);
- improvement methods (IM);
- tabu search (TS);
- simulated annealing (SA);
- genetic algorithms (GA);
- greedy randomized adaptive search procedures (GRASP);
- ant systems (AS).

Construction Methods

Construction methods were introduced in [60]. They are iterative approaches which usually start with an empty permutation, and iteratively complete a partial permutation into a solution of the QAP by assigning some facility which has not been assigned yet to some free location.

```
PROCEDURE construction(φ0, Γ)
    φ = {};
    DO i = 1, . . . , n − 1 →
        IF (i, j) ∉ Γ →
            j = heur(i);
            update(φi, (i, j));
            Γ = Γ ∪ (i, j);
        FI;
        φ = φi;
    OD;
    RETURN(φ)
END construction;
```

Pseudocode for construction method

A generic construction method is presented in pseudocode under the name PROCEDURE construction (ϕ_0, Γ). Here ϕ_0, ϕ_1, …, ϕ_{n-1} are partial permutations, and heur(i) is some heuristic procedure that assigns facility i to some location j, and returns j. Γ is the set of already assigned pairs of facilities to locations. The procedure update constructs a permutation ϕ_i by adding the assignment (i, j) to ϕ_{i-1}. The heuristic heur(i) employed by update could be any heuristic which chooses a location j for facility i, $(i, j) \notin \Gamma$, in a greedy fashion or by applying local search. One of the oldest heuristics used in practice, the CRAFT heuristic, developed in [17], is a construction method. Another construction method which yields good results has been proposed in [100].

Limited Enumeration Methods

It has been observed that often enumeration methods (e. g. branch and bound algorithms) find good solutions in early stages of the search, and then employ a lot of time to marginally improve that solution or prove its optimality. Based on this observation, limited enumeration methods impose a limit on the enumeration process, which can be either a maximum number of iterations or time limit, to produce a heuristic solution. Another strategy which serves the same goal is to manipulate the lower bound. This can be done by increasing the lower bound if no improvement in the solution

is achieved during a large number of iterations, and would yield deeper cuts in the search tree to speed up the process. Clearly, such an approach may cut off the optimal solution and hence should be used carefully, possibly in conjunction with certain heuristics that perform elaborate searches in the feasible space.

Improvement Methods

These methods are otherwise called *local search algorithms*. For a comprehensive discussion of theoretical and practical aspects of local search in combinatorial optimization, see [1].

Basic ingredients of improvement methods are the neighborhood and the order in which the neighborhood is searched. A frequently used neighborhood for the QAP is the *k-exchange neighborhood* which we will define as follows. Let the difference between two permutations ϕ and ψ be $\delta(\phi, \psi) := \{i : \phi(i) \neq \psi(i)\}$, and define the distance between the two permutations to be $d(\phi, \psi) := |\delta(\phi, \psi)|$. The k-exchange neighborhood $N_k(\phi)$ for a permutation $\phi \in \mathcal{S}_n$ is

$$N_k(\phi) := \{\psi : \ d(\phi, \psi) \leq k, \ 2 \leq k \leq n\} \ .$$

The size of the neighborhood used in the k-exchange local search is $\binom{n}{k} = n!/k!(n-k)!$. For the QAP the most frequently used values for k are 2 and 3, with $N_2(\phi)$ producing better empirical results.

Another important ingredient of improvement methods is the order in which the neighborhood is scanned. This order can be either fixed previously or chosen at random. Given a neighborhood structure and a scanning order, a rule for the update of the current solution (from the current iteration to the subsequent one) should be chosen. The following update rules are frequently used:

- first improvement;
- best improvement;
- Heider's rule [70].

In the case of first improvement the current solution is updated as soon as the first improving neighbor solution is found. Best improvement scans the whole neighborhood and chooses the best improving neighbor solution (if such a solution exists at all). Heider's rule starts by scanning the neighborhood of the initial solution in a prespecified cyclic order. The current solution is updated as soon as an improving neighbor solution is found. The scanning of the neighborhood of the new solution starts there where the scanning of the previous one was interrupted (in the prespecified cyclic order).

Tabu Search

Tabu search was introduced in [62,63] as a technique to overcome local optimality. See [61] for a comprehensive introduction to tabu search algorithms.

Different implementations of tabu search have been proposed for the QAP, for example, a tabu search with fixed tabu list ([131]), the robust tabu search ([133]), where the size of the tabu list is randomly chosen between a maximum and a minimum value, and the reactive tabu search ([12]) which involves a mechanism for adopting the size of the tabu list. Reactive tabu search aims at improving the robustness of the algorithm. The algorithm notices when a cycle occurs, and increases the tabu list size according to the length of the detected cycle. The numerical results show that generally the reactive tabu search outperforms other tabu search algorithms for the QAP (see [12]). More recently, also parallel implementations of tabu search have been proposed, see for example, [36]. Tabu search algorithms allow a natural parallel implementation by dividing the burden of the search in the neighborhood among several processors.

Simulated Annealing

Simulated annealing exploits the analogy between combinatorial optimization problems and problems from statistical mechanics. S. Kirkpatrick, C.D. Gelatt and M.P. Vecchi [82] and V. Černý [135] were among the first authors who recognized this analogy, and showed how the Metropolis algorithm (see [96]) used to simulate the behavior of a physical many-particle system can be applied as a heuristic for the traveling salesman problem.

Burkard and Rendl [33] showed that a simulated cooling process yields a general heuristic which can be applied to any combinatorial optimization problem, as soon as a neighborhood structure has been introduced in the set of its feasible solutions. In particular, they applied simulated annealing to the QAP. Other simulated annealing (SA) algorithms for the QAP have been proposed by different authors, see for example, [136] and [40]. All these algorithms employ the 2-exchange neighborhood. They differ on the way the cooling process or the thermal equilibrium is implemented. The

numerical experiments show that the performance of SA algorithms strongly depends on the values of the control parameters, and especially on the choice of the cooling schedule.

Genetic Algorithms

The so-called genetic algorithms (GA) are a nature inspired approach for combinatorial optimization problems. The basic idea is to adapt the evolutionary mechanisms acting in the selection process in nature to combinatorial optimization problems. The first genetic algorithm for optimization problems was proposed by Holland [53] in 1975. For a good coverage of theoretical and practical issues on genetic algorithms, see [43] and [64].

A number of authors have proposed genetic algorithms for the QAP. Standard algorithms, like the one developed in [134], have difficulties to generate the best known solutions even for QAPs of small or moderate size. Hybrid approaches, such as combinations of GA techniques with tabu search as the one developed in [52] seem to be more promising. More recently another hybrid algorithm, the so-called greedy genetic algorithm proposed in [5], produced very good results on large scale QAPs from QAPLIB [31].

Greedy Randomized Adaptive Search Procedure

The greedy randomized adaptive search procedure (GRASP) was introduced in [48] and has been applied successfully to different hard combinatorial optimization problems [49,83,84,125] and among them to the QAP [94,109,110] and the BiQAP [95]. See [48] for a survey and tutorial on GRASP, and to [117] for a comprehensive presentation of the implementation of GRASP to the QAP and related problems.

GRASP is a combination of greedy elements with random search elements in a two phase heuristic. It consists of a construction phase and a local improvement phase. In the construction phase good solutions from the available feasible space are constructed, whereas in the local improvement phase the neighborhood of the solution constructed in the first phase is searched for possible improvements. A pseudocode of GRASP is shown below. The input parameters are the size RCLsize of the restricted candidate list (RCL), a maximum number of iterations, and a random seed. RCL contains the candidates upon which the sampling related to the construction of a solution in the first phase will be performed.

```
PROCEDURE
        GRASP(RCLSize,MaxIter,RandomSeed)
    InputInstance();
    DO k = 1, . . . , MaxIter →
        ConstructSolution(RCLSize,RandomSeed);
        LocalSearch(BestSolutionFound);
        UpdateSolution(BestSolutionFound);
    OD;
    RETURN BestSolutionFound
END GRASP;
```

Pseudocode for generic GRASP

Ant Systems

Ant systems (AS) is a recently developed heuristic for combinatorial optimization problems which tries to imitate the behavior of an ant colony in search for food. AS was originally introduced in [45] and [38] and has already produced good results for well known problems like the traveling salesman problem (TSP) and the QAP [39,57]. Numerical results in [39,57] show that ant systems are competitive heuristics especially for real life instances of the QAP with a few very good solutions clustered together. For randomly generated instances which have many good solutions distributed somehow uniformly in the search space, AS are outperformed by other heuristics, that is, genetic algorithms or tabu search approaches.

Related Problems

Generalizations of the QAP appeared almost as soon as the problem itself. Specifically, Lawler [88] addressed the issue of extending to cubic, quartic, and *N-adic assignments problems* in general, in the same fashion as the LAP was extended to QAP in formulation (1). For the cubic assignment problem for example, we have n^6 cost coefficients c_{ijklmp} where $i, j, k, l, m, p = 1, \ldots, n$, and the problem is then defined to be

$$\begin{cases} \min & \sum_{i,j=1}^{n} \sum_{k,l=1}^{n} \sum_{m,p=1}^{n} c_{ijklmp} x_{ij} x_{kl} x_{mp} \\ \text{s.t} & (x_{ij}) \in \mathbf{X}_n. \end{cases}$$

As it is noted [88], we can construct an $n^3 \times n^3$ matrix S containing the cost coefficients, such that the cubic

assignment problem is equivalent to the LAP

$$\begin{cases} \min & \langle S, Y \rangle \\ \text{s.t.} & Y = X \otimes X \otimes X, \\ & X \in \mathbf{X}_n. \end{cases}$$

In an analogous way the LAP can be extended to any N-adic assignment problem, by considering the solution matrix Y to be the Kronecker Nth power of a permutation matrix in $\mathbf{X}_n$. In this section several generalizations and related problems of the QAP are presented, for which real applications have been found that initiated an interest in analyzing them and proposing solution techniques.

Biquadratic Assignment Problem

A generalization of the QAP is the biquadratic assignment problem (BiQAP), which is essentially a quartic assignment problem with cost coefficients formed by the products of two four-dimensional arrays. More specifically, consider two $n^4 \times n^4$ arrays $F = (f_{ijkl})$ and $D = (d_{mpst})$. The BiQAP can then be stated as:

$$\begin{cases} \min & \displaystyle\sum_{i,j,k,l=1}^{n} \sum_{m,p,s,t=1}^{n} f_{ijkl} d_{mpst} x_{im} x_{jp} x_{ks} x_{lt} \\ \text{s.t.} & \displaystyle\sum_{i=1}^{n} x_{ij} = 1, \quad j = 1, \ldots, n, \\ & \displaystyle\sum_{j=1}^{n} x_{ij} = 1, \quad i = 1, \ldots, n, \\ & x_{ij} \in \{0, 1\}, \quad i, j = 1, \ldots, n. \end{cases}$$

The major application of the BiQAP arises in very large scale integrated (VLSI) circuit design. A detailed description of the mathematical modeling of the VLSI problem to a BiQAP can be found in [24]. Deterministic improvement methods and variants of simulated annealing and tabu search have been developed for the BiQAP in [22]. Computational experiments on test problems of size up to $n = 32$, with known optimal solutions (a test problem generator is presented in [24]), suggest that one version of simulated annealing is best among those tested. The GRASP heuristic has also been applied to the BiQAP in [95], and produced the optimal solution for all the test problems generated in [24].

Multidimensional Assignment Problems

A close relative to the class of M-adic assignment problems is that of the *multidimensional assignment problems* (MAPs), often referred to as *multi-index assignment problems*, that also arise as natural extensions from the LAP. The general formulation of the MAP is

$$\begin{cases} \min & \displaystyle\sum_{i_1=1}^{M_1} \cdots \sum_{i_N=1}^{M_N} c_{i_1 \cdots i_N} x_{i_1 \cdots i_N} \\ \text{s.t.} & \displaystyle\sum_{i_2=1}^{M_2} \cdots \sum_{i_N=1}^{M_N} x_{i_1 \cdots i_N} = 1, \\ & \quad \text{for } i_1 = 1, \ldots, M_1, \\ & \displaystyle\sum_{i_1=1}^{M_1} \cdots \sum_{i_{k-1}=1}^{M_{k-1}} \sum_{i_{k+1}=1}^{M_{k+1}} \cdots \sum_{i_N=1}^{M_N} x_{i_1 \cdots i_N} = 1, \\ & \quad \text{for } i_k = 1, \ldots, M_k, \ k = 2, \ldots, N-1, \\ & \displaystyle\sum_{i_1=1}^{M_1} \cdots \sum_{i_{N-1}=1}^{M_{N-1}} x_{i_1 \cdots i_N} = 1, \\ & \quad \text{for } i_N = 1, \ldots, M_N, \\ & x_{i_1 \cdots i_N} \in \{0, 1\} \\ & \quad \text{for all } i_1 \cdots i_N, \end{cases}$$

with n^N cost coefficients $c_{i_1 \cdots i_N}$. A feasible solution to the above problem will be an N-dimensional permutation array. Multidimensional assignment problems in their general form have found many applications as a means of solving the data association problem. More specifically, the central problem in any multitarget tracking and multisensor surveillance is the data association problem of partitioning the observations into tracks and false alarms. General classes of these problems can be formulated as multidimensional assignment problems. For a detailed description on the application of MAPs for multiple target tracking applications, as well as for solution approaches, see [101,102,118].

Various applications are also contributed to special cases of the MAP. Specifically, the five-dimensional assignment problem has been successfully used for tracking elementary particles. By solving a five-dimensional assignment problem, physicists reconstruct tracks generated by charged elementary particles produced by the large electron-positron collider (LEP) at CERN institute [119]. The 3-index assignment problem is also a special case of the MAP.

Bottleneck QAP

In the *bottleneck quadratic assignment problem* (BQAP) of size n we are given an $n \times n$ flow matrix F and an $n \times n$ distance matrix D, and wish to find a permutation $\phi \in S_n$ which minimizes the objective function

$$\max \left\{ f_{ij} d_{\phi(i)\phi(j)} : \ 1 \le i, j \le n \right\}.$$

A more general BQAP analogous to the QAP in (2) is obtained if the coefficients of the problem are of the form c_{ijkl}, $1 \le i, j, k, l \le n$:

$$\min_{\phi \in S_n} \max_{1 \le i,j \le n} c_{ij\phi(i)\phi(j)}.$$

Besides the application in backboard wiring mentioned above, the BQAP has many other applications. Basically, all QAP applications give raise to applications of the BQAP because it often makes sense to minimize the largest cost instead of the overall cost incurred by some decision. A well studied problem in graph theory which can be modeled as a BQAP is the *bandwidth problem*. In the bandwidth problem we are given an undirected graph $G = (V, E)$ with vertex set V and edge set E, and seek a labeling of the vertices of G by the numbers 1, ..., n, where $|V| = n$, such that the minimum absolute value of differences of labels of vertices which are connected by an edge is minimized. In other words, we seek a labeling of vertices such that the maximum distance of 1-entries of the resulting adjacency matrix from the diagonal is minimized, that is, the bandwidth of the adjacency matrix is minimized. It is easy to see that this problem can be modeled as a special BQAP with flow matrix equal to the adjacency matrix of G for some arbitrary labeling of vertices, and distance matrix $D = (|i - j|)$.

The BQAP is *NP*-hard since it contains the bottleneck TSP as a special case. Some enumeration algorithms to solve BQAP to optimality have been proposed in [19]. These algorithms employ a Gilmore–Lawler-like bound for the BQAP which involves in turn the solution of bottleneck linear assignment problems. The algorithm for the general BQAP involves also a threshold procedure useful to reduce to 0 as many coefficients as possible. Burkard and Fincke [27] investigated the asymptotic behavior of the BQAP and proved results analogous to those obtained for the QAP: If the coefficients are independent random variables taken from a uniform distribution on [0, 1], then the relative difference between the worst and the best value of the objective function approaches 0 with probability tending to 0 as the size of the problem approaches infinity.

The BQAP and the QAP are special cases of a more general quadratic assignment problem which can be called the *algebraic QAP* (in analogy to the algebraic linear assignment problem (LAP) introduced in [30]). If $(H, *, \prec)$ is a totally ordered commutative semigroup with composition $*$ and order relation $\prec$, the algebraic QAP with cost coefficients $c_{ijkl} \in H$ is formulated as

$$\min_{\phi \in S_n} c_{11\phi(1)\phi(1)} * \ldots * c_{1n\phi(1)\phi(n)} * \ldots * c_{nn\phi(n)\phi(n)}.$$

The study of the bottleneck QAP and more generally the algebraic QAP was the starting point for the investigation of a number of algebraic combinatorial optimization problem with coefficients taken from linearly ordered semimodules, that is, linear assignment and transportation problems, flow problems, and other. See [34] for a detailed discussion on this topic.

Other Problems Which Can Be Formulated As QAPs

There are a number of other well known combinatorial optimization problems which can be formulated as QAPs with specific coefficient matrices. Of course, since QAP is not a well tractable problem, it does not make sense to use algorithms developed for the QAP to solve these other problems. All known solution methods for the QAP are far inferior compared to any of the specialized algorithms developed for solving these problems. However, the relationship between the QAP and these problems might be of benefit for a better understanding of the QAP and its inherent complexity.

Two well studied *NP*-hard combinatorial optimization problems which are special cases of the QAP, are the *graph partitioning problem* (GPP) and the *maximum clique problem* (MCP). In GPP we are given an (edge) weighted graph $G = (V, E)$ with n vertices and a number k which divides n. We want to partition the set V into k sets of equal cardinality such that the total weight of the edges cut by the partition is minimized. This problem can be formulated as a QAP with distance matrix D equal to the weighted adjacency matrix of G, and flow matrix F obtained by multiplying with -1 the adjacency matrix of the union of k disjoint complete

subgraphs with n/k vertices each. For more informations on graph partitioning problems, see [90]. In the maximum clique problem we are again given a graph $G = (V, E)$ with n vertices and wish to find the maximum $k \leq n$ such that there exists a subset $V_1 \subseteq V$ with $|V_1| = k$, which induces a clique in G, that is, all vertices of V_1 are connected by edges of G. In this case consider a QAP with distance matrix D equal to the adjacency matrix of G and flow matrix F given as adjacency matrix of a graph consisting of a clique of size k and $n - k$ isolated vertices, multiplied by -1. A clique of size k in G exists only if the optimal value of the corresponding QAP is $-k^2$. For a review on the maximum clique problem, see [114].

The traveling salesman problem (TSP) is another well known combinatorial optimization problem which is *NP*-hard, and much research has been devoted to finding efficient algorithms that will provide near-optimal solutions. In the TSP we are given a set of cities and the distances between them, and our task is to find the optimal tour that will visit each city once and will minimize the total distance traveled. In formulating the TSP as a QAP the distance matrix D is the corresponding distance matrix of the TSP, and the flow matrix F is the adjacency matrix of a complete cycle of length n. Without loss of generality the distance matrix D is considered to be symmetric. A complete cycle or tour is then defined by a permutation ϕ. The traveling salesman problem (TSP) is a notorious *NP*-hard combinatorial optimization problem. Among the abounding literature on the TSP, [89] is a comprehensive reference.

In the *linear arrangement problem* we are given a graph $G = (V, E)$ and wish to place its vertices at the points $1, \ldots, n$ on the line so as to minimize the sum of pairwise distances between vertices of G which are joined by some edge. If we consider the more general version of weighted graphs than we obtain the *backboard wiring problem*. This is an *NP*-hard problem as mentioned in [58]. It can be formulated as a QAP with distance matrix the (weighted) adjacency matrix of the given graph, and flow matrix $F = (f_{ij})$ given by $f_{ij} = |i - j|$, for all i, j. In the *minimum weight feedback arc set problem* (FASP) a weighted digraph $G = (V, E)$ with vertex set V and arc set E is given. The goal is to remove a set of arcs from E with minimum overall weight, such that all directed cycles, so-called *dicycles*, in G are destroyed and an acyclic directed subgraph remains. Clearly, the minimum weight feedback arc set problem is equivalent to the problem of finding an acyclic subgraph of G with maximum weight. The unweighted version of the FASP, that is a FASP where the edge weights of the underlying digraph equal 0 or 1, is called the *acyclic subdigraph problem* and is treated extensively in [74]. An interesting application of the FASP is the so-called triangulation of input-output tables which arises along with input-output analysis in economics used to forecast the development of industries, see [91]. For details and a concrete description of the application of triangulation results in economics, see [41] and [122]. Since the vertices of an acyclic subdigraph can be labeled topologically, that is, such that in each arc the label of its head is larger than that of its tail, the FASP can be formulated as a QAP. The distance matrix of the QAP is the weighted adjacency matrix of G and the flow matrix $F = (f_{ij})$ is a lower triangular matrix, that is, $f_{ij} = -1$ if $i \leq j$ and $f_{ij} = 0$, otherwise. The FASP is well known to be *NP*-hard (see [58,79]).

Another well known *NP*-hard problem which can be formulated as a QAP is the *graph packing problem* (cf. [16]). The graph packing problem can be formulated as a QAP with distance matrix equal to the adjacency matrix of G_2 and flow matrix equal to the adjacency matrix of G_1. A packing of G_2 into G_1 exists if and only if the optimal value of this QAP is equal to 0. In the positive case the optimal solution of the QAP determines a packing.

QAP Problem Generators

Since the QAP is a very hard problem from a practical point of view, often heuristics are the only reasonable approach to solve it, and so far there exists no performance guarantees for any of the algorithms developed for the QAP. One possibility to evaluate the performance of heuristics and to compare different heuristics is given by QAP instances with known optimal solution. Heuristics are applied to these instances and the heuristic solution is compared to the optimal one known before hand. The instances with known optimal solution should ideally have two properties: first, they should be representative in terms of their hardness, and secondly, they should not be especially easy for any of the heuristics.

Two generators of QAP instances with known optimal solution have been proposed so far: *Palubeckis' generator* [107] and the *Li–Pardalos generator* [92].

The first method for generating QAP instances with a known optimal solution was proposed by G.S. Palubeckis [107] in 1988. The input of the Palubeckis' algorithm consists of the size n of the instance to be generated, the optimal solution (permutation) ϕ of the output instance, two control parameters w and z, where $z < w$, and the distance matrix A of an $r \times s$ grid with $rs = n$. A contains rectilinear distances also called *Manhattan distances*, that is, the distance a_{ij} between two given knots i, j lying in rows r_i, r_j and in columns c_i, c_j, respectively, is given by $a_{ij} = |r_i - r_j| + |c_i - c_j|$. The output of the algorithm is a second matrix B such that ϕ is an optimal solution of QAP(A, B). The idea is to start with a matrix B such that QAP(A, B) is a trivial instance with optimal solution ϕ. Then B is transformed such that QAP(A, B) is not any more trivial, but ϕ continues to be its optimal solution.

Palubeckis starts with a constant matrix $B = (b_{ij})$ with $b_{ij} = w$. QAP(A, B) is a trivial problem because all permutations yield the same value of the objective function and thus, are optimal solutions. Hence, also the identity permutation id is an optimal solution of QAP(A, B). Then matrix B is iteratively transformed so that it is not a constant matrix any more and the identity permutation remains an optimal solution of QAP(A, B). In the last iteration the algorithm constructs an instance QAP(A', B) with optimal solution ϕ with the help of QAP(A, B) with optimal solution the identity permutation id, by setting $A' = (a^{\phi(i)\phi(j)})$. The optimal value of QAP(A', B) equals $w \sum_{i=1}^{n} \sum_{j=1}^{n} a_{ij}$. D. Cyganski, R.F. Vaz and V.G. Virball [42] have observed that the QAP instances generated by Palubeckis' generator are 'easy' in the sense that their optimal value can be computed in polynomial time by solving a linear program.

Another generator of QAP instances with known solution has been proposed by Li and Pardalos [92]. As Palubeckis' generator, Li and Pardalos starts with a trivial instance QAP(A, B) with the identity permutation id as optimal solution and iteratively transforms A and B so that the resulting QAP instance still has the optimal solution id but is not trivial any more. The transformations are such that for all i, j, i', j', $a_{ij} \geq a_{i'j'}$ is equivalent to $b_{ij} \leq b'_{ij}$ at the end of each iteration.

If the coefficient matrices are considered as weighted adjacency matrices of graphs, each iteration transforms entries corresponding to some specific subgraph equipped with signs on the edges and hence called sign-subgraphs. The application of the Li–Pardalos algorithm with different sign-subgraphs yields different QAP generators. A number of generators involving different sign-subgraphs, for example, subgraphs consisting of a single edge, signed triangles and signed spanning trees have been tested. It is perhaps interesting and surprising that QAP instances generated by involving more complex sign-subgraphs are generally 'easier' than those generated by involving subgraphs consisting of single edges. Here a QAP instance is considered to be 'easy', if most heuristics applied to it find a solution near to the optimal one in a relatively short time. Nothing is known about the complexity of QAP instances generated by the Li–Pardalos generator, since the arguments used to analyze Palubeckis' generator do not apply in this case.

Surveys and Books

In this concluding section a list of survey articles and books which cover all the aspects of the QAP in depth is given.

One of the early survey articles is [51] where the eigenvalue based lower bounds for the QAP are introduced. The survey papers [20,112] and [25] cover every aspect of the QAP. Specifically, the article [25] is the most recent one, and the most comprehensive. A collection of articles with theoretical and algorithmic contributions for the QAP can be found in the book [113]. The book [35] has a comprehensive introduction on the QAP, and focuses on special cases of the QAP which can be solved in polynomial time. Finally the book [106] focuses on polyhedral aspects of the QAP.

See also

- ► Assignment and Matching
- ► Assignment Methods in Clustering
- ► Bi-objective Assignment Problem
- ► Communication Network Assignment Problem
- ► Complexity Theory: Quadratic Programming
- ► Feedback Set Problems
- ► Frequency Assignment Problem
- ► Generalized Assignment Problem

- ▶ Graph Coloring
- ▶ Graph Planarization
- ▶ Greedy Randomized Adaptive Search Procedures
- ▶ Linear Ordering Problem
- ▶ Maximum Partition Matching
- ▶ Quadratic Fractional Programming: Dinkelbach Method
- ▶ Quadratic Knapsack
- ▶ Quadratic Programming with Bound Constraints
- ▶ Quadratic Programming Over an Ellipsoid
- ▶ Quadratic Semi-assignment Problem
- ▶ Standard Quadratic Optimization Problems: Algorithms
- ▶ Standard Quadratic Optimization Problems: Applications
- ▶ Standard Quadratic Optimization Problems: Theory

References

1. Aarts E, Lenstra JK (eds) (1997) Local search in combinatorial optimization. Wiley, New York
2. Adams WP, Johnson TA (1994) Improved linear programming-based lower bounds for the quadratic assignment problem. In: Pardalos PM, Wolkowicz H (eds) Quadratic Assignment and Related Problems. DIMACS. Amer. Math. Soc., Providence, pp 43–75
3. Adams WP, Sherali HD (1986) A tight linearization and an algorithm for zero-one quadratic programming problems. Managem Sci 32(10):1274–1290
4. Adams WP, Sherali HD (1990) Linearization strategies for a class of zero-one mixed integer programming problems. Oper Res 38(2):217–226
5. Ahuja RK, Orlin JB, Tivari A (1995) A greedy genetic algorithm for the quadratic assignment problem. Techn Report 3826-95, Sloan School Management
6. Arora S, Frieze A, Kaplan H (1996) A new rounding procedure for the assignment problem with applications to dense graph arrangement problems. In: Proc. 37-th Annual IEEE Symp. Foundations of Computer Sci. (FOCS). IEEE, New York, pp 21–30
7. Balas E, Mazzola JB (1980) Quadratic 0-1 programming by a new linearization. In: Proc TIMS/ORSA, May 1980
8. Balas E, Mazzola JB (1984) Nonlinear programming: I. Linearization techniques. Math Program 30:1–21
9. Balas E, Mazzola JB (1984) Nonlinear programming: II. Dominance relations and algorithms. Math Program 30:22–45
10. Balas E, Qi L (1993) Linear-time separation algorithms for the three-index assignment polytope. Discrete Appl Math (1993):1–12
11. Balas E, Saltzman MJ (1989) Facets of the three-index assignment polytope. Discrete Appl Math (1989):201–229
12. Battiti R, Tecchiolli G (1994) The reactive tabu search. ORSA J Comput (1994):126–140
13. Bazaraa MS, Sherali HD (1980) Bender's partitioning scheme applied to a new formulation of the quadratic assignment problem. Naval Res Logist Quart 27:29–41
14. Bazaraa MS, Sherali HD (1982) On the use of exact and heuristic cutting plane methods for the quadratic assignment problem. J Oper Res Soc 33:991–1003
15. Birkoff G (1946) Tres observaciones sobre el algebra lineal. Univ Nac Tucuman Rev (A):147–151
16. Bollobás B (1978) Extremal graph theory. Acad. Press, New York
17. Buffa ES, Armour GC, Vollmann TE (1962) Allocating facilities with CRAFT. Harvard Business Rev 42:136–158
18. Burkard RE (1973) Die Störungsmethode zur Lösung quadratischer Zuordnungsprobleme. Oper Res Verfahren 16:84–108
19. Burkard RE (1974) Quadratische Bottleneckprobleme. Oper Res Verfahren 18:26–41
20. Burkard RE (1991) Locations with spatial interactions: the quadratic assignment problem. In: Mirchandani PB, Francis RL (eds) Discrete Location Theory. Wiley, New York
21. Burkard RE, Bönniger T (1983) A heuristic for quadratic Boolean programs with applications to quadratic assignment problems. Europ J Oper Res 13:374–386
22. Burkard RE, Çela E (1995) Heuristics for biquadratic assignment problems and their computational comparison. Europ J Oper Res 83:283–300
23. Burkard RE, Çela E, Demidenko VM, Metelski NN, Woeginger GJ (1997) Perspectives of easy and hard cases of the quadratic assignment problems. Techn Report SFB 104, Techn Univ Graz
24. Burkard RE, Çela E, Klinz B (1994) On the biquadratic assignment problem. In: Pardalos PM, Wolkowicz H (eds) Quadratic assignment and related problems. DIMACS. Amer. Math. Soc., Providence, pp 117–146
25. Burkard RE, Çela E, Pardalos PM, Pitsoulis LS (1998) The quadratic assignment problem. In: Du D-Z and Pardalos PM (eds) Handbook Combinatorial Optim., vol 3. Kluwer, Dordrecht, pp 241–337
26. Burkard RE, Çela E, Rote G, Woeginger GJ (1995) The quadratic assignment problem with an Anti-Monge and a Toeplitz matrix: Easy and hard cases. Techn Report SFB 34, Techn Univ Graz
27. Burkard RE, Fincke U (1982) On random quadratic bottleneck assignment problems. Math Program 23:227–232
28. Burkard RE, Fincke U (1983) The asymptotic probabilistic behaviour of quadratic sum assignment problems. Z Oper Res 27:73–81
29. Burkard RE, Fincke U (1985) Probabilistic asymptotic properties of some combinatorial optimization problems. Discrete Appl Math 12:21–29
30. Burkard RE, Hahn W, Zimmermann U (1977) An algebraic approach to assignment problems. Math Program 12:318–327

31. Burkard RE, Karisch S, Rendl F (1997) QAPLIB – Aquadratic assignment problem library. J Global Optim 10:391–403
32. Burkard RE, Klinz B, Rudolf R (1996) Perspectives of Monge properties in optimization. Discrete Appl Math 70:95–161
33. Burkard RE, Rendl F (1984) A thermodynamically motivated simulation procedure for combinatorial optimization problems. Europ J Oper Res 17:169–174
34. Burkard RE, Zimmermann U (1982) Combinatorial optimization in linearly ordered semimodules: A survey. Modern Applied Mathematics. North-Holland, Amsterdam, pp 392–436
35. Çela E (1998) The quadratic assignment problem: Theory and algorithms. Kluwer, Dordrecht
36. Chakrapani J, Skorin-Kapov J (1993) Massively parallel tabu search for the quadratic assignment problem. Ann Oper Res 41:327–342
37. Christofides N (1976) Worst case analysis of a new heuristic for the traveling salesman problem. Grad School Industr Admin Carnegie-Mellon Univ 338
38. Colorni A, Dorigo M, Maniezzo V (1996) The ant system: optimization by a colony of cooperating agents. IEEE Trans Syst, Man Cybern Part B 26:29–41
39. Colorni A, Maniezzo V (1998) The ant system applied to the quadratic assignment problem. IEEE Trans Knowledge and Data Enginto
40. Connolly DT (1990) An improved annealing scheme for the QAP. Europ J Oper Res 46:93–100
41. Conrad K (1971) Das quadratische Zuweisungsproblem und zwei seiner Spezialfalle. Mohr-Siebeck, Tübingen
42. Cyganski D, Vaz RF, Virball VG (1994) Quadratic assignment problems with the Palubeckis' algorithm are degenerate. IEEE Trans Circuits and Systems I 41:481–484
43. Davis L (1987) Genetic algorithms and simulated annealing. Pitman, Boston
44. Deneko VG, Woeginger GJ (1996) A solvable case of the quadratic assignment problem. Techn Report SFB 88,Inst Math, Techn Univ Graz
45. Dorigo M (1992) Optimization, learning, and natural algorithms. PhD Thesis, Dip. Elettronica e Informazione Politecn. Milano. (In Italian)
46. Dyer ME, Frieze AM, McDiarmid CJH (1986) On linear programs with random costs. Math Program 35:3–16
47. Edwards CS (1980) A branch and bound algorithm for the Koopmans–Beckman quadratic assignment problem. Math Program Stud 13:35–52
48. Feo TA, Resende MGC (1995) Greedy randomized adaptive search procedures. J Global Optim 6:109–133
49. Feo TA, Resende MGC, Smith SH (1994) A greedy randomized adaptive search procedure for maximum independent set. Oper Res 42:860–878
50. Finke G, Burkard RE, Rendl F (1984) Eigenvalue approach to quadratic assignment problems. In: 5th Symp. Oper. Res., 1984
51. Finke G, Burkard RE, Rendl F (1987) Quadratic assignment problems. Ann Discret Math 31:61–82
52. Fleurent C, Ferland J (1994) Genetic hybrids for the quadratic assignment problem. In: Pardalos PM, Wolkowicz H (eds) Quadratic assignment and related problems. DIMACS. Amer. Math. Soc., Providence pp 43–75
53. Floudas CA, Pardalos PM (1990) A collection of test problems for constrained global optimization algorithms. Lecture Notes Computer Sci, vol 455. Springer, Berlin
54. Frenk JCB, van Houweninge M and, Rinnooy Kan AHG (1985) Asymptotic properties of assignment problems. Math Oper Res 10:100–116
55. Frieze AM (1974) A bilinear programming formulation of the 3-dimensional assignment problem. Math Program 7:376–379
56. Frieze AM, Yadegar J (1983) On the quadratic assignment problem. Discrete Appl Math 5:89–98
57. Gambardella LM, Taillard ED, Dorigo M (1997) Ant colonies for the QAP. Techn Report IDSIA-4-97, Ist dalle Molle Di Studi sull'Intelligenza Artificiale Lugano
58. Garey MR, Johnson DS (1979) Computers and intractability: A guide to the theory of NP-completeness. Freeman, New York
59. Gavett JW, Plyter NV (1966) The optimal assignment of facilities to locations by branch and bound. Oper Res 14:210–232
60. Gilmore PC (1962) Optimal and suboptimal algorithms for the quadratic assignment problem. SIAM J Appl Math 10:305–313
61. (1993) Tabu search. Ann Oper Res 41
62. Glover F (1989) Tabu search – Part I. ORSA J Comput 1:190–206
63. Glover F (1990) Tabu search – Part II. ORSA J Comput 2:4–32
64. Goldberg DE (1989) Genetic algorithms in search, optimization, and machine learning. Addison-Wesley, Reading
65. Hadley SW (1989) Continuous optimization approaches for the quadratic assignment problem. PhD Thesis, Univ. Waterloo
66. Hadley SW, Rendl F, Wolkowicz H (1990) Bounds for the quadratic assignment problem using continuous optimization techniques. Integer Programming and Combinatorial Optimization. Univ. Waterloo Press, Waterloo, pp 237–248
67. Hadley SW, Rendl F, Wolkowicz H (1992) A new lower bound via projection for the quadratic assignment problem. Math Oper Res 17(3):727–739
68. Hadley SW, Rendl F, Wolkowicz H (1992) Nonsymmetric quadratic assignment problems and the Hoffman–Wielandt Inequality. Linear Alg & its Appl 58:109–124
69. Hardy GG, Littlewood JE, Polya G (1952) Inequalities. Cambridge Univ. Press, Cambridge
70. Heider CH (1972) A computationally simplified pair exchange algorithm for the quadratic assignment problem. Paper 101, Center Naval Anal, Arlington

71. Jansen B (1993) A note on lower bounds for the QAP. Techn Report (dec), Delft Univ Techn Math and Computer Sci
72. Johnson DS, Papadimitriou CH, Yannakakis M (1988) How easy is local search? J Comput Syst Sci 37:79–100
73. Johnson TA (1992) New linear programming-based solution procedures for the quadratic assignment problem. PhD Thesis, Clemson Univ.
74. Jünger M (1985) Polyhedral combinatorics and the acyclic subdigraph problem. Heldermann, Berlin
75. Kaibel V (1997) Polyhedral combinatorics of the quadratic assignment problem. PhD Thesis, Univ. Köln
76. Karisch SE (1995) Nonlinear approaches for quadratic assignment and graph partition problems. PhD Thesis, Techn. Univ. Graz
77. Karisch SE, Rendl F, Wolkowicz H, Zhao Q (1998) Semidefinite programming relaxations for the quadratic assignment problem. J Combin Optim 2(1):71–109
78. Karisch SE, Rendl F, Wolkowicz H (1994) Trust regions and the quadratic assignment problem. In: Pardalos PM, Wolkowicz H (eds) Quadratic assignment and related problems. DIMACS. Amer. Math. Soc., Providence, pp 199–220
79. Karp R (1972) Reducibility among combinatorial problems. In: Miller RE, Thatcher JW (eds) Proc. Complexity of Computer Computations. Plenum, New York, pp 85–104
80. Kaufman L, Broeckx F (1978) An algorithm for the quadratic assignment problem using Benders' decomposition. Europ J Oper Res 2:204–211
81. Kernighan B, Lin S (1972) An efficient heuristic procedure for partitioning graphs. Bell Systems J 49:291–307
82. Kirkpatrick S, Gelatt CD, Vecchi MP (1983) Optimization by simulated annealing. Science 220:671–680
83. Klincewicz JG (1992) Avoiding local optima in the p-hub location problem using tabu search and GRASP. Ann Oper Res 40:283–302
84. Klincewicz JG, Rajan A (1992) Using GRASPto solve the component grouping problem. Techn Report, AT&T Bell Lab
85. Koopmans TC, Beckmann MJ (1957) Assignment problems and the location of economic activities. Econometrica 25:53–76
86. Land AM (1963) A problem of assignment with interrelated costs. Oper Res Quart 14:185–198
87. Laporte G, Mercure H (1988) Balancing hydraulic turbine runners: A quadratic assignment problem. Europ J Oper Res 35:378–382
88. Lawler EL (1963) The quadratic assignment problem. Managem Sci 9:586–599
89. Lawler EL, Lenstra JK, Rinnooy Kan AHG, Shmoys DB (1985) The traveling salesman problem: A guided tour of combinatorial optimization. Wiley, New York
90. Lengauer T (1990) Combinatorial algorithms for intergrated circuit layout. Wiley, New York
91. Leontief W (1966) Input–output economics. Oxford Univ. Press, Oxford
92. Li Y, Pardalos PM (1992) Generating quadratic assignment test problems with known optimal permutations. Comput Optim Appl 1(2):163–184
93. Li Y, Pardalos PM, Ramakrishnan KG, Resende MGC (1994) Lower bounds for the quadratic assignment problem. Ann Oper Res 50:387–410
94. Li Y, Pardalos PM, Resende MGC (1994) A greedy randomized adaptive search procedure for the quadratic assignment problem. In: Pardalos PM, Wolkowicz H (eds) Quadratic assignment and related problems. DIMACS 16. Amer. Math. Soc., Providence, pp 237–261
95. Mavridou T, Pardalos PM, Pitsoulis LS, Resende MGC (1998) A GRASPfor the biquadratic assignment problem. Europ J Oper Res 105(3):613–621
96. Metropolis N, Rosenbluth A, Rosenbluth M, Teller A, Teller E (1953) Equations of state calculations by fast computing machines. J Chem Phys 21:1087–1092
97. Mirchandani PB, Obata T (1979) Locational decisions with interactions between facilities: the quadratic assignment problem a review. Working Paper May Ps-79-1, Rensselaer Polytechnic Inst, Troy, New York
98. Mirsky L (1956) The spread of a matrix. Mathematika 3:127–130
99. Mosevich J (1986) Balancing hydraulic turbine runners – a discrete combinatorial optimization problem. Europ J Oper Res 26:202–204
100. Muller-Merbach H (1970) Optimale Reihenlorgen. Springer, Berlin
101. Murphey RA, Pardalos PM, Pitsoulis L (1997) A greedy randomized adaptive search procedure for the multitarget multisensor tracking problem. In: Pardalos PM, Du D-Z (eds) Network Design: Connectivity and Facility Location. DIMACS 40. Amer. Math. Soc., Providence, pp 277–302
102. Murphey RA, Pardalos PM, Pitsoulis L (1998) A parallel GRASP for the data association multidimensional assignment problem. Parallel Processing of Discrete Problems. In: IMA vol Math Appl, vol 106. Springer, Berlin, pp 159–180
103. Murthy KA, Pardalos PM, Li Y (1992) A local search algorithm for the quadratic assignment problem. Informatica 3(4):524–538
104. Murty KG (1968) An algorithm for ranking all the assignments in order of increasing cost. Oper Res 16:682–287
105. Nugent CE, Vollmann TE, Ruml J (1969) An experimental comparison of techniques for the assignment of facilities to locations. J Oper Res 16:150–173
106. Padberg MW, Rijal MP (1996) Location, scheduling, design and integer programming. Kluwer, Dordrecht
107. Palubetskis GS (1988) Generation of quadratic assignment test problems with known optimal solutions. Zh Vychisl Mat Mat Fiz 28(11):1740–1743. (In Russian.)

108. Papadimitriou CH, Wolfe D (1985) The complexity of facets resolved. In: Proc Foundations Computer Sci, pp 74–78
109. Pardalos PM, Pitsoulis LS, Resende MGC (1995) A parallel GRASP implementation for solving the quadratic assignment problem. In: Ferreira A and Rolim JDP (eds) Parallel Algorithms for Irregular Problems: State of the Art. Kluwer, Dordrecht, pp 115–133
110. Pardalos PM, Pitsoulis LS, Resende MGC (1997) Algorithm 769: FORTRAN subroutines for approximate solution of sparse quadratic assignment problems. ACM Trans Math Softw 23:196–208
111. Pardalos PM, Ramakrishnan KG, Resende MGC, Li Y (1997) Implementation of a variable reduction based lower bound in a branch and bound algorithm for the quadratic assignment problem. SIAM J Optim 7(1):280–294
112. Pardalos PM, Rendl F, Wolkowicz H (1994) The quadratic assignment problem: A survey and recent developments. In: Pardalos PM, Wolkowicz H (eds) Quadratic assignment and related problems. DIMACS vol 16. Amer. Math. Soc., Providence, pp 1–42
113. Pardalos PM, Wolkowicz H (1994) Quadratic assignment and related problems. DIMACS, vol 16. Amer. Math. Soc., Providence
114. Pardalos PM, Xue Jue (1994) The maximum clique problem. J Global Optim 4:301–328
115. Pierskalla WP (1967) The tri-substitution method for the three-multidimensional assignment problem. CORS J 5:71–81
116. Pierskalla WP (1968) The multidimensional assignment problem. Oper Res 16:422–431
117. Pitsoulis LS (1998) Algorithms for nonlinear assignment problems. PhD Thesis, Dept. Industr. Systems Engin., Univ. Florida
118. Poore AB, Rijavec N (1994) Partitioning multiple data sets: multidimensional assignments and Lagrangian relaxation. In: Pardalos PM, Wolkowicz H (eds) Quadratic assignment and related problems. DIMACS vol 16. Amer. Math. Soc., Providence, pp 317–342
119. Pusztaszeri J, Rensing PE, Liebling TM (1995) Tracking elementary particles near their primary vertex: A combinatorial approach. J Global Optim 16:422–431
120. Qi L, Balas E, Gwan G (1994) A new facet class and a polyhedral method for the three-index assignment problem. In: Du D-Z (ed) Advances in Optimization. Kluwer, Dordrecht, pp 256–274
121. Queyranne M (1986) Performance ratio of heuristics for triangle inequality quadratic assignment problems. Oper Res Lett 4:231–234
122. Reinelt G (1985) The linear ordering problem: Algorithms and applications Res and Exposition in Math, vol 8. Heldermann, Berlin
123. Rendl F (1985) Ranking scalar products to improve bounds for the quadratic assignment problem. Europ J Oper Res 20:363–372
124. Rendl F, Wolkowicz H (1992) Applications of parametric programming and eigenvalue maximization to the quadratic assignment problem. Math Program 53:63–78
125. Resende MGC, Pitsoulis LS, Pardalos PM (1997) Approximate solution of weighted MAX-SAT problems using GRASP. In: Pardalos PM Resende MGC, Du DZ (eds) The Satisfiability Problem. DIMACS vol 35. Amer. Math. Soc., Providence, pp 393–405
126. Rhee WT (1989) A note on asymptotic properties of the quadratic assigment problem. Oper Res Lett 4:197–200
127. Rhee WT (1991) Stochastic analysis of the quadratic assignment problem. Math Oper Res 2:223–239
128. Roucairol C (1979) A reduction method for quadratic assignment problems. Oper Res Verfahren 32:183–187
129. Sahni S, Gonzalez T (1976) P-complete approximation problems. J ACM 23:555–565
130. Schäffer AA, Yannakakis M (1991) Simple local search problems that are hard to solve. SIAM J Comput 20:56–87
131. Skorin-Kapov J (1990) Tabu search applied to the quadratic assignment problem. ORSA J Comput 2(1):33–45
132. Szpankowski W (1995) Combinatorial optimization problems for which almost every algorithm is asymptotically optimal! Optim 33:359–367
133. Taillard E (1991) Robust tabu search for the quadratic assignment problem. Parallel Comput 17:443–455
134. Tate DM, Smith AE (1995) A genetic approach to the quadratic assignment problem. Comput Oper Res 22:73–83
135. Černý V (1985) Thermodynamical approach to the traveling salesman problem: An efficient simulation algorithm. J Optim Th Appl 45:41–51
136. Wilhelm MR, Ward TL (1987) Solving quadratic assignment problems by simulated annealing. IEEE Trans 19(1):107–119
137. Zhao Q (1996) Semidefinite programming for assignment and partitioning problems. PhD Thesis, Univ. Waterloo

Quadratic Fractional Programming: Dinkelbach Method

ANDREW T. PHILLIPS
Computer Sci. Department,
University Wisconsin–Eau Claire, Eau Claire, USA

MSC2000: 90C32

Article Outline

Keywords

Global optimization; Fractional programming; Dinkelbach method

Given two continuous functions $f : \mathbf{R}^n \to \mathbf{R}$ and $g\mathbf{R}^n \to \mathbf{R}$ defined on a polyhedral set $S \subseteq \mathbf{R}^n$ such that $g(x) > 0$ for all $x \in S$, the *fractional programming problem* is to find some point x^* which satisfies

$$\frac{f(x^*)}{g(x*)} = \max_{x \in S} \frac{f(x)}{g(x)} \,. \tag{1}$$

Applications and algorithms for fractional programs have been treated in considerable detail since the early work of J.R. Isbell and W.H. Marlow [5]. Included among the many applications are portfolio selection, stock cutting, game theory, and numerous decision problems in management science. See [3] for work known to up to 1971 and [1,4,12,13] for the most recent surveys.

If $f(x)$ is concave and nonnegative and $g(x)$ and S are convex (and S is bounded), then (1) is called a concave-convex fractional program. It was shown in [10] that such problems can be solved by a single concave problem using a simple variable transformation. This provides an efficient approach for solving a limited class of fractional programming problems. Unfortunately, even in some of the simplest cases (for example when $f(x)$ and $g(x)$ are quadratic) a new constraint, which may be nonlinear, must be added (to the transformed feasible region), and the transformed problem becomes very difficult to solve. In addition, if the problem is not concave-convex initially, then the transformation does not even necessarily yield a concave problem. In fact, in the most general case, Eq (1) may have many local maxima which are different from the optimal one, and hence determining the global maximum is very difficult (i. e., *NP*-hard).

A different and more recent method is to replace the nonlinear functions by suitable linear underestimators and then obtain the global optimum by a vertex ranking procedure. This method, due to P.M. Pardalos [6], is applicable only when $f(x)$ is a convex quadratic function and $g(x)$ is linear (hence the ratio is quasiconvex).

Another well-known approach, and one of the oldest, is to consider the global optimization problem

$$\max_{x \in S} f(x) - \lambda g(x) \,, \tag{2}$$

where $\lambda \in \mathbf{R}$ is a constant. This 'parametric' approach, which was first proposed by W. Dinkelbach [2], generates a sequence of values λ_i that converges to the global optimum function value [11]. This method has since then been applied to many specific types of fractional programs including the concave-convex type, but very little work has been done to solve fractional programs where the ratio of two concave, two convex, or the ratio of a convex and a concave function is to be maximized. In addition, this method does not provide a sequence of improving upper bounds, and hence even though the sequence λ_i may be converging to the global optimum function value, no bound on the error is available at any iteration.

The method discussed here improves Dinkelbach's algorithm by providing a means for obtaining a sequence of improving upper bounds which, along with the corresponding sequence of improving lower bounds, will provide a bound on the error at each iteration of the solution procedure. In addition, both the sequence of lower bounds and the sequence of upper bounds converge to the global optimum function value at a 'superlinear' rate. This algorithm is also appropriate for the class of quadratic fractional programs (i. e., one or both of $f(x)$ and $g(x)$ are quadratic) where the ratio may involve concave, convex, or even indefinite terms. It combines Dinkelbach's approach with a method guaranteed to solve linearly constrained quadratic programming problems regardless of the definiteness of the quadratic from [8].

Two algorithms which are similar to the one presented here are given in [4,11]. *Schaible's method* [11] first computes a sequence of improving upper and lower bounds using an efficient section method. *Dinkelbach's algorithm* is then started as soon as the section method achieves a set of bounds that differ by some pre-specified tolerance. The algorithm presented here differs from Schaible's method in that the upper and lower bounds are continuously improving throughout the procedure. Nevertheless, in both algorithms the se-

quence of upper and lower bounds converges superlinearly.

Likewise, [4] presents a variety of related algorithms which also provide upper and lower bounds. These algorithms combine Dinkelbach's approach with various search techniques (e. g., Newton, binary, modified binary). The result is a set of related algorithms with convergence rates that vary depending on the search technique employed. T. Ibaraki [4] also provides a collection of computational results for the fractional knapsack problem and quadratic fractional programs.

Problem Formulation and Properties

The fundamental result which relates the global optimization problem (2) to the general fractional programming problem (1) is as follows: x^* solves the fractional programming problem (1) if and only if x^* solves the global optimization problem (2) with constant $\lambda^* = f(x^*)/g(x^*)$.

Dinkelbach's original iterative algorithm is based on this theorem and can be described as follows:

1	Select some $x^{(0)} \in S$. Set $\lambda^{(0)} = f(x^{(0)})/g(x^{(0)})$ and $k = 0$.
2	Solve the constrained global optimization problem (2) to get the optimal solution point $x^{(k+1)}$.
3	IF $f(x^{(k+1)}) - \lambda^{(k)} g(x^{(k+1)}) = 0$, THEN set $x^* = x^{k+1}$, $\lambda^* = \lambda^{(k)}$, STOP.
4	IF $f(x^{(k+1)}) - \lambda^{(k)} g(x^{(k+1)}) > 0$, THEN set $\lambda^{(k+1)} = f(x^{(k+1)})/g(x^{(k+1)})$ and $k = k + 1$. Go to Step 2.

Dinkelbach(S, f, g)

The efficiency of this algorithm depends on the number of times the constrained global optimization problem must be solved, and on the time spent solving it during each iteration. Also note that a test of the form $f(x^{(k+1)}) - \lambda^{(k)} g(x^{(k+1)}) < 0$ is not necessary since, for any fixed k,

$$\begin{aligned} f(x^{(k+1)}) - \lambda^{(k)} g(x^{(k+1)}) &= \max_{x \in S} f(x) - \lambda^{(k)} g(x) \\ &\geq f(x^{(k)}) - \lambda^{(k)} g(x^{(k)}) \\ &= 0 \,. \end{aligned}$$

Now consider the function $M(\lambda)$ defined as

$$M(\lambda) = \max_{x \in S} f(x) - \lambda g(x) \,. \tag{3}$$

The function $M(\lambda)$ has two interesting properties that are important in guaranteeing convergence of upper and lower bounds to λ^* and in determining the rate of this convergence. The first of these properties is that for any lower bound λ of λ^*, $M(\lambda)$ is positive, and for any upper bound λ of λ^*, $M(\lambda)$ is negative. Secondly, the function $M(\lambda)$ is convex. That is,

1) $M(\lambda) > 0$ for all $\lambda < \lambda^*$, and $M(\lambda) < 0$ for all $\lambda > \lambda^*$; and
2) $M(\lambda)$ is convex.

The sequence of iterates $\lambda^{(0)}, \lambda^{(1)}, \ldots$ generated by the algorithm Dinkelbach(S, f, g) is strictly monotone increasing, and satisfy $M(\lambda^{(i)}) > 0$ for $i = 0, 1, \ldots$ [2]. Hence, by the properties of $M(\lambda)$ listed above, they provide a strictly monotone increasing sequence of lower bounds for λ^*.

Bounds and Convergence Rates

The sequence of lower bounds $\lambda^{(i)}$ converges superlinearly to $\lambda^* \equiv f(x^*) : g(x^*) \rightarrow$ where x^* is any optimal solution for (1) as shown in [7]. However, as it now stands, the algorithm Dinkelbach(S, f, g) does not provide upper bounds on the global optimum function value λ^*. One way to obtain an initial upper bound is to solve the following two problems:

$$\max_{x \in S} f(x) \tag{3a}$$

to get the optimal solution $f(x')$, and

$$\min_{x \in S} g(x) \tag{4}$$

to get the optimal solution $g(x'')$. Then an initial upper bound is clearly given by $\gamma^{(-1)} \equiv f(x')/g(x'')$. In fact, according to the properties of M (part 1), any $\gamma \in \mathbf{R}$ satisfying $M(\gamma) < 0$ would also be an upper bound. Hence, if we define

$$\gamma^{(n)} \equiv \gamma^{(n-1)} - M(\gamma^{(n-1)}) \cdot \left(\frac{\gamma^{(n-1)} - \lambda^{(n)}}{M(\gamma^{(n-1)}) - M(\lambda^{(n)})} \right)$$

where $\lambda^{(n)}$ is the most recent lower bound of λ^* and $\gamma^{(n-1)}$ is the most recent upper bound of λ^*, then the new upper bound is given by $\gamma^{(n)}$. As Fig. 1 illustrates,

$\gamma^{(n)}$ is just the root of the line segment joining the points $(\lambda^{(n)}, M(\lambda^{(n)}))$ and $(\gamma^{(n-1)}, M(\gamma^{(n-1)}))$.

This leads to an important modification of the algorithm Dinkelbach(S, f, g):

1 Select some $x^{(0)} \in S$.
Set $\lambda^{(0)} = f(x^{(0)})/g(x^{(0)})$.
2 Solve the constrained global optimization problems (4) and (5) to get the optimal function values $f(x')$ and $g(x')$, respectively.
Set $\gamma^{(-1)} = f(x')/g(x")$ and $k = 0$.
IF $\gamma^{(-1)} - \lambda^{(0)} \leq \delta$,
THEN set $\lambda^* = \lambda^{(0)}$ and $x^* = x^{(0)}$;
STOP.
3 Solve the constrained global optimization problem

$$M(\lambda^{(k)}) = \max_{x \in S} f(x) - \lambda^{(k)} g(x) \tag{6}$$

to get the optimal solution point $x^{(k+1)}$.
4 IF $M(\lambda^{(k)}) = 0$,
THEN set $x^* = x^{(k+1)}$ and $\lambda^* = \lambda^{(k)}$;
STOP.
5 Solve the constrained global optimization problem

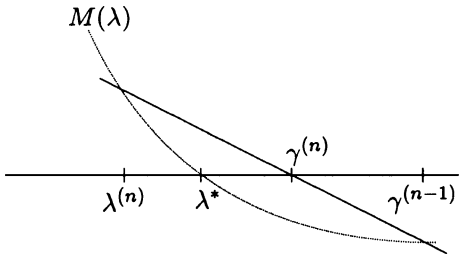
$$M(\gamma^{(k-1)}) = \max_{x \in S} f(x) - \gamma^{(k-1)} g(x) \tag{7}$$

to get the optimal solution point $y^{(k)}$.
6 IF $M(\gamma^{(k-1)}) = 0$,
THEN set $x^* = y^{(k)}$ and $\lambda^* = \gamma^{(k-1)}$;
STOP.
7 Set

$$\gamma^{(k)} = \gamma^{(k-1)} - M(\gamma^{(k-1)}) \cdot \left(\frac{\gamma^{(k-1)} - \lambda^{(k)}}{M(\gamma^{(k-1)}) - M(\lambda^{(k)})} \right).$$

8 IF $\gamma^{(k)} - \lambda^{(k)} \leq \delta$,
THEN set $\lambda^* = \lambda^{(k)}$ and $x^* = x^{(k+1)}$;
STOP.
9 Set $\lambda^{(k+1)} = f(x^{(k+1)})/g(x^{(k+1)})$ and $k = k + 1$.
Go to Step 3.

Fract(S, f, g, δ)



Quadratic Fractional Programming: Dinkelbach Method, Figure 1

Note that the parameter $\delta \geq 0$ is a user supplied stopping tolerance. The following assertion from [7] shows that the sequence iterates $\gamma^{(-1)}, \gamma^{(0)}, \gamma^{(1)}, \ldots$ is, in fact, a sequence of upper bounds on λ^*, and that the sequence is strictly monotonically decreasing:

$$\lambda^* \leq \gamma^{(i+1)} \leq \gamma^{(i)} \quad \text{for } i = -1, 0, 1, \ldots .$$

In fact, the sequence of upper bounds $\gamma^{(i)}$ also converges to λ^*, and this convergence is superlinear as well [7].

Special Cases

If the feasible set S is polyhedral and the functions $f(x)$ and $g(x)$ are either linear or quadratic, then the algorithm solves a sequence of linear or quadratic programs, respectively. In particular, if $f(x) = c^\intercal x$ and $g(x) = d^\intercal x$ then the algorithm solves the sequence of linear programs

$$\max_{x \in S} (c - \lambda^{(k)} d)^\top x \tag{8}$$

If both $f(x)$ and $g(x)$ are quadratic, i. e., $f(x) = (\frac{1}{2})\, x^\intercal Q x + c^\intercal x$ and $g(x) = (\frac{1}{2})\, x^\intercal P x + d^\intercal x$, then the algorithm solves the sequence of quadratic programs

$$\max_{x \in S} \frac{1}{2} x^\top (Q - \lambda^{(k)} P) x + (c - \lambda^{(k)} d)^\top x \,. \tag{9}$$

Notice that the matrix $(Q - \lambda^{(k)} P)$ may be indefinite, in which case the algorithm is required to find the global maximum of a linearly constrained indefinite quadratic function. Even though this is an *NP*-hard problem (e. g., when $(Q - \lambda^{(k)} P)$ is positive definite), the method developed by A.T. Phillips and J.B. Rosen [8] is guaranteed to find an ϵ-approximate global maximum (i. e., the relative error is no larger than ϵ) for any specified $\epsilon > 0$.

Furthermore, if $f(x)$ and $g(x)$ are such that $f(x) - \lambda g(x)$ is only 'partially separable', then the method developed in [9] can be used to find an ϵ-approximate global maximum for any $\epsilon > 0$. Specifically, the method in [9] is guaranteed to find solutions to the sequence of subproblems (6) and (7) if x can be partitioned into two components $x = (w, z)$ such that $f(x) - \kappa\, g(x)$ (where the constant $\kappa = \lambda^{(k)}$ or $\gamma^{(k-1)}$) can be written in the form $\phi(w) + \psi(z)$ where $\phi(w)$ is a separable convex function of w and $\psi(z)$ is a concave (but not necessarily separable) function of z. The applicability of these methods to the solution of these subproblems greatly extends the class of fractional programming problems that can be solved in practice.

See also

- ▶ Bilevel Fractional Programming
- ▶ Complexity Theory: Quadratic Programming
- ▶ Fractional Combinatorial Optimization
- ▶ Fractional Programming
- ▶ Quadratic Assignment Problem
- ▶ Quadratic Knapsack
- ▶ Quadratic Programming with Bound Constraints
- ▶ Quadratic Programming Over an Ellipsoid
- ▶ Standard Quadratic Optimization Problems: Algorithms
- ▶ Standard Quadratic Optimization Problems: Applications
- ▶ Standard Quadratic Optimization Problems: Theory

References

1. Avriel M, Diewart WE, Shaible S, Zang I (1988) Generalized concavity. Plenum, New York
2. Dinkelbach W (1967) On nonlinear fractional programming. Managem Sci 13(7):492–498
3. Grunspan M (1971) Fractional programming: A survey. Techn Report FL 50, Dept Industr Systems Engin, Univ Florida, Gainesville
4. Ibaraki T (1983) Parametric approaches to fractional programs. Math Program 26:345–362
5. Isbell JR, Marlow WH (1956) Attrition games. Naval Res Logist Quart 3:71–93
6. Pardalos PM (1986) An algorithm for a class of nonlinear fractional problems using ranking of the vertices. BIT 26:392–395
7. Pardalos PM, Phillips AT (1991) Global optimization of fractional programs. J Global Optim 1:173–182
8. Phillips AT, Rosen JB (1990) Guaranteed ϵ-approximate solution for indefinite quadratic global minimization. Naval Res Logist 37:499–514
9. Phillips AT, Rosen JB (1990) A parallel algorithm for partially separable non-convex global minimization. Ann Oper Res 25:101–118
10. Schaible S (1976) Fractional programming I, duality. Managem Sci 22(8):858–867
11. Schaible S (1976) Fractional programming II, on Dinkelbach's algorithm. Managem Sci 22(8):868–873
12. Schaible S (1981) Fractional programming: Applications and algorithms. Europ J Oper Res 12:325–338
13. Stancu-Minasian IM (1997) Fractional programming. Kluwer, Dordrecht

Quadratic Integer Programming: Complexity and Equivalent Forms

W. Art Chaovalitwongse[1], Ioannis P. Androulakis[2], Panos M. Pardalos[3]

[1] Department of Industrial and Systems Engineering, Rutgers University, Piscataway, USA

[2] Department of Biomedical Engineering, Rutgers University, Piscataway, USA

[3] Department of Industrial and Systems Engineering, University of Florida, Gainesville, USA

MSC2000: 65K05, 90C11, 90C20

Article Outline

Keywords and Phrases

Quadratic zero-one programming; Indefinite quadratic programming; Complexity; Optimality conditions

Introduction

In this paper we consider a quadratic programming (QP) problem of the following form:

$$\begin{aligned} \min \quad & f(x) = \frac{1}{2}x^{\mathrm{T}}Qx + c^{\mathrm{T}}x \\ \text{s.t.} \quad & x \in D \end{aligned} \tag{1}$$

where D is a polyhedron in $\mathbb{R}^n$, $c \in \mathbb{R}^n$. Without any loss of generality, we can assume that Q is a real symmetric $(n \times n)$-matrix. If this is not the case, then the matrix Q can be converted to symmetric form by replacing Q by $(Q + Q^{\mathrm{T}})/2$, which does not change the value of the objective function $f(x)$. Note that if Q is positive semidefinite, then Problem (1) is considered to be a convex minimization problem. When Q is negative semidefinite, Problem (1) is considered to be a concave minimization problem. When Q has at least one positive and one negative eigenvalue (i. e., Q is indefinite), Problem (1) is considered to be an indefinite quadratic programming problem. We know that in the case of convex minimization problem, every Kuhn-Tucker point is a local minimum, which is also a global minimum. In this case, there are a number of classical optimization methods that can obtain the globally optimal solutions of quadratic convex programming problems. These methods can be found in many places in the literature. In the case of concave minimization over polytopes, it is well known that if the problem has an optimal solution, then an optimal solution is attained at a vertex of D. On the other hand, the global minimum is not necessarily attained at a vertex of D for infinite quadratic programming problems. In this case, from second order optimality conditions, the global minimum is attained at the boundary of the feasible domain. In this research, without loss of generality, we are interested in developing solution techniques to solve general (convex, concave and indefinite) quadratic programming problems.

Complexity of Quadratic Programming

In this section we discuss the complexity of quadratic programming problems. The complexity analysis can give an idea of the possibility of developing efficient algorithms for solving the problem. In [10], the QP was shown to be $\mathcal{NP}$-hard in the case of a negative definite matrix Q. The QP was also proven to be $\mathcal{NP}$-hard by reduction to the satisfiability problem [11], and reduction to the knapsack feasibility problem [5]. Moreover, it has also been shown that checking local optimality for the QP itself is an $\mathcal{NP}$-hard problem [11]. In addition, checking for strict convexity (checking local optimality as part of the second order necessary conditions) in the QP was proven to be $\mathcal{NP}$-hard [8]. In fact, finding a local minimum and proving local optimality of such a solution to the QP may take exponential time. This is true even in the case of a small number of concave variables. For instance, although the matrix Q is of rank one with exactly one negative eigenvalue, the QP is still $\mathcal{NP}$-hard [9]. However, a large number of negative eigenvalues does not necessarily make the problem harder to solve. For example, consider the following problem:

$$\begin{aligned} \min \quad & \frac{1}{2}x^{\mathrm{T}}Qx + c^{\mathrm{T}}x \\ \text{s.t.} \quad & x \geq 0\,. \end{aligned}$$

If the matrix Q has $(n-1)$ negative eigenvalues, then there must be at least $(n-1)$ active constraints at the optimal solution [3]. Correspondingly, it is sufficient to solve $(n-1)$ different problems, in each case setting $(n-1)$ of the constraints to equalities, to find the optimal solution. In general, if the matrix Q has $(n-k)$ negative eigenvalues, then we are required to solve $\frac{n!}{k!(n-k)!}$ independent problems. In addition, the total computational time required to solve this problem is proportional to $\frac{k^3 c^k n!}{k!(n-k)!}$. Thus, if k is an constant and independent of n, then the computational time is bounded by a polynomial in n. On the other hand, if k grows with n, then the computational time can grow exponentially with n [3].

Equivalence Between Discrete and Continuous Problems

Before we show the equivalence between discrete and continuous programs, it is important to discuss an equivalence property between two extremum problems [2]. Therefore, we refer to the following theorem (see [2] for a proof).

Theorem 1 *Let $\bar{Z}$ and $\bar{X}$ be compact sets in $\mathbb{R}^n$, R be a closed set in $\mathbb{R}^n$, and let the following hypotheses hold.*

H_1) *$f\colon \mathbb{R}^n \to \mathbb{R}$ is a bounded function on $\bar{X}$, and there exists an open set $A \subset \bar{Z}$ and real number α, $L > 0$ such that, for any x, $y \in S$, f satisfies the following Hölder condition: $|f(x) - f(y)| \le L\|x - y\|^{\alpha}$.*

H_2) *It is impossible to find $\varphi : \mathbb{R}^n \to \mathbb{R}$ such that*

(i) φ is continuous on $\bar{X}$,

(ii) $\varphi(x) = 0$, $x \in \bar{Z}$; $\varphi(x) > 0$, $x \in \bar{X} - \bar{Z}$,

(iii) $\forall z \in \bar{Z}$, there exists a neighborhood $S(z)$ and a real $\bar{\varepsilon} > 0$ such that, for any $x \in S(z) \cap (\bar{X} - \bar{Z})$, $\varphi(x) \ge \bar{\varepsilon}\|x - z\|^{\alpha}$.

Then a real μ_0 exists such that for any real $\mu \ge \mu_0$, $\min f(x)$, $x \in \bar{Z} \cap R$ is equivalent to $\min[f(x) + \mu\varphi(x)]$, $x \in \bar{X} \cap R$.

Now we can show an equivalence between discrete and continuous programs from the following theorem [2].

Theorem 2 *Let $e^T = (1, 1, \ldots, 1)$, $\bar{Z} = \mathbb{B}^n$, $\bar{X} = \{x \in \mathbb{R}^n;\ 0 \le x \le e\}$, $R = \{x \in \mathbb{R}^n;\ g(x) \ge 0\}$. Consider the problem*

$$\begin{aligned} \min \quad & f(x) \\ s.t. \quad & g(x) \ge 0, \quad x \in B^n , \end{aligned} \tag{2}$$

and the problem

$$\begin{aligned} \min \quad & [f(x) + \mu x^T(e - x)] \\ s.t. \quad & g(x) \ge 0, \quad 0 \le x \le e . \end{aligned} \tag{3}$$

Then we suppose that f verifies assumption H_1 from Theorem 1 with $\alpha = 1$; that is, it is bounded on $\bar{X}$ and Lipschitz continuous on an open set $A \supseteq \bar{Z}$. Subsequently, there exists some $\mu_0 \in \mathbb{R}$ such that $\forall \mu < \mu_0$ Problems (2) and (3) are equivalent.

Integer Programming Problems and Complementarity Problems

The connections between integer programs and complementarity problems can be exhibited by applying KKT conditions. The results can be generalized in the quadratic programming case [4].

Theorem 3 *Let us first assume*

3a) $f\colon \mathbb{R}^n \to \mathbb{R}$, $g\colon \mathbb{R}^n \to \mathbb{R}$ are continuously differentiable functions.

3b) $g(x)$ satisfies a constraint qualification condition at x^0 to ensure that KKT conditions are validated.

Then the nonlinear programming problem

$$\begin{aligned} \min \quad & f(x) \\ s.t. \quad & g(x) \ge 0, \quad x \ge 0 , \end{aligned} \tag{4}$$

has an optimal solution x^0 if there exist $u^0 \in \mathbb{R}^n$, y^0, $v^0 \in \mathbb{R}^{\nu}$ such that (x^0, y^0, u^0, v^0) is an optimal solution to the following problem:

$$\begin{aligned} \min \quad & f(x) \\ s.t. \quad & f'(x) - y^T g'(x) - u = 0, \\ & g(x) - v = 0, \\ & y^T v = 0 \\ & x^T u = 0 \\ & x, y, u, v \ge 0 . \end{aligned} \tag{5}$$

Proof 1 *Necessity.* If x^0 is an optimal solution to Problem (4), from KKT conditions we obtain (y^0, u^0) such that

$$\begin{aligned} f'(x^0) - y^{0^{\mathrm{T}}} g(x^0) - u^0 &= 0, \\ g(x^0) &\ge 0, \\ x^{0^{\mathrm{T}}} u^0 &= 0, \\ x^0,\ y^0,\ u^0 &\ge 0 . \end{aligned}$$

Let $v^0 = g(x^0)$, then (x^0, y^0, u^0, v^0) is an optimal solution to Problem (5).

Sufficiency. The proof is trivial. □

We now generalize the results of Theorem 3 to the quadratic programming case. Consider the following problem

$$\begin{aligned} \min \quad & \frac{1}{2} x^{\mathrm{T}} Q x + c^{\mathrm{T}} x \\ \text{s.t.} \quad & Ax \ge b, \\ & x \in B^n , \end{aligned} \tag{6}$$

where Q is a symmetric matrix. Using Theorem 2, Problem (6) is equivalent to

$$\begin{aligned} \min \quad & \left[\frac{1}{2} x^{\mathrm{T}}(Q - 2\mu I)x + (c^{\mathrm{T}} + \mu e^{\mathrm{T}})x\right] \\ \text{s.t.} \quad & Ax \ge b, \\ & x \le e, \\ & x \ge 0 . \end{aligned} \tag{7}$$

Applying Theorem 3 to Problem (7), we then obtain

$$\min \quad \left[\frac{1}{2}x^{\mathrm{T}}(Q-2\mu I)x+(c^{\mathrm{T}}+\mu e^{\mathrm{T}})x\right] \tag{8}$$

$$\text{s.t.} \quad c+Qx+\mu(e-2x)-y^{\mathrm{T}}A+t=u\,, \tag{9}$$

$$b-Ax=v\,, \tag{10}$$

$$e-x=w, \tag{11}$$

$$x^{\mathrm{T}}u=0\,, \tag{12}$$

$$y^{\mathrm{T}}v=0\,, \tag{13}$$

$$t^{\mathrm{T}}w=0\,, \tag{14}$$

$$x,y,t,u,v,w\geq 0\,. \tag{15}$$

Arrange the terms in (9), we then have $Qx-2\mu x=-(c+\mu e)+y^{\mathrm{T}}A-t+u$. Consequently, (8) becomes $\min[\frac{1}{2}(c^{\mathrm{T}}+\mu e^{\mathrm{T}})x+\frac{1}{2}(b^{\mathrm{T}}y-e^{\mathrm{T}}t)$. From (12), (13), and (14), we have

$$\begin{aligned} x^{\mathrm{T}}u &= 0,\\ 0 &= y^{\mathrm{T}}v=y^{\mathrm{T}}b-y^{\mathrm{T}}Ax,\\ 0 &= t^{\mathrm{T}}w=t^{\mathrm{T}}e-t^{\mathrm{T}}x\,; \end{aligned}$$

therefore, $y^{\mathrm{T}}b=y^{\mathrm{T}}Ax$ and $t^{\mathrm{T}}e=t^{\mathrm{T}}x$. Taken all together, Problem (6) is equivalent to the following problem.

$$\begin{aligned} \min \quad & \hat{c}^{\mathrm{T}}\hat{x}\\ \text{s.t.} \quad & \hat{A}\hat{x}+\hat{u}=\hat{b},\\ & \hat{x}\hat{u}=0,\\ & \hat{x},\quad \hat{u}\geq 0\,, \end{aligned}$$

where

$$\begin{aligned} \hat{x}^{\mathrm{T}} &= (x^{\mathrm{T}},y^{\mathrm{T}},t^{\mathrm{T}}),\\ \hat{u}^{\mathrm{T}} &= (u^{\mathrm{T}},v^{\mathrm{T}},w^{\mathrm{T}}),\\ \hat{A} &= \begin{pmatrix} -Q+2\mu I & A^{\mathrm{T}} & -I\\ A & 0 & 0\\ I & 0 & 0 \end{pmatrix},\\ \hat{c}^{\mathrm{T}} &= \frac{1}{2}(c^{\mathrm{T}}+\mu e^{\mathrm{T}}+e^{\mathrm{T}},b^{\mathrm{T}},e^{\mathrm{T}}),\\ \hat{b}^{\mathrm{T}} &= (c^{\mathrm{T}},b^{\mathrm{T}},e^{\mathrm{T}})\,. \end{aligned}$$

Note that there are no restrictive assumptions made on Q, this transformation is applicable to the convex case as well as the nonconvex case.

Integer Programming Problems and Quadratic Integer Programming Problems

Integer programming is used to model a variety of important practical problems in operations research, engineering, and computer science. Consider the following linear zero-one programming problem:

$$\begin{aligned} \min \quad & c^{\mathrm{T}}x\\ \text{s.t.} \quad & Ax\leq b,\quad x_i\in\{0,1\},\quad (i=1,\ldots,n) \end{aligned}$$

where A is a real $(m\times n)$-matrix, $c\in\mathbb{R}^n$ and $b\in\mathbb{R}^m$. Let $e^{\mathrm{T}}=(1,\ldots,1)\in\mathbb{R}^n$ denote the vector whose components are all equal to 1. Then the zero-one integer linear programming problem is equivalent to the following concave minimization problem:

$$\begin{aligned} \min \quad & f(x)=c^{\mathrm{T}}x+\mu x^{\mathrm{T}}(e-x)\\ \text{s.t.} \quad & Ax\leq b,\ 0\leq x\leq e \end{aligned}$$

where μ is a sufficiently large positive integer. We know that the function $f(x)$ is concave because $-x^{\mathrm{T}}x$ is concave.

The equivalence of the two problems is based on the facts that a concave function attains its minimum at a vertex and that $x^{\mathrm{T}}(x-e)=0,\ 0\leq x\leq e$, implies $x_i=0$ or 1 for $i=1,\ldots,n$. We note that a vertex of the feasible domain is not necessarily a vertex of the unit hypercube $0\leq x\leq e$, but the global minimum is attained only when $x^{\mathrm{T}}(e-x)=0$, provided that μ is a sufficiently large number.

These transformation techniques can be applied to reduce quadratic zero-one problems to equivalent concave minimization problems. For instance, consider a quadratic zero-one problem of the following form:

$$\begin{aligned} \min \quad & f(x)=c^{\mathrm{T}}x+x^{\mathrm{T}}Qx\\ \text{s.t.} \quad & x\in\{0,1\} \end{aligned}$$

where Q is a real symmetric $(n\times n)$ matrix. Given any real number μ, let $\bar{Q}=Q+\mu I$ where I is the $(n\times n)$ unit matrix, and $\bar{c}=c-\mu e$. Because of $\bar{f}(x)=f(x)$, the above quadratic zero-one problem is equivalent to the problem:

$$\begin{aligned} \min f(x) \quad & =\bar{c}^{\mathrm{T}}x+x^{\mathrm{T}}\bar{Q}x\\ \text{s.t.} \quad & x_i\in\{0,1\},\quad (i=1,\ldots,n) \end{aligned}$$

In this case, if we choose μ such that $\bar{Q} = Q + \mu I$ becomes a negative semidefinite matrix (e. g., $\mu = -\lambda$, where λ is the largest eigenvalue of Q), then the objective function $\bar{f}(x)$ becomes concave and the constraints can be replaced by $0 \le x \le e$. Thus, this problem is equivalent to the minimization of a quadratic concave function over the unit hypercube [4].

Various Equivalent Forms of Quadratic Zero-One Programming Problems

The problem considered here is a quadratic zero-one program, which has the form

$$\begin{aligned} \min f(x) \quad &= x^{\mathrm{T}} Q x, \\ \text{s.t.} \quad &x_i \in \{0,1\}, \quad i = 1, \ldots, n, \end{aligned} \tag{16}$$

where Q is an $n \times n$ matrix [6,7]. Throughout this section the following notation will be used.

- $\{0,1\}^n$: set of n dimensional 0–1 vectors.
- $R^{n\times n}$: set of $n \times n$ dimensional real matrices.
- R^n: set of n dimensional real vectors.

In order to formalize the notion of equivalence we need some definitions.

Definition 1 The problem P is "polynomially reducible" to problem P_0 if given an instance $I(P)$ of problem P, an instance $I(P_0)$ of problem P_0 can be obtained in polynomial time such that solving $I(P)$ will solve $I(P_0)$.

Definition 2 Two problems P_1 and P_2 are called "equivalent" if P_1 is "polynomially reducible" to P_2 and P_2 is "polynomially reducible" to P_1.

Consider the following three problems:

$$\begin{aligned} P\colon \min f(x) = \quad & x^{\mathrm{T}} Q x, \quad x \in \{0,1\}^n, \\ & Q \in R^{n\times n}, \\ P_1\colon \min f(x) = \quad & x^{\mathrm{T}} Q x + c^{\mathrm{T}} x, \quad x \in \{0,1\}^n, \\ & Q \in R^{n\times n}, c \in R^n, \\ P_2\colon \min f(x) = \quad & x^{\mathrm{T}} Q x, \quad x \in \{0,1\}^n, \\ & Q \in R^{n\times n}, \\ & \sum_{i=1}^{n} x_i = k \text{ for some } k \\ \text{s.t.} \quad & 0 \le k \le n, \\ & \text{where } x = (x_1, x_2, \ldots, x_n). \end{aligned}$$

Next we show that problems P, P_1, and P_2 are all "equivalent". Then, formulation P_2 will be used in the rest of the sections.

Lemma 1 *P is "polynomially reducible" to P_1.*

Proof 2 It is very easy to see that P is a special case of P_1. □

Lemma 2 *P_1 is "polynomially reducible" to P.*

Proof 3 Problem P_1 is defined as follows: $\min f(x) = x^{\mathrm{T}} Q x + c^{\mathrm{T}} x, x \in \{0,1\}^n, Q \in R^{n\times n}, c \in R^n$. If $Q = (q_{ij})$ then let $B = (b_{ij})$ where

$$b_{ij} = \begin{cases} q_{ij} & \text{if } i \ne j \\ q_{ij} + c_i & \text{if } i = j. \end{cases}$$

Since $x_i^2 = x_i$ (because $x_i \in \{0,1\}$), we have $g(x) = x^{\mathrm{T}} B x = x^{\mathrm{T}} Q x + c^{\mathrm{T}} x$. So the following problem is equivalent to problem $P_1 : \min g(x) = x^{\mathrm{T}} B x, x \in \{0,1\}^n, B \in R^{n\times n}$. □

Using Lemma 1 and Lemma 2, it is evident that P and P_1 are "equivalent".

Lemma 3 *P_2 is "polynomially reducible" to P.*

Proof 4 Problem P_2 is as follows: $\min f(x) = x^{\mathrm{T}} Q x, \quad x \in \{0,1\}^n, \quad Q \in R^{n\times n}, \sum_{i=1}^{n} x_i = k$ for some k s.t. $0 \le k \le n$. If $Q = (q_{ij})$ then let $M = 2[\sum_{j=1}^{n} \sum_{i=1}^{n} |q_{ij}|] + 1$. Now, define the following problem P: $\min g(x) = x^{\mathrm{T}} Q x + M(\sum_{i=1}^{n} x_i - k)^2$ s.t. $x \in \{0,1\}^n, Q \in R^{n\times n}$. Let $x_b = (x_1^b, \ldots, x_n^b)$ and $x_0 = (x_1^0, \ldots, x_n^0)$ such that $\sum_{i=1}^{n} x_i^b \ne k$ and $\sum_{i=1}^{n} x_i^0 = k$, then $g(x_0) \le \frac{M-1}{2}$ as $\sum_{i=1}^{n} x_i^0 = k$, $g(x_b) \ge \frac{-(M-1)}{2} + M$ or $g(x_b) \ge \frac{M+1}{2}$ as $|\sum_{i=1}^{n} x_i^b - k| \ge 1$. Therefore, $g(x_0) < g(x_b)$ if $\sum_{i=1}^{n} x_i^b \ne k$ and $\sum_{i=1}^{n} x_i^0 = k$. Hence, if $\min g(x) = g(x_0)$ where $x_0 = (x_1^0, \ldots, x_n^0)$ then $\sum_{i=1}^{n} x_i^0 = k$. So $\min f(x) = \min g(x)$. From the above discussion, it can be easily seen that P_2 is "polynomially reducible" to P. □

The proof of Lemma 3 also illustrates how equality (knapsack) constraints in a quadratic zero-one program can be eliminated.

Lemma 4 *P is "polynomially reducible" to P_2.*

Proof 5 Let problem P be defined as follows: $\min f(x) = x^{\mathrm{T}} Q x, \quad x \in \{0,1\}^n, \quad Q \in R^{n\times n}$. Define a series of $(n+1)$ problems: $P_2(0), P_2(1), P_2(2), \cdots,$

$P_2(n)$, where $P_2(j)$ is the following problem $\min f(x) = x^{\mathrm{T}}Qx, \quad x \in \{0,1\}^n, \quad Q \in R^{n\times n}, \sum_{i=1}^{n} x_i = j$. Let the minimum of the problem $P_2(j)$ be y_j, then the minimum of problem P is easily seen to be the min $\{y_0, y_1, \ldots, y_n\}$. □

Lemma 3 and Lemma 4 imply that P and P_2 are "equivalent". Since "equivalent" is a transitive relative, P, P_1, P_2 are all "equivalent".

Complexity of Quadratic Zero-One Programming Problems

Quadratic zero-one programming is a difficult problem. We next will show that the quadratic knapsack zero-one problem in (P_2) is a NP hard problem by proving that it is equivalent to the k-clique problem. A k-clique is a complete graph with k vertices.

k-clique Problem

Given a graph G=(V, E) (V is the set of vertices and E is the set of edges), does the graph G have a k-clique as one of its subgraphs?

k-clique problem is known to be NP-complete. We will show that the k-clique problem is "polynomially reducible" to problem P_2 defined in the previous subsection.

Theorem 4 *The k-clique problem is "polynomially reducible" to P_2.*

Proof 6 Problem P_2 was defined as $\min f(x) = x^{\mathrm{T}}Qx$, s.t. $x_i \in \{0,1\}, i = 1, \cdots, n, \sum_{i=1}^{n} x_i = m$ for some $0 \le m \le n$. Given the graph $G = (V, E)$, define $Q = (q_{ij})$ such that

$$q_{ij} = \begin{cases} 0 & \text{if } (v_i, v_j) \in E \\ -1 & \text{if } (v_i, v_j) \notin E\,, \end{cases}$$

where $n = |V|, m = k$ (we are trying to find a k-clique). The meaning attached to the vector $\mathbf{x} \in \{0,1\}^n$ in problem P_2 is as follows

$$x_i = \begin{cases} 1 & \text{means that } v_i \text{ is in the clique,} \\ 0 & \text{means that } v_i \text{ is not in the clique}\,. \end{cases}$$

We can easily prove that the graph G has a k-clique if and only if $\min f(x) = -k(k-1)$. So the k-clique problem is "polynomially reducible" to P_2. □

Problem P_2 is "equivalent" to P, so problem P is also NP-hard. Therefore, as the dimension of the problem increases, the necessary CPU time to solve the problem increases exponentially.

Quadratic Zero-One Programming and Mixed Integer Programming

In this section, we consider a quadratic zero-one programming problem in the following form:

$$\begin{aligned} \min f(x) \quad &= x^{\mathrm{T}}Qx, \\ \text{s.t.} \quad &\sum_{i=1}^{n} x_i = k, \quad x \in \{0,1\}^n\,. \end{aligned} \tag{17}$$

Let Q be $n \times n$ matrix, whose each element $q_{i,j} \ge 0$. Define $x = (x_1, \ldots, x_n)$, where each x_i represents binary decision variables. We will show that the problem in (17) can be linearized as the following mixed integer programming problems. The first linearization technique is trivial and can be found elsewhere. Recently, more efficient linearization technique was introduced in [1]. In addition, the linearization technique for more general case (where $q_{i,j} \in real$) and multi-quadratic programming was also proposed in [1].

Conventional Linearization Approach

For each product $x_i x_j$ in the objective function of the problem (17) we introduce a new continuous variable, $x_{ij} = x_i x_j (i \ne j)$. Note that $x_{ii} = x_i^2 = x_i$ for $x_i \in \{0,1\}$. The equivalent mixed integer programming problem (MIP) is given by:

$$\begin{aligned} \min \quad & \sum_i \sum_j q_{ij} x_{ij} \\ \text{s.t.} \quad & \sum_{i=1}^{n} x_i = k, \\ & x_{ij} \le x_i, && \text{for } i, j = 1, \ldots, n (i \ne j) \\ & x_{ij} \le x_j, && \text{for } i, j = 1, \ldots, n (i \ne j) \\ & x_i + x_j - 1 \le x_{ij}, && \text{for } i, j = 1, \ldots, n (i \ne j) \\ & 0 \le x_{ij} \le 1, && \text{for } i, j = 1, \ldots, n (i \ne j) \end{aligned} \tag{18}$$

where $x_i \in \{0,1\}$, $i, j = 1, \ldots, n$.

The main disadvantage of this approach is that the number of additional variables we need to introduce is

$O(n^2)$, and the number of new constraints is also $O(n^2)$. The number of 0–1 variables remains the same.

A New Linearization Approach

Consider the following mixed integer programming problem:

$$\begin{aligned} \min_{x,y,s} g(s) \quad &= \sum_{i=1}^{n} s_i = e^{\mathrm{T}} s \\ \text{s.t.} \quad & \sum_{i=1}^{n} x_i = k, \\ & Qx - y - s = 0, \\ & y \le \mu(e - x), \\ & x_i \in \{0, 1\}, \quad \text{for } i = 1, \ldots, n \\ & y_i,\ s_i \ge 0, \quad \text{for } i = 1, \ldots, n\,. \end{aligned} \tag{19}$$

where Q is an $n \times n$ matrix, whose each element $q_{i,j} \ge 0$.

In [1], the mixed integer 0–1 programming problem in (19) was proved equivalent to the quadratic zero-one programming in (17). The main advantage of this approach is that we only need to introduce $O(n)$ additional variables and $O(n)$ new constraints, where the number of 0–1 variables remains the same. This linearization technique proved more robust and more efficiently solving quadratic zero-one and multi-quadratic zero-one programming problems [1].

References

1. Chaovalitwongse WA, Pardalos PM, Prokoyev OA (2004) Reduction of multi-quadratic 0–1 programming problems to linear mixed 0–1 programming problems. Oper Res Lett 32(6):517–522
2. Giannessi F, Niccolucci F (1976) Connections between nonlinear and integer programming problems. Istituto Nazionale di Alta Matematica, Symposia Mathematica 19:161–176
3. Hager WW, Pardalos PM, Roussos IM, Sahinoglou HD (1991) Active constraints, indefinite quadratic programming, and test problems. J Optim Theor Appl 68(3):499–511
4. Horst R, Pardalos PM, Thoai NV (1995) Introduction to global optimization. Kluwer, Dordrecht
5. Pardalos PM (1991) Global optimization algorithms for linearly constrained indefinite quadratic problems. Comput Math Appl 21:87–97
6. Pardalos PM, Rodgers G (1989) Parallel branch and bound algorithms for unconstrained quadratic zero-one programming. In: Sharda R et al (eds) Impact of recent computer advances on operations research. Elsevier, pp 131–143
7. Pardalos PM, Rodgers G (1990) Computational aspects of a branch and bound algorithm for quadratic zero-one programming. Comput 45:131–144
8. Pardalos PM, Schnitger G (1988) Checking local optimality in constrained quadratic programming is $\mathcal{NP}$-hard. Oper Res Lett 7(1):33–35
9. Pardalos PM, Vavasis S (1991) Quadratic programming with one negative eigenvalue is $\mathcal{NP}$-hard. J Global Optim 1:15–23
10. Sahni S (1974) Computationally related problems. SIAM J Comput 3:262–279
11. Vavasis S (1991) Nonlinear optimization: complexity issues. Oxford Univeristy Press, Oxford

Quadratic Knapsack

YASUTOSHI YAJIMA
Tokyo Institute Technol., Tokyo, Japan

MSC2000: 90C20, 90C60

Article Outline

Keywords

Quadratic programming; Knapsack constraint

The quadratic knapsack problem is one of the simplest quadratic programming problems as defined below (cf. also ► Quadratic programming with bound constraints):

$$(\mathrm{P}) \begin{cases} \min & f(x) = \frac{1}{2} x^{\top} Q x + c^{\top} x \\ \text{s.t.} & \sum_{i=1}^{n} a_i x_i = M, \\ & 0 \le x_i \le 1, \quad i = 1, \ldots, n, \end{cases}$$

where $x \in \mathbf{R}^n$ is a variable vector, $Q \in \mathbf{R}^{n \times n}$, $c \in \mathbf{R}^n$ and M is a scalar.

The problems are mainly classified by the nature of matrix Q. When the matrix Q is *positive semidefinite*, i. e., the objective function $f(x)$ is convex, problem (P) can be solved in polynomial time by the *ellipsoid algorithm* [8], and several kinds of interior point algorithms (e. g. [5,7,11], which solve general convex quadratic problems including (P) as a special case). Also, P.M. Pardalos, Y. Ye and C.G. Han [15] show a potential reduction algorithm for the special case of (P) defined below:

$$\begin{cases} \min & \frac{1}{2}x^\top Qx \\ \text{s.t.} & \sum_{i=1}^{n} x_i = 1, \\ & x_i \geq 0, \quad i = 1, \ldots, n. \end{cases}$$

In particular, when (P) has a diagonal matrix Q with positive elements, an $O(n)$ algorithm has been proposed by P. Brucker [3]. The algorithm generates the corresponding KKT condition using binary search. Pardalos and N. Kovoor [13] also propose an $O(n)$ randomized method.

The convex case is important because of its frequent appearance as a subproblem in many application areas. Among those are general convex quadratic programming [9], multicommodity network flow problems [1], resource management [2], and portfolio selection problems [10].

The problem becomes extremely difficult if $f(x)$ is not convex. S. Sahni [16] shows that the problems with the negative diagonal matrix Q are *NP*-hard (cf. also ► Computational complexity theory; ► Complexity theory), which implies that the general indefinite case is also *NP*-hard.

Let $a_1, \ldots, a_n$ and b be positive integers, and let us consider the *subset sum problem*, which finds a feasible solution of the set defined below:

$$\left\{ x : \sum_{i=1}^{n} a_i x_i = b, \quad x_i \in \{0, 1\}, \ i = 1, \ldots, n \right\}.$$

The feasibility is determined by the the following concave quadratic knapsack problem:

$$\begin{cases} \min & \sum_{i=1}^{n} x_i(1 - x_i) \\ \text{s.t.} & \sum_{i=1}^{n} a_i x_i = M, \ \ 0 \leq x_i \leq 1, \ \ i = 1, \ldots, n. \end{cases}$$

The subset sum problem is feasible if and only if the global optimum value of the corresponding quadratic knapsack problem is zero.

As we see in the above, the indefinite case arises in several combinatorial optimization problems. For example, given a graph $G(V, E)$ where $V = \{1, \ldots, n\}$ is a set of vertices and $E \subseteq V^2$ is a set of edges, find the *maximum clique* of G. This problem can be formulated in the following way:

$$\begin{cases} \min & \sum_{(i,j)\in E} -x_i x_j \\ \text{s.t.} & \sum_{i=1}^{n} x_i = 1, \\ & x_i \geq 0, \quad i = 1, \ldots, n. \end{cases}$$

If G has a maximum clique of size k, then the global maximum is $(1/k - 1)/2$. We can also formulate the *maximum independent set problem* and the *node covering problem* in a similar fashion.

One can also formulate any quadratic minimization problem over a convex hull by the quadratic knapsack problem. Consider the problem of the form:

$$\begin{cases} \min & q(z) = z^\top Mz + r^\top z \\ \text{s.t.} & z \in P, \end{cases} \tag{1}$$

where $z, r \in \mathbf{R}^m$, $M \in \mathbf{R}^{m\times m}$ and $P \subseteq \mathbf{R}^m$ is the polytope described as the convex hull of a given set of points $\{v_1, \ldots, v_n\}$. It can be verified easily that the above general quadratic problem has the following equivalent formulation

$$\begin{cases} \min & f(x) = x^\top (V^\top MV)x + r^\top Vx \\ \text{s.t.} & \sum_{i=1}^{n} x_i = 1, \\ & x_i \geq 0, \quad i = 1, \ldots, n, \end{cases} \tag{2}$$

where $V = [v_1, \ldots, v_n]$. Let z^* and x^* be optimum solutions of (1) and (2), respectively. Then we have

$$q(z^*) = f(x^*),$$

and moreover $z^* = Vx^*$.

There exist only a few algorithms for obtaining a global optimum solution for the case of the general indefinite Q. See [15] for a partitioning approach as well as an interior point method, while [4] surveys algorithms for general nonconvex quadratic problems.

The case when the objective function is separable has also been well investigated by several authors. Some practical algorithms to obtain an exact solution are reported in [6,14]. S.A. Vavasis [18] shows an $O(n(\log n)^2)$ algorithm for finding a local minimum of the problem, while K.G. Murty and S.N. Kabadi [12] show that verifying a local minimum for an indefinite quadratic problem with general constraints is *NP*-hard. Also, [17] gives an ϵ-approximation algorithm which is weakly polynomial in the problem size if the number of negative diagonal elements is fixed.

See also

- ▶ αBB Algorithm
- ▶ Complexity Theory: Quadratic Programming
- ▶ D.C. Programming
- ▶ Integer Programming
- ▶ Multidimensional Knapsack Problems
- ▶ Quadratic Assignment Problem
- ▶ Quadratic Fractional Programming: Dinkelbach Method
- ▶ Quadratic Programming with Bound Constraints
- ▶ Quadratic Programming Over an Ellipsoid
- ▶ Reverse Convex Optimization
- ▶ Standard Quadratic Optimization Problems: Algorithms
- ▶ Standard Quadratic Optimization Problems: Applications
- ▶ Standard Quadratic Optimization Problems: Theory

References

1. Ali A, Helgason R, Kennington J, Lall H (1980) Computational comparison among three multicommodity network flow algorithms. Oper Res 28(4):995–1000
2. Bitran GR, Hax AC (1981) Disaggregation and resource allocation using convex knapsack problems with bounded variables. Managem Sci 27(4):431–441
3. Brucker P (1984) An O(n) algorithm for quadratic knapsack problems. Oper Res Lett 3(3):163–166
4. Floudas CA, Visweswaran V (1995) Quadratic optimization. In: Handbook Global Optim. Nonconvex Optim. Appl., vol 2. Kluwer, Dordrecht, pp 217–269
5. Goldfarb D, Liu S (1990/1) An O(n3L) primal interior point algorithm for convex quadratic programming. Math Program A 49(3):325–340
6. Horst R, Tuy H (1993) Global optimization: Deterministic approaches, 2nd edn. Springer, Berlin
7. Kojima M, Mizuno S, Yoshise A (1991) An O ($\sqrt{n}$ L) iteration potential reduction algorithm for linear complementarity problems. Math Program A 50(3):331–342
8. Kozlov MK, Tarasov SP, Khachiyan LG (1979) Polynomial solvability of convex quadratic programming. R-Dokl 248(5):1049–1051
9. Lin YY, Pang J-S (1987) Iterative methods for large convex quadratic programs: A survey. SIAM J Control Optim 25(2):383–411
10. Markowitz HM (1952) Portfolio selection. Finance 7:77–91
11. Monteiro RDC, Adler I, Resende MGC (1990) A polynomial-time primal-dual affine scaling algorithm for linear and convex quadratic programming and its power series extension. Math Oper Res 15(2):191–214
12. Murty KG, Kabadi SN (1987) Some NP-complete problems in quadratic and nonlinear programming. Math Program 39(2):117–129
13. Pardalos PM, Kovoor N (1990) An algorithm for a singly constrained class of quadratic programs subject to upper and lower bounds. Math Program A 46(3):321–328
14. Pardalos PM, Rosen JB (1987) Constrained global optimization: Algorithms and applications. Lecture Notes Computer Sci, vol 268. Springer, Berlin
15. Pardalos PM, Ye Y, Han Chi-Geun (1991) Algorithms for the solution of quadratic knapsack problems. Linear Alg & Its Appl 152:69–91
16. Sahni S (1974) Computationally related problems. SIAM J Comput 3:262–279
17. Vavasis SA (1992) Approximation algorithms for indefinite quadratic programming. Math Program B 57(2):279–311
18. Vavasis SA (1992) Local minima for indefinite quadratic knapsack problems. Math Program A 54(2):127–153

Quadratic Programming with Bound Constraints

Pasquale L. De Angelis[1], Gerardo Toraldo[2]

[1] Naval Institute, Naples, Italy

[2] University Naples 'Federico II' and CPS, Naples, Italy

MSC2000: 90C20, 65K05

Article Outline

Keywords

Quadratic programming; Bounds constraints; Global optimization; Indefinite quadratic problems; Optimality conditions

Synonyms

QPwBC

Problem Statement

The *bound constrained quadratic problem* has the following form:

$$\min_{x\in\Omega} f(x) = \min_{x\in\Omega} \frac{1}{2}x^\top Qx + c^\top x\,, \qquad \Omega = \{x \in \mathbf{R}^n\colon\ l \le x \le u\}\,, \tag{1}$$

where $Q = (q_{ij}) \in \mathbf{R}^{n\times n}$ is an indefinite symmetric matrix and $x, c, l, u \in \mathbf{R}^n$. Here (as always in the sequel), all inequalities involving vectors are interpreted componentwise, and $\nabla f(x) = Qx + c$ is the gradient of f. The region Ω is assumed to be nonempty (i. e. $l_i \le u_i$ for each $i \in \{1, \ldots, n\}$) and may be unbounded (i. e. $l_i = -\infty$ and/or $u_i = +\infty$ for some $i \in \{1, \ldots, n\}$). The function $f(x)$ is assumed to be bounded below on Ω. For each $x \in \Omega$, the *active set* $A(x)$ is defined as:

$$A(x) = \{i\colon\ x_i \in \{l_i, u_i\}\}\,.$$

Problems of the form (1) naturally arise in a number of different applications. Moreover, QPwBC is a basic subroutine for many nonlinear programming codes, and the monotone *linear complementarity* problem can be written in the above form. For the convex case (i. e. Q positive semidefinite), which is known to be polynomially solvable [16], many efficient algorithms exist [4,5,7,9,10,12,18,36]. However, not many algorithms exist for the efficient solution of the general nonconvex problem [8,17,22,24,25,26].

From the complexity point of view, problem (1) is *NP*-hard [32], and even checking local optimality for a feasible point is *NP*-hard [20,27]. The complexity of finding a stationary point for (1) is an open question (in the concave case this problem is PLS-complete [15]). Algorithms to construct approximate solutions [33] in polynomial time exist.

Optimality Conditions

For problem (1) the classical local optimality conditions can be stated in a very special form. Moreover, there exist interesting results about global optimality which lead to efficient numerical procedures.

Local Optimality Conditions

Proposition 1 *If $x^* \in \Omega$ is a local minimum for problem (1) then:*

A) if $q_{ii} \ge 0$, then
- *i) $[\nabla f(x^*)]_i = 0$; or*
- *ii) $[\nabla f(x^*)]_i > 0$ and $x_i^* = l_i$; or*
- *iii) $[\nabla f(x^*)]_i < 0$ and $x_i^* = u_i$.*

B) if $q_{ii} < 0$, then
- *i) $[\nabla f(x^*)]_i > 0$ and $x_i^* = l_i$; or*
- *ii) $[\nabla f(x^*)]_i < 0$ and $x_i^* = u_i$.*

Proposition 1 specializes the classical *KKT stationarity conditions*, which only involve first order information, to problem (1) by taking into the account the sign of the second order pure derivatives. If x^* is *nondegenerate*, i. e.

$$(x_i^* - l_i)(x_i^* - u_i) + \left|\left[\nabla f(x^*)\right]_i\right| \ne 0$$

for each $i \in A(x^*)$, then the conditions A)–B) are sufficient for local minimization.

The following proposition states a relationship between the number of negative eigenvalues of the matrix Q and the cardinality of the active set at a stationarity point x^*.

Proposition 2 *If the matrix Q has k negative eigenvalues counting multiplicities, then at least k constraints are active at a local solution x^* of problem (1).*

Because of Proposition 2, if f is concave, the problem is bounded if and only if all upper and lower bounds are finite, and the solution can be found by checking all the vertices of Ω. Therefore the concave QPwBC problem is equivalent to a *quadratic zero-one problem* [1,22].

Global Optimality Conditions

Global optimality conditions for problem (1) can be stated in terms of copositivity [14] of the Hessian matrix.

Definition 3 An $n \times n$ matrix Q is *copositive* with respect to a polyhedral cone $\Gamma \subset \mathbf{R}^n$ (denoted by Γ-copositive) if and only if

$$v^\top Q v \geq 0 \text{ for all } v \in \Gamma \setminus \{0\}$$

(for *strict copositivity*, $\geq$ has to be replaced by $>$).

Definition 4 Given $\overline{x} \in \Omega$, the *tangent cone* $\Gamma(\overline{x})$ of Ω in $\overline{x}$ is defined as

$$\Gamma(\overline{x}) = \{v \in \mathbf{R}^n : \ \overline{x} + \alpha v \in \Omega \text{ for some } \alpha > 0\}.$$

Definition 5 Given $\overline{x} \in \Omega$ and $v \in \mathbf{R}^n$, we define $\lambda(\overline{x}, v)$ as follows:

$$\lambda(\overline{x}, v) = \max\{\lambda \geq 0 : \ \overline{x} + \lambda v \in \Omega\}.$$

Let us consider the following decomposition for the cone $\Gamma(x)$:

$$\Gamma(x) = \left(\bigcup_{i=1}^{n} \Gamma_i^+(x)\right) \cup \left(\bigcup_{i=1}^{n} \Gamma_i^-(x)\right),$$

where

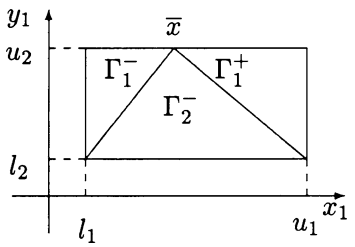
$$\Gamma_i^+(x) = \{v \in \Gamma : \ [x + \lambda(x, v) \cdot v]_i = u_i\},$$
$$\Gamma_i^-(x) = \{v \in \Gamma : \ [x + \lambda(x, v) \cdot v]_i = l_i\},$$
$$i = 1, \ldots, n,$$

i. e. if $v \in \Gamma_i^+(x) \setminus \{0\}$ (or $v \in \Gamma_i^-(x) \setminus \{0\}$), then $v_i \neq 0$ and the maximum stepsize along v moving from x saturates the ith upper (lower) constraint (see Fig. 1).

Proposition 6 *A KKT point $\overline{x}$ yields a global minimum if and only if $\overline{x}$ is stationary point and the Q_i^+ (or Q_i^-) are Γ_i^+-copositive (respectively, Γ_i^--copositive), where*

$$Q_i^+ = \left((u_i - \overline{x}_i)Q + 2\nabla f(\overline{x}) e_i^\top\right),$$
$$Q_i^- = \left((\overline{x}_i - l_i)Q - 2\nabla f(\overline{x}) e_i^\top\right).$$

Finally, the following Proposition [21] gives a sufficient condition for a KKT point to be a global minimum, in terms of convexity of some augmented function $L(x)$.



Quadratic Programming with Bound Constraints, Figure 1
Partitioning of the set $\Gamma(\overline{x})$ for the two-dimensional case

Proposition 7 *Let $\overline{x}$ be a KKT point for problem (1). Let l_i and u_i be finite for each $i \in \{1, \ldots, n\}$. Let*

$$D = \operatorname{diag}\left(\frac{|[\nabla f(\overline{x})]_1|}{u_1 - l_1}, \ldots, \frac{|[\nabla f(\overline{x})]_n|}{u_n - l_n}\right).$$

If $L(x) = f(x) + (x - \overline{x})^\top D(x - \overline{x})$ is convex in Ω, then $\overline{x}$ is a global solution of (1). Moreover, if $L(x)$ is strictly convex in Ω, then this solution is unique.

This kind of result can be a useful tool for branch and bound algorithms for global optimization. Moreover, Proposition 7 allows one to construct test problems in *quadratic programming* with known global minimum.

More results on the *global optimization* criteria for (1) exist in the literature (see, for example, [21] and references therein).

Algorithms for Local Minimization

Most of the algorithms to locally solve problem (1) can be classified in the so-called *active set strategies*, which reduce the solution of the problem to a sequence of auxiliary unconstrained subproblems on affine subspaces of $\mathbf{R}^n$ (faces). They generate a sequence of feasible points $x^{(k)}$, each $x^{(k)}$ associated with a working set $W^{(k)} \subseteq A(x^{(k)})$. The *active set algorithms* can be described according to the very general framework in Table 1.

These methods differentiate on the way they solve the subproblems $P(x^{(k)}, W^{(k)})$ and on the definition of a new face. One of the first of such algorithms, due to B.T. Polyak [29], uses a conjugate gradient algorithm to solve $P(x^{(k)}, W^{(k)})$. Since then, many modifications have been proposed for the solution of the auxiliary problem. In particular, the approximate solution of such problems is suitable to deal with large scale problems. With

Quadratic Programming with Bound Constraints, Table 1
Active set algorithm for QPwBC

Initialization:
take a first point $x^{(0)} \in \Omega$;
$W^{(0)} = A(x^{(0)})$; $k = 0$;
REPEAT
  Solve the quadratic unconstrained problem: $P(x^{(k)}, W^{(k)}) = \min f(x^{(k)} + v)$,
  $v_i = 0, \forall i \in W^{(k)}$;
  IF $P(x^{(k)}, W^{(k)})$ is unbounded below THEN
    ACT1) choose $x^{(k+1)} \in \Omega$:
    $A(x^{(k+1)}) = A(x^{(k)})$ and $f(x^{(k+1)}) < f(x^{(k)})$;
    choose $W^{(k+1)} \supset W^{(k)}$;
  ELSE
    ACT2) $\alpha^{(k)} = \max\{\alpha \in [0, 1] : x^{(k)} + \alpha v^{(k)} \in \Omega\}$;
    $x^{(k+1)} = x^{(k)} + \alpha^{(k)} v^{(k)}$;
    choose $W^{(k+1)}$
    such that $A(x^{(k+1)}) \supseteq W^{(k+1)} \neq W^{(k)}$;
  ENDIF
  $k = k + 1$;
UNTIL(stop condition holds)

regard to the definition of a new working set, in ACT2) a projected gradient step can be taken in order to add more than a new variable to the new working set [18]. Arguments of combinatorial nature show that, in non-degeneracy assumptions, an active set strategy terminates in a finite number of steps at a stationary point, provided that the exact minimization of the subproblems is performed (at least once every j steps, for some prefixed j). In case of degeneracy, the finite termination still holds for some active set algorithms. Specialized versions of active set strategies have been successfully proposed for solving large sparse problems [4,5,19].

On a completely different approach are based the algorithms that belong to the family of the interior point methods (cf. also ► Linear programming: Interior point methods); after Karmarkar's polynomial algorithm for linear programming, many interior point algorithms have been developed for the convex linear complementary problem (and therefore for the convex QPwBC). They include the primal-dual potential reduction algorithm and the path following algorithms [34]. For more detail see ► Linear complementarity problem. Finally, *penalty techniques* have been successfully proposed for the convex QPwBC [6].

Algorithms for Global Minimization

The global optimality conditions expressed in Proposition 6, suggest a very simple algorithmic framework for solving (1), whose main ingredient is the procedure COPOS(Q, Γ, d). Such a procedure [2], given an $n \times n$ matrix Q and a polyhedral cone Γ, detects either the Γ-copositivity of Q or a direction $d \in \Gamma$ such that $d^\top Qd < 0$. In the sequel all Q_i matrices and the cones Γ_i are relative to the stationary point $\overline{x}$.

In the algorithm in Table 2, COPOS is used to escape from local solution which are not global.

In [3] the basic algorithm escape has been improved using pseudoconvexity and a preprocessing procedure. However, because of complexity reasons (the problem of exactly checking copositivity is itself *NP* complete!) algorithms based on copositivity are suitable only for very small size problems.

A different approach [23], originally proposed for concave quadratic problems [30], uses a *separable formulation* based on the eigenstructure of the quadratic form. Using the linear variable transformation $x = Py$, where P is an orthogonal matrix whose columns are the eigenvectors of Q, the original problem is transformed into the separable form

$$\min_{y \in M} \phi_1(y) + \phi_2(y) ,$$

Quadratic Programming with Bound Constraints, Table 2
Global QPwBC algorithm

Initialization:
take a first stationary point $\overline{x}$;
$i = 1$;
REPEAT
  IF $\Gamma_i^+ \neq \{0\}$ THEN call COPOS(Q_i^+, Γ_i^+, d);
  IF $\Gamma_i^- \neq \{0\}$ THEN call COPOS(Q_i^-, Γ_i^-, d);
  IF a direction d is found such that $d^\top Q^+ d < 0$ or $d^\top Q^- d < 0$;
  THEN
    $x^* = \overline{x} + \lambda_{max}(d)d$;
    use x^* as starting point for a procedure that generates a new stationary point $\overline{x}$;
    $i = 0$;
  ENDIF
  $i = i + 1$;
UNTIL($i = n + 1$).

where M is a rectangle of minimum volume that contains $\widehat{\Omega} = \{y \in \mathbf{R}^n : l \leq Py \leq u\}$. The functions $\Phi_1(y)$ and $\Phi_2(y)$ are, respectively, the concave part and the convex part of the objective function.

The function $\Phi_2(y)$ can be underestimated by using a piecewise linear approximation and this gives a convex problem which approximates the original problem and for which an error bound can be given, depending both on the size of the negative eigenvalues of Q and on the size of the range of allowed displacements along the respective eigenvectors. This technique can be incorporated within a branch and bound framework. A way to improve the approximation is to make a partitioning of the domain along the eigendirections, based on the error estimate, and bounding techniques can be devised. An efficient parallel implementation is described in [28].

The *reformulation-linearization/convexification techniques* [31] are based on a suitable *linearized reformulation* of the problem (1). The goal of RLT is to try to approximate the convex envelope of the objective function over the feasible region in deriving tighter and tighter lower bounding linear programs.

Based on the combinatorial nature of the problem some *branch and bound enumerative techniques* have been proposed [13], that can be very expensive from a computational point of view, and therefore only suitable for small size problems or problems whose sparsity allows only a low number of subproblems to be explored.

More attracting from the computational point of view are algorithms based on *interior point methods*, whose main drawback is unfortunately that no guarantee exist about the convergence to the global solution of problem (1) [11].

See also

- ▶ Complexity Theory: Quadratic Programming
- ▶ D.C. Programming
- ▶ Quadratic Assignment Problem
- ▶ Quadratic Fractional Programming: Dinkelbach Method
- ▶ Quadratic Knapsack
- ▶ Quadratic Programming Over an Ellipsoid
- ▶ Reverse Convex Optimization
- ▶ Standard Quadratic Optimization Problems: Algorithms
- ▶ Standard Quadratic Optimization Problems: Applications
- ▶ Standard Quadratic Optimization Problems: Theory

References

1. Benson HH (1995) Concave minimization: Theory, applications and algorithms. In: Horst R, Pardalos PM (eds) Handbook Global Optim. Kluwer, Dordrecht, pp 43–142
2. Bomze IM, Danninger G (1993) A global optimization algorithm for concave quadratic problems. SIAM J Optim 3:826–842
3. Bomze IM, Danninger G (1994) A finite algorithm for solving general quadratic problems. J Global Optim 4:1–16
4. Coleman TF, Hulbert LA (1989) A direct active set algorithm for large sparse quadratic programs with simple bounds. Math Program 45:373–406
5. Coleman TF, Hulbert LA (1993) A globally and superlinearly convergent algorithm for convex quadratic programs with simple bounds. SIAM J Optim 3:298–321
6. Facchinei F, Lucidi S (1992) A class of penalty functions for optimization problems with bound constraints. Optim 26:239, 259
7. Fletcher R, Jackson MP (1974) Minimization of a quadratic function of many variables subject only to lower and upper bounds. J Inst Math Appl 14:159–174
8. Floudas CA, Visweswaran V (1995) Quadratic optimization. In: Horst R, Pardalos PM (eds) Handbook Global Optim. Kluwer, Dordrecht, pp 217–270
9. Frank M, Wolfe P (1956) An algorithm for quadratic programming. Naval Res Logist Quart 3:95–110
10. Gill PE, Murray W, Saunders MA, Wright MH (1991) Inertia-controlling methods for general quadratic programming. SIAM Rev 33(1):1–36
11. Han CG, Pardalos PM, Ye Y (1992) On the solution of indefinite quadratic problems using an interior point algorithm. Informatica 3(4):474–496
12. Han CG, Pardalos PM, Ye Y (1990) Computational aspects of an interior point algorithm for quadratic programming problems with box constraints. SIAM, pp 92–112
13. Hansen P, Jaumard B, Ruiz M, Xiong J (1993) Global minimization of indefinite quadratic functions subject to box constraints. Naval Res Logist 40:373–392
14. Hiriart-Urruty JB (1995) Conditions for global optimality. In: Horst R, Pardalos PM (eds) Handbook Global Optim. Kluwer, Dordrecht, pp 1–26
15. Johnson DS, Papadimitriou CH, Yannakakis M (1988) How easy is local search? J Comput Syst Sci 37:79–100
16. Kozlov MK, Tarasov SP, Khachian LG (1979) Polynomial solvability of convex quadratic programming. S-Dokl 20:1108–1111
17. Manas M (1968) An algorithm for a nonconvex programming problem. Econ Math Obzor Acad Nacl Ceskoslov 4:202–212

18. Moré JJ, Toraldo G (1989) Algorithms for bound constrained quadratic programming problems. Numer Math 55:377–400
19. Moré JJ, Toraldo G (1991) On the solution of large quadratic programming problems with bound constraints. SIAM J Optim 1:93–113
20. Murty KG, Kabadi SN (1987) Some NP-complete problems in quadratic and nonlinear programming. Math Program 39:117–129
21. Neumaier A (1996) Second-order sufficient optimality conditions for local and global nonlinear programming. J Global Optim 9:141–151
22. Pardalos PM (1990) Polynomial time algorithms for some classes of constrained non-convex quadratic problems. Optim 21:843–853
23. Pardalos PM (1991) Global optimisation algorithms for linearly constrained indefinite quadratic problems. Comput Math Appl 21:87–97
24. Pardalos PM, Glick JH, Rosen JB (1987) Global minimization of indefinite quadratic Problems. Computing 39:281–291
25. Pardalos PM, Rosen JB (1986) Methods for global concave minimization: A bibliographic survey. SIAM Rev 28:367–379
26. Pardalos PM, Rosen JB (1987) Constrained global optimization: Algorithms and applications. Lecture Notes Computer Sci, vol 268. Springer, Berlin
27. Pardalos PM, Schnitger G (1988) Checking local optimality in constrained quadratic programming is NP-hard. Oper Res Lett 7(1):33–35
28. Phillips AT, Rosen JB (1990) A parallel algorithm for partially separable non-convex global minimization: Linear constraints. Ann Oper Res 25:101–118
29. Poljak BT (1969) The conjugate gradient method in extremal problems. USSR Comput Math Math Phys 9:94–112
30. Rosen JB (1983) Global minimization of a linearly constrained concave function by partition of feasible domain. Math Oper Res 8:215–230
31. Sherali HD, Tuncbilek CH (1995) A reformulation-convexification approach for solving nonconvex quadratic programming problems. J Global Optim 7:1–31
32. Vavasis SA (1991) Nonlinear optimization: Complexity issues. Oxford Univ. Press, Oxford
33. Vavasis SA (1993) Polynomial time weak approximation algorithms for quadratic programming. In: Pardalos PM (ed) Complexity in Numerical Optimization. World Sci., Singapore
34. Wright S (1993) A path following infeasible-interior point algorithm for linear complementarity problems. Optim Methods Softw 2:79–106
35. Yang EK, Tolle JW (1991) A class of methods for solving large convex programs subject to box constraints. Math Program 51:223–228
36. Ye Y, Tse E (1989) An extension of Karmarkar's projective algorithm for convex quadratic programming. Math Program 44:157–179

Quadratic Programming over an Ellipsoid

Yinyu Ye
Department Management Sci., University Iowa, Iowa City, USA

MSC2000: 90C20, 90C25

Article Outline

Keywords

Quadratic programming; Bisection method; Newton method; Trust region model; Complexity

Quadratic programming (QP) plays an important role in optimization theory. In one sense it is a continuous optimization and a fundamental subroutine for general nonlinear programming, but it is also considered one of the most challenging combinatorial optimization problems.

One of QP problems is to minimize a *quadratic function* over an *ellipsoid constraint*. Since any ellipsoid can be transformed to a *ball* by an affine transformation, without of loss generality, we consider the following ball-constrained QP problem BQP (r):

$$\begin{cases} \min & \frac{1}{2}x^\top Qx + c^\top x \\ \text{s.t.} & x \in \mathcal{B}(r) = \{x \in \mathbf{R}^n : \ \|x\| \le r\}, \end{cases} \tag{1}$$

where $Q \in \mathbf{R}^{n\times n}$, $c \in \mathbf{R}^n$, and superscript $^\top$ denotes the transpose operation. Here, $\|\cdot\|$ denotes L_2 norm and $r > 0$ is the radius of the ball. A main recent result is that this problem is an 'easy' problem, even when the objective function is nonconvex.

We begin with a brief history of this problem. There is a class of nonlinear programming algorithms called model *trust region methods*. In these algorithms, a quadratic function is used as an approximate model of the true objective function around the current iterate. Then the main step is to minimize the model function. In general, however, the model is expected to be accurate or trusted only in a neighborhood of the current

iterate. Accordingly, the quadratic model is minimized in a L_2-norm neighborhood, which is a ball, around the current iterate. Recently (1996), it was demonstrated [5] that a class of combinatorial optimization problems can be solved by solving a sequence of ball-constrained QP problem.

The model-trust region method is due to K. Levenberg [7] and D.W. Marquardt [8]. These authors considered only the case where Q is positive definite. J.J. Moré [10] proposed an algorithm with a convergence proof for this case. D.M. Gay [4] and D.C. Sorenson [15] proposed algorithms for the general case, see also [2]. These algorithms work very well in practice, but no theoretical complexity result was established for this problem then.

It is well known [4,15] that the solution x of problem BQP (r) satisfies the following necessary and sufficient conditions:

$$\begin{aligned} (Q + \mu I)x &= -c\,, \\ \mu &\geq \max\{0, -\underline{\lambda}\}\,, \\ \|x\| &= r\,, \end{aligned} \tag{2}$$

where $\underline{\lambda}$ denotes the least eigenvalue of matrix Q. Since Q is not positive semidefinite, we must have $\underline{\lambda} < 0$.

Let μ^* and x^* satisfy conditions (2). It has been shown that μ^* is unique and

$$\mu^* \leq |\underline{\lambda}| + \frac{\|c\|}{r}\,. \tag{3}$$

It is also known that

$$|\underline{\lambda}| \leq n \max\{|q_{ij}|\}\,,$$

where q_{ij} is the (i, j)th component of matrix Q. Thus, we have

$$0 \leq \mu^* \leq \mu^0 := n \max\{|q_{ij}|\} + \frac{\|c\|}{r}\,, \tag{4}$$

where μ^0 is a computable upper bound. It is further proved that ([19])

$$\frac{1}{2}r^2\,|\underline{\lambda}| \leq \frac{1}{2}r^2\mu^* \leq q(0) - q(x^*) \leq \frac{1}{2}r^2\,|\underline{\lambda}| + r\,\|c\|\,. \tag{5}$$

This inequality can be used to develop an *approximation algorithm* for general quadratic optimization, see [3].

We now analyze the complexity of solving BQP (r). A simple *bisection method* was proposed in [18] and in [19]. For any given μ, denote solutions of the top linear equations by x_μ in conditions (2), i. e.,

$$x_\mu := -(Q + \mu I)^{-1}c\,, \quad \forall \mu > |\underline{\lambda}|\,. \tag{6}$$

For any given μ we can check to see if $\mu \geq |\lambda|$ by checking the positive definiteness of matrix $Q + \mu\, I$, which can be solved as a $LDL^\intercal$ decomposition. These facts lead to a bisection method to search for the root of $\| x_\mu \| = r$ over the interval $\mu \in [|\underline{\lambda}|, \mu^0] \subset [0, \mu^0]$. Obviously, for a given $\epsilon'' \in (0, 1)$, a μ such that, say $0 \leq \mu - \mu^* \leq \epsilon'\, \mu^*/8$, can be obtained in $O(\log(\mu^0/\mu^*) + \log(1/\epsilon'))$ bisection steps, and the cost of each step is $O(n^3)$ arithmetic operations (for performing $LDL^\intercal$ decomposition).

The remaining question is what ϵ' would be sufficient to generate an ϵ-minimizer of $q(x)$ over the ball $\mathcal{B}(r)$, that is, an x satisfying

$$\frac{q(x) - q(x^*)}{q(0) - q(x^*)} \leq \epsilon\,.$$

Let μ denote the right endpoint of the interval generated by the bisection search. Then, $\mu \geq \mu^*$. If $\mu = \mu^*$, then we get an exact solution $x^* = x_{\mu^*}$. Thus, we assume $\mu > \mu^* \geq \underline{\lambda}$. By the positive semidefiniteness of $Q + \mu^*\, I$, we have

$$\|x_\mu\| < \|x^*\| = r\,.$$

We consider two cases.

Case I. In the first case we assume

$$\left(1 - \frac{\epsilon}{8\sqrt{n}}\right)\mu^* \geq |\underline{\lambda}|$$

or

$$\mu^* \geq |\underline{\lambda}| + \frac{\epsilon}{8\sqrt{n}}\mu^*\,.$$

Using the relation (6) and simplifying, we obtain

$$\begin{aligned} \|x^*\|^2 - \|x_\mu\|^2 &= (x^*)^\top \\ &\quad \times (I - (Q + \mu^* I)(Q + \mu I)^{-2}(Q + \mu^* I))x^* \\ &= (x^*)^\top (2(\mu - \mu^*)(Q + \mu I)^{-1} \\ &\quad - (\mu - \mu^*)^2(Q + \mu I)^{-2})x^*\,. \end{aligned}$$

Next we bound the above expression by using the smallest *eigenvalue* $\underline{\lambda}$ of Q. This gives

$$\begin{aligned}
&\|x^*\|^2 - \|x_\mu\|^2 \le \left(\frac{2(\mu-\mu^*)}{(\mu-|\underline{\lambda}|)} - \frac{(\mu-\mu^*)^2}{((\mu-|\underline{\lambda}|))^2}\right)\|x^*\|^2 \\
&= \left(\frac{2(\mu-\mu^*)}{(\mu-\mu^*)+(\mu^*-|\underline{\lambda}|)}\right. \\
&\quad \left. - \frac{(\mu-\mu^*)^2}{((\mu-\mu^*)+(\mu^*-|\underline{\lambda}|))^2}\right)\|x^*\|^2 \\
&= \frac{(\mu-\mu^*)^2 + 2(\mu-\mu^*)(\mu^*-|\underline{\lambda}|)}{((\mu-\mu^*)+(\mu^*-|\underline{\lambda}|))^2} r^2 \\
&= \left(1 - \frac{(\mu^*-|\underline{\lambda}|)^2}{((\mu-\mu^*)+(\mu^*-|\underline{\lambda}|))^2}\right) r^2 \\
&\le \left(1 - \frac{(\frac{\epsilon\mu^*}{8\sqrt{n}})^2}{((\mu-\mu^*)+\frac{\epsilon\mu^*}{8\sqrt{n}})^2}\right) r^2 ,
\end{aligned}$$

where in the last step we used the assumption

$$\mu^* \ge |\underline{\lambda}| + \frac{\epsilon}{8\sqrt{n}}\mu^* .$$

Therefore, if we have $\mu - \mu^* \le \epsilon'\,\mu^*/8$, then

$$\|x^*\|^2 - \|x_\mu\|^2 \le \frac{\left(\frac{2\sqrt{n}\epsilon'}{\epsilon}\right) + \left(\frac{\sqrt{n}\epsilon'}{\epsilon}\right)^2}{\left(1 + \left(\frac{\sqrt{n}\epsilon'}{\epsilon}\right)\right)^2} r^2 \le \frac{2\sqrt{n}\epsilon'}{\epsilon} r^2 . \tag{7}$$

On the other hand, note that

$$\begin{aligned}
&q(x_\mu) - q(x^*) \\
&\quad = \frac{1}{2}x_\mu^\top Q x_\mu + c^\top x_\mu - \frac{1}{2}(x^*)^\top Q x^* - c^\top x^* \\
&\quad = \frac{1}{2}(Qx_\mu + c)^\top (x_\mu - x^*) \\
&\qquad + \frac{1}{2}(Qx^* + c)^\top (x_\mu - x^*) \\
&\quad = -\frac{1}{2}\mu x_\mu^\top (x_\mu - x^*) - \frac{1}{2}\mu^*(x^*)^\top (x_\mu - x^*) \\
&\quad = -\frac{1}{2}(\mu-\mu^*)x_\mu^\top (x_\mu - x^*) \\
&\qquad - \frac{1}{2}\mu^*(\|x_\mu\|^2 - \|x^*\|^2) .
\end{aligned} \tag{8}$$

Now we use the bound (7), the assumption $\mu - \mu^* \le \epsilon'\mu^*/8$ and the fact $\| x_\mu \| \le \| x^* \| = r$ to obtain:

$$\begin{aligned}
q(x_\mu) - q(x^*) &\le \frac{\mu^* r^2 \epsilon'}{8} + r^2\mu^*\frac{\sqrt{n}\epsilon'}{\epsilon} \\
&= \left(\frac{\epsilon'}{4} + \frac{2\sqrt{n}\epsilon'}{\epsilon}\right)\frac{\mu^* r^2}{2} \\
&\le \left(\frac{\epsilon'}{4} + \frac{2\sqrt{n}\epsilon'}{\epsilon}\right)(q(0) - q(x^*)) ,
\end{aligned}$$

where the last step is due to (5). Thus, if we select

$$\epsilon' \le \frac{\epsilon^2}{2\sqrt{n} + \frac{1}{4}} ,$$

then x_μ is feasible for BQP(r) and

$$q(x_\mu) - q(x^*) \le \epsilon(q(0) - q(x^*)) ,$$

i. e., x_μ is an ϵ-*minimizer* to x^*.

Case II. In this case, we have

$$\left(1 - \frac{\epsilon}{8\sqrt{n}}\right)\mu^* < |\underline{\lambda}|$$

or

$$\mu^* < |\underline{\lambda}| + \mu^*\frac{\epsilon}{8\sqrt{n}} .$$

Again, if we have $\mu - \mu^* < \epsilon'\,\mu^*/8$, then $\mu - |\underline{\lambda}| < \frac{\epsilon'\mu^*}{8} + \frac{\mu^*\epsilon}{8\sqrt{n}}$. However, unlike Case I, we find that $\| x_\mu \|$ is not sufficiently close to r. When we observe this fact, we do the following computation, essentially due to S.A. Vavasis and R. Zippel [18], to enhance x_μ.

Let $\underline{q}$, $\left\|\underline{q}\right\| = 1$, be an *eigenvector* associated with the eigenvalue $\underline{\lambda}$. Then, one of the unit vectors e_j, $j = 1, \ldots, n-m$, must have $\left|e_j^\top \underline{q}\right| \ge \frac{1}{\sqrt{n}}$. (In fact, we can use any unit vector q to replace e_j as long as $q^\top \underline{q} \ge \frac{1}{\sqrt{n}}$. A randomly generated q will do it with high probability.) Now we solve for y from

$$(Q + \mu I)y = e_j$$

and let

$$x = x_\mu + \alpha y ,$$

where α is chosen such that $\|x\| = r$. Note we have

$$(Q + \mu I)x = -c + \alpha e_j ,$$

and in the computation of x_μ and y, matrix $Q + \mu I$ needs to be factorized only once.

It is easy to show that

$$\|y\| \geq \frac{1}{\sqrt{n}(\mu - |\underline{\lambda}|)}$$

and

$$|\alpha| \leq 2r(\mu - |\underline{\lambda}|)\sqrt{n} \leq 2r\left(\frac{\epsilon'\mu^*}{8} + \frac{\epsilon\mu^*}{8\sqrt{n}}\right)\sqrt{n}\,.$$

Then, we have from (8)

$$\begin{aligned}
&q(x) - q(x^*)\\
&= \frac{1}{2}(Qx + c)^\top(x - x^*)\\
&\quad + \frac{1}{2}(Qx^* + c)^\top(x - x^*)\\
&= \frac{1}{2}(Qx + c - \alpha e_j)^\top(x - x^*)\\
&\quad + \frac{1}{2}\alpha e_j^\top(x - x^*) - \frac{1}{2}\mu^*(x^*)^\top(x - x^*)\\
&= -\frac{1}{2}\mu x^\top(x - x^*)\\
&\quad + \frac{1}{2}\alpha e_j^\top(x - x^*) - \frac{1}{2}\mu^*(x^*)^\top(x - x^*)\\
&= -\frac{1}{2}(\mu x + \mu^* x^*)^\top(x - x^*)\\
&\quad + \frac{1}{2}\alpha e_j^\top(x - x^*)\\
&= -\frac{1}{2}(\mu - \mu^*)x^\top(x - x^*) + \frac{1}{2}\alpha e_j^\top(x - x^*)\,,
\end{aligned}$$

where the last step follows from $\|x\| = \| x^* \| = r$. Now we use $\mu - \mu^* < \epsilon'\mu^*/8$ and the preceding upper bound on α to estimate the right-hand side:

$$\begin{aligned}
q(x) - q(x^*) &\leq \frac{r^2\mu^*\epsilon'}{8} + 2\left(\frac{\epsilon'\mu^*}{8} + \frac{\epsilon\mu^*}{8\sqrt{n}}\right)r^2\sqrt{n}\\
&= \left(\frac{\epsilon'}{4} + \frac{\sqrt{n}\epsilon'}{2} + \frac{\epsilon}{2}\right)\frac{\mu^* r^2}{2}\\
&\leq \left(\frac{\epsilon'}{4} + \frac{\sqrt{n}\epsilon'}{2} + \frac{\epsilon}{2}\right)(q(0) - q(x^*))\,,
\end{aligned}$$

where the last step is due to (5). Thus, if we choose

$$\epsilon' \leq \frac{\epsilon}{\sqrt{n} + \frac{1}{2}}\,,$$

then x is feasible for BQP(r) and

$$q(x) - q(x^*) \leq \epsilon(q(0) - q(x^*)) \leq \epsilon(\overline{z} - \underline{z})\,,$$

i. e., x is an *ϵ-minimizer* of $q(x)$ over $\mathcal{B}(r)$.

Hence, the bisection method will terminate with an ϵ-minimizer of BQP(r) in at most

$$O\left(\log\left(\frac{\mu^0}{\mu^*}\right) + \log\left(\frac{1}{\epsilon}\right) + \log n\right)$$

steps, or in a total of $O(n^3(\log(\mu^0/\mu^*) + \log(1/\epsilon) + \log n))$ arithmetic operations.

Theorem *The total running time of the bisection algorithm for generating an ϵ-minimal solution to the ball-constrained QP is bounded by $O(n^3(\log(\mu^0/\mu^*) + \log(1/\epsilon) + \log n))$ arithmetic operations.*

Recently, F. Rendl and H. Wolkowicz [14] showed that BQP(r) can be reformulated as a positive semidefinite problem, which is a convex nonlinear problem. There are polynomial *interior point algorithms* (see [11]) to compute an d_x' such that $q'(d_x') - q(d_x'(\mu^k)) \leq \epsilon'$ in $O(n^3 \log(M^k/\epsilon'))$ arithmetic operations. This will also establish an

$$O\left(\left(\frac{n^6}{\epsilon}\log\frac{1}{\epsilon} + n^4\log n\right)\left(\log\frac{1}{\epsilon} + \log n\right)\right)$$

arithmetic operation bound for the algorithm.

The polynomial complexity in Theorem 1 can be further improved. In particular, see [20] for a mixed bisection and *Newton method* for solving BQP(r) and for an arithmetic operation bound $O(n^3 \log(\log(\mu^0/\mu^*) + \log(1/\epsilon')))$ to yield a μ such that $0 \leq \mu - \mu^* \leq \epsilon'$. The brief idea of the method is to first find an approximate $\underline{\mu}$ to the absolute value of the least eigenvalue $|\underline{\lambda}|$ and an approximate eigenvector q to the true $\underline{q}$, such that $0 \leq \underline{\mu} - \underline{\lambda} \leq \epsilon'$ and $q^\top q^k \geq 1 - \epsilon'$. This approximation can be done in $O(n^3 \log(\log(1/\epsilon')))$ arithmetic operations. Then, we will use q to replace e_j in Case II (i. e., $\| x_{\underline{\mu}} < r$) to enhance $x(\underline{\mu})$ and generate a desired approximation. Otherwise, we know $\mu^* > \underline{\mu}$ and, using the mixed method in [20], we will generate a $\mu \in (\underline{\mu}, \mu^0)$ such that $|\mu - \mu^*| \leq \epsilon'\,\mu^*/8$ in $O(n^3 \log(\log(\mu^0/\mu^*) + \log(1/\epsilon')))$ arithmetic operations.

Finally, let Q and c have integer data. Consider the decision problem: Is there an $x \in \mathbf{R}^n$ satisfying $\|x\| \leq 1$, and $q(x) < 0$? Under the *Turing machine* computational model, this problem can be answered in polynomial time (see [18]).

See also

- Complexity Theory: Quadratic Programming
- Quadratic Assignment Problem
- Quadratic Fractional Programming: Dinkelbach Method
- Quadratic Knapsack
- Quadratic Programming with Bound Constraints
- Standard Quadratic Optimization Problems: Algorithms
- Standard Quadratic Optimization Problems: Applications
- Standard Quadratic Optimization Problems: Theory
- Volume Computation for Polytopes: Strategies and Performances

References

1. Celis MR, Dennis JE, Tapia RA (1984) A trust region strategy for nonlinear equality constrained optimization. In: Boggs PT Byrd R, Schnable R (ed) Numerical Optimization. SIAM, Philadelphia, pp 71–82
2. Dennis JE Jr, Schnable RE (1983) Numerical methods for unconstrained optimization and nonlinear equations. Prentice-Hall, Englewood Cliffs
3. Fu M, Luo Z-Q, Ye Y (1996) Approximation algorithms for quadratic programming. J Combin Optim (to appear)
4. Gay DM (1981) Computing optimal locally constrained steps. SIAM J Sci Statist Comput 2:186–197
5. Gibbons LE, Hearn DW, Pardalos PM (1996) A continuous based heuristic for the maximum clique problem. In: DIMACS. Amer. Math. Soc., Providence, pp 103–124
6. Kamath AP, Karmarkar NK, Ramakrishnan KG, Resende MGC (1992) A continuous approach to inductive inference. Math Program 57:215–238
7. Levenberg K (1963) A method for the solution of certain non-linear problems in least squares. Quart Appl Math 2:164–168
8. Marquardt DW (1963) An algorithm for least-squares estimation of nonlinear parameters. J SIAM 11:431–441
9. Martinez JM (1994) Local minimizers of quadratic functions on Euclidean balls and spheres. SIAM J Optim 4:159–176
10. Moré JJ (1977) The Levenberg-Marquardt algorithm: Implementation and theory. In: Watson GA (ed) Numerical Analysis. Springer, Berlin
11. Nesterov YuE, Nemirovskii AS (1993) Interior point polynomial methods in convex programming: Theory and algorithms. SIAM, Philadelphia
12. Pardalos PM, Rosen JB (1987) Constrained global optimization: Algorithms and applications. Lecture Notes Computer Sci, vol 268. Springer, Berlin
13. Powell MJD, Yuan Y (1991) A trust region algorithm for equality constrained optimization. Math Program 49:189–211
14. Rendl F, Wolkowicz H (1994) A semidefinite framework for trust region subproblems with applications to large scale minimization. CORR Report 94-32, Dept Combinatorics and Optim, Univ Waterloo, Ontario
15. Sorenson DC (1982) Newton's method with a model trust region modification. SIAM J Numer Anal 19:409–426
16. Vavasis SA (1991) Nonlinear optimization: Complexity issues. Oxford Sci. Publ., Oxford
17. Vavasis SA (1993) Polynomial time weak approximation algorithms for quadratic programming. In: Pardalos PM (ed) Complexity in Numerical Optimization. World Sci., Singapore
18. Vavasis SA, Zippel R (1990) Proving polynomial time for sphere-constrained quadratic programming. Techn Report 90–1182, Dept Computer Sci, Cornell Univ, Ithaca, NY
19. Ye Y (1992) On affine scaling algorithms for nonconvex quadratic programming. Math Program 56:285–300
20. Ye Y (1994) Combining binary search and Newton's method to compute real roots for a class of real functions. J Complexity 10:271–280
21. Yuan Y (1990) On a subproblem of trust region algorithms for constrained optimization. Math Program 47:53–63

Quadratic Semi-assignment Problem

QSAP

LEONIDAS PITSOULIS
Princeton University, Princeton, USA

MSC2000: 90C27, 90C11, 90C08

Article Outline

Keywords

Optimization

Consider that we have n 'objects' and m 'locations', $n > m$, and we want to assign all objects to locations with at least one object to each location, so as to minimize the overall distance covered by the flow of materials moving between different objects. Given a flow matrix $F =$

(f_{ij}) and a distance matrix $D = (d_{ij})$, we can formulate the *quadratic semi-assignment problem* as follows:

$$\begin{cases} \min & \displaystyle\sum_{i=1}^{n}\sum_{j=1}^{n}\sum_{k=1}^{m-1}\sum_{l=k+1}^{m} f_{ij}d_{kl}x_{ki}x_{lj} \\ & \quad + \displaystyle\sum_{i=1}^{m}\sum_{j=1}^{n} b_{ij}x_{ij} \\ \text{s.t.} & \displaystyle\sum_{j=1}^{n} x_{ij} = 1, \quad i = 1,\ldots,m, \\ & x_{ij} \in \{0,1\}, \\ & \qquad i = 1,\ldots,m, \quad j = 1,\ldots,n. \end{cases}$$

Comparing the above formulation with that of the quadratic assignment problem (cf. ► Quadratic assignment problem), we can see that the QSAP is a relaxed version of the QAP, where instead of assignment constraints we have semi-assignment constraints. SQAP unifies some interesting combinatorial optimization problems like clustering and m-coloring. In a *clustering problem* we are given n objects and a dissimilarity matrix $F = (f_{ij})$. The goal is to find a partition of these objects into m classes so as to minimize the sum of dissimilarities of objects belonging to the same class. Obviously this problem is a QSAP with coefficient matrices F and D, where D is an $m \times m$ identity matrix. In the *m-coloring problem* we are given a graph with n vertices and want to check whether its vertices can be colored by m different colors such that each two vertices which are joined by an edge receive different colors. This problem can be modeled as a SQAP with F equal to the adjacency matrix of the given graph and D the $m \times m$ identity matrix. The m-coloring has an answer 'yes' if and only if the above SQAP has optimal value equal to 0. Practical applications of the SQAP include distributed computing [5] and scheduling [1].

SQAP was originally introduced by D.E. Greenberg [2]. As pointed out in [3], this problem is *NP*-hard. I.Z. Milis and V.F. Magirou [5] propose a Lagrangian relaxation algorithm for this problem, and show that similarly as for for the QAP, it is very hard to provide optimal solutions even for SQAPs of small size. Lower bounds for the SQAP have been provided in [4], and polynomially solvable special cases have been discussed in [3].

See also

- ► Feedback Set Problems
- ► Generalized Assignment Problem
- ► Graph Coloring
- ► Graph Planarization
- ► Greedy Randomized Adaptive Search Procedures
- ► Quadratic Assignment Problem

References

1. Chretienne P (1989) A polynomial algorithm to optimally schedule tasks on a virtual distributed system under tree-like precedence constraints. Europ J Oper Res 43:225–230
2. Greenberg DE (1969) A quadratic assignment problem without column constraints. Naval Res Logist Quart 16:417–422
3. Malucelli F (1993) Quadratic assignment problems: solution methods and applications. PhD Thesis, Dip. Informatica, Univ. Pisa
4. Malucelli F, Pretolani D (1993) Lower bounds for the quadratic semi-assignment problem. Techn Report 955, Centre Rech Transports, Univ Montréal, Canada
5. Milis IZ, Magirou VF (1995) A Lagrangean relaxation algorithm for sparse quadratic assignment problems. Oper Res Lett 17:69–76
6. Stone HS (1977) Multiprocessor scheduling with the aid of network flow algorithms. IEEE Trans Softw Eng 4:85–93

Quasidifferentiable Optimization

GEORGIOS E. STAVROULAKIS
Carolo Wilhelmina Techn. University, Braunschweig, Germany

MSC2000: 49J52, 26B25, 90C99, 26E25

Article Outline

See also
References

Keywords

Nonsmooth optimization; Quasidifferentiability; Nonsmooth analysis; Nonconvex optimization; Mathematical programming

Smoothness, or the existence of the classical derivative information for a function plays a significant role in the theory and the tools used today for modeling, approximation, optimization and for their applications. Nevertheless, nature seems to be more rich than the assumptions done within current mathematical or physical theories. Nonsmoothness arises in a very large number of applications. The arising phenomena, including complex dynamics, pattern formation and chaos, are appealing for both theoretical investigations and practical applications. Most of them have not yet been studied. Abandoning smoothness assumptions one arrives at the area of *nonsmooth analysis*.

Within nonsmooth approximation the classical notion of the derivative is replaced by some set-valued generalized derivative. This is required for the construction of qualitative and quantitative first order approximations of a function with points of nondifferentiability (kinks) or, respectively of a set with corners. In fact, the linearization (i. e., the linear or affine approximation) of a function at a given point which is based on the familiar Taylor expansion formula is based on the assumption that the derivative of the function (or its gradient) exists at the considered point.

Historically, for convex nondifferentiable functions, a suitable set-valued extension of the derivative has been provided by the *subdifferential* of *convex analysis*, in the sense of J.-J. Moreau and R.T. Rockafellar [8,12]. For the general case of nonconvex, nondifferentiable functions, a direct extension of the convex analysis subdifferential has been provided by the *generalized subdifferential* in the sense of F.H. Clarke and Rockafellar [1,2,13]. This notion has been used in a variety of applications, although it does not possess the above mentioned first order approximation property. One should note that a large number of notions have been proposed for the approximation of nonconvex and nonsmooth functions (or sets) or of the solution of affiliated optimization problems. A complete list would go beyond the limits of this short article. This activity demonstrates the large practical interest of this area.

The *quasidifferential* in the sense of V.F. Demyanov and A.M. Rubinov is an appropriate tool for the construction of first order approximations of functions and sets and, subsequently, for the solution of nonsmooth and nonconvex optimization problems. By treating separately convex and concave contributions of the function the quasidifferential introduces an ordered pair of convex sets. Intuitively speaking, the convex analysis subdifferential is present, for the convex contribution, while the superdifferential takes into account the concave parts (which, in turn, can also be studied by means of convex analysis arguments, since a concave function becomes convex if one changes its sign). The links of the quasidifferential with other notions of nonsmooth analysis have been discussed in ► Quasidifferentiable optimization: Dini derivatives, Clarke derivatives and [4]). More important is that certain calculus rules have been developed for the calculation of the quasidifferential of sums, differences, products, quotients and, more general, of every function that can be constructed by using finite number times the minimum and maximum operators over a finite number of classical, smooth constituent functions (see ► Quasidifferentiable optimization: Calculus of quasidifferentials and [7]). Finally, based on the notion of the quasidifferential, certain new variational formulations can be constructed which generalize the notion of *variational inequalities* of convex analysis. These variational formulations have the form of sets of variational inequalities, are valid for the general nonsmooth and nonconvex case (see also ► Quasidifferentiable optimization: Variational formulations [6,11]) and give a computationally advantageous form to the *hemivariational inequalities* in the sense of P.D. Panagiotopoulos (see, among others, ► Nonconvex energy functions: Hemivariational inequalities; ► Hemivariational inequalities: Applications in mechanics and [6,9,10,11]).

Here, the definition of the quasidifferential for one-dimensional and finite-dimensional functions is given and hints for its extension into functionals are discussed. Finally, some information on the related, and more convenient for the numerical applications notion of the codifferential and on the construction of optimization algorithms is provided.

One-Dimensional Nonsmooth Functions

Let f be a real-valued finite function defined on the real line $\mathbf{R}$. The most powerful and widely used tool to study the properties of f is the notion of derivative. Function f is called *differentiable* at $x \in \mathbf{R}$ if there exists its derivative $f'(x)$ at x, which is defined by

$$f'(x) = \lim_{\alpha \to 0} \frac{1}{\alpha} \left[f(x+\alpha) - f(x) \right] . \tag{1}$$

If this limit exists for every point of some open set $S \in \mathbf{R}$, the function f is called differentiable on S.

Among the variety of applications of the derivative one recalls here the first order approximation (linearization) of f in the neighborhood of a point x:

$$f(x+\Delta) = f(x) + f'(x)\Delta + o_x(\Delta) \tag{2}$$

with

$$\frac{o_x(\Delta)}{\Delta} \to 0 \quad \text{as } \Delta \to 0 . \tag{3}$$

Moreover, x^* is a minimum of the function f if

$$f'(x^*) = 0 . \tag{4}$$

Relation (4) defines a stationary point of f, since it also holds true for a maximum and for a saddle point of f. As is usual, higher order derivatives are checked in order to specify the nature of the stationary point.

One-Sided Differentials

Assume now that the limit (1) does not exist, but at the same time the following directional derivatives exist: the right-hand side derivative $f'_+(x)$ and the left-hand side derivative $f'_-(x)$ of f at x. The right-hand side derivative is defined by:

$$f'_+(x) = \lim_{\alpha \downarrow 0} \frac{1}{\alpha} \left[f(x+\alpha) - f(x) \right] . \tag{5}$$

Analogously, the left-hand side derivative is defined by the limit:

$$f'_-(x) = \lim_{\alpha \uparrow 0} \frac{1}{\alpha} \left[f(x+\alpha) - f(x) \right] . \tag{6}$$

Here $\alpha \downarrow 0$ means that $\alpha \to 0$, by taking positive values $\alpha > 0$ and $\alpha \uparrow 0$ means that $\alpha \to 0$, with negative values $\alpha < 0$.

It is clear that for a function f to be differentiable at x it is necessary and sufficient that $f'_+(x) = f'_-(x)$.

The *directional derivative* of a function f at point x and in the direction $x \in \mathbf{R}$ is defined by the limit:

$$f'(x,g) = \lim_{\alpha \downarrow 0} \frac{1}{\alpha} \left[f(x+\alpha g) - f(x) \right] , \tag{7}$$

if this limit exists.

The notion of directional derivative is a proper extension of the notion of the derivative. For example, it can be used to linearize a given function (cf. (2)) along a direction g. In this case relation (2) holds along a given direction, a different value holds for the opposite direction, etc, so that it provides the basis for a quasilinearization of the function f.

From the definition one may easily see that a necessary condition for a directionally differentiable function f to attain a minimum at point x^* is that:

$$f'(x^*, g) \geq 0 , \quad \forall g \in \mathbf{R} . \tag{8}$$

If strict inequality holds in (8) for every direction g not equal to zero, the condition becomes also sufficient for x^* to be a strict local minimum of f. On the other hand, a necessary condition for a directionally differentiable function f to attain a maximum at point x^* is that:

$$f'(x^{**}, g) \leq 0 , \quad \forall g \in \mathbf{R} , \tag{9}$$

with analogous implications for a strict local maximum.

A point x^* which satisfies relation (8) is called an inf-stationary point of f, while a point x^{**} satisfying (9) is called a sup-stationary point. It is interesting to observe that for a nonsmooth function first order optimality conditions may, in some cases, become sufficient for a minimum or a maximum.

Quasidifferential

A function $f\colon \mathbf{R} \to \mathbf{R}$ is called *quasidifferentiable* (q.d) at a point x if it is directionally differentiable at x and there exists a pair of closed intervals $\underline{\partial} f(x) = [v_1, v_2]$ and $\overline{\partial} f(x) = [w_1, w_2]$ such that

$$f'(x,g) = \max_{v \in \underline{\partial} f(x)} vg + \min_{w \in \overline{\partial} f(x)} wg , \quad \forall g \in \mathbf{R} . \tag{10}$$

The pair of intervals $Df(x) = [\underline{\partial} f(x), \overline{\partial} f(x)]$ is called a *quasidifferential* of f at x. The set $\underline{\partial} f(x)$ is called the

subdifferential and the set $\overline{\partial} f(x)$ the *superdifferential* of f at x.

It is clear that a quasidifferential is not uniquely defined. In fact, if a function f is quasidifferentiable at x and $Df(x)$ is its quasidifferential at this point, i. e., $Df(x) = [\underline{\partial} f(x), \overline{\partial} f(x)]$, then every pair of the form $[\underline{\partial} f(x) + C\overline{\partial} f(x) - C]$, where C is an interval $C = [c_1, c_2] \in \mathbf{R}$, with $c_1 \leq c_2$ is also a quasidifferential of f at x. In fact, the quasidifferential is a class of equivalent ordered pairs of convex sets.

Necessary and Sufficient Optimality Conditions

For a quasidifferentiable function the necessary and sufficient optimality conditions (see (8)–(9)) can be written as follows. Let $Df(x) = [\underline{\partial} f(x), \overline{\partial} f(x)]$ be a quasidifferential of f at x. A necessary condition for function f to attain a minimum at point x^* is that:

$$-\overline{\partial} f(x^*) \subset \underline{\partial} f(x^*) \,. \tag{11}$$

The condition

$$-\overline{\partial} f(x^*) \subset \operatorname{int} \underline{\partial} f(x^*) \tag{12}$$

is sufficient for x^* to be a strict local minimum of f. Analogously, a necessary condition for a maximum of f at x^{**} is that:

$$-\underline{\partial} f(x^{**}) \subset \overline{\partial} f(x^{**}) \,, \tag{13}$$

with an analogous result for a sufficient condition for a strict local maximum:

$$-\underline{\partial} f(x^{**}) \subset \operatorname{int} \overline{\partial} f(x^{**}) \,. \tag{14}$$

Finite-Dimensional Nonsmooth Functions

Subdifferentiable Functions

Let a function f defined on an open set $X \subset \mathbf{R}^n$ be directionally differentiable at a point $x \in X$. The function f is *subdifferentiable* at x if its directional derivative is a superlinear function, i. e. there exists a convex compact set U such that

$$f'(x, g) = \max_{h \in U} (h, g), \forall g \in \mathbf{R}^n \,. \tag{15}$$

Superdifferentiable Functions

A function is *superdifferentiable* at x if its directional derivative can be written by means of a convex compact set V as

$$f'(x, g) = f'_x(g) = \min_{h \in V} (h, g), \quad \forall g \in \mathbf{R}^n \,. \tag{16}$$

Quasidifferentiable Functions

A directionally differentiable function f defined on an open set $X \subset \mathbf{R}^n$ is called *quasidifferentiable* at a point $x \in X$, if there exists an ordered pair of convex compact sets $[U, V]$ in $R^n \times \mathbf{R}^n$ which produces the directional derivative of the function by:

$$f'(x, g) = f'_x(g) = \max_{h \in U} (h, g) + \min_{h \in V} (h, g), \quad \forall g \in \mathbf{R}^n \,. \tag{17}$$

Clearly, the first term on the right of (17) is a sublinear function while the second term is a superlinear function. Thus, the directional derivative of a quasidifferentiable function belongs to the space L of functions which can be written as the sum of a sublinear function and a superlinear function. Moreover with an element $[U, V]$ of the space of compact sets it is associated the class of equivalent ordered pairs of compact convex sets.

Thus, the class of equivalent ordered pairs of convex compact sets $[U, V]$ of (17) (the quasidifferential $Df(x)$ of f at x) fully describes the first order derivative of the directionally differentiable function f and gives rise to the quasilinearization (17) and, subsequently, to a qualitative and quantitative first order approximation of f in the sense of (2).

As an example, let us mention that for a differentiable function f either $Df = [\nabla f, \{0\}]$ or $Df = [\{0\}, \nabla f]$ can be used as the quasidifferential of f. For a convex, nondifferentiable function f, $Df = [\partial f, \{0\}]$, where ∂f denotes the classical subdifferential of convex analysis [12] can be used. Analogously, for a concave function f, one may uses $Df = [\{0\}, \partial f]$, where ∂f denotes the superdifferential of the concave function f. A *difference convex function* (d.c. function) is a function f which can be expressed as the difference of two appropriately determined convex constituents, i. e., $f(x) = f_1(x) - f_2(x)$, $\forall x \in X$, where $f_1(x)$ and $f_2(x)$ are convex functions. In this case one constructs a quasidifferential simply by

$Df(x) = [\partial f_1(x), \partial f_2(x)]$, where the convex subdifferentials of the functions $f_1(x)$ and $f_2(x)$ are used.

Further Related Topics

Extension of the theory of quasidifferentiability to infinite-dimensional function spaces has not been studied till now (1999) in details. First hints can be found in [3,6].

The notion of the quasidifferential has been extended by Demyanov to the notion of the *codifferential*, which has certain advantages for numerical applications (see ► Quasidifferentiable optimization: Codifferentiable functions and [4]). Several applications of the quasidifferentiability concept and related references are given in ► Quasidifferentiable optimization: Applications and in [5].

See also

- ► Generalized Monotonicity: Applications to Variational Inequalities and Equilibrium Problems
- ► Hemivariational Inequalities: Applications in Mechanics
- ► Hemivariational Inequalities: Eigenvalue Problems
- ► Hemivariational Inequalities: Static Problems
- ► Nonconvex Energy Functions: Hemivariational Inequalities
- ► Nonconvex-Nonsmooth Calculus of Variations
- ► Quasidifferentiable Optimization: Algorithms for Hypodifferentiable Functions
- ► Quasidifferentiable Optimization: Algorithms for QD Functions
- ► Quasidifferentiable Optimization: Applications
- ► Quasidifferentiable Optimization: Applications to Thermoelasticity
- ► Quasidifferentiable Optimization: Calculus of Quasidifferentials
- ► Quasidifferentiable Optimization: Codifferentiable Functions
- ► Quasidifferentiable Optimization: Dini Derivatives, Clarke Derivatives
- ► Quasidifferentiable Optimization: Exact Penalty Methods
- ► Quasidifferentiable Optimization: Optimality Conditions
- ► Quasidifferentiable Optimization: Stability of Dynamic Systems
- ► Quasidifferentiable Optimization: Variational Formulations
- ► Quasivariational Inequalities
- ► Sensitivity Analysis of Variational Inequality Problems
- ► Solving Hemivariational Inequalities by Nonsmooth Optimization Methods
- ► Variational Inequalities
- ► Variational Inequalities: F. E. Approach
- ► Variational Inequalities: Geometric Interpretation, Existence and Uniqueness
- ► Variational Inequalities: Projected Dynamical System
- ► Variational Principles

References

1. Clarke FH (1983) Optimization and nonsmooth analysis. Wiley/Interscience, New York. reprinted as: Classics Appl. Math., vol 5. SIAM, Philadelphia, PA, 1990
2. Clarke FH, Ledyaev YuS, Stern RJ, Wolenski PR (1998) Nonsmooth analysis and control theory. Springer, Berlin
3. Demyanov VF, Rubinov AM (1985) Quasidifferentiable calculus. Optim. Software, New York
4. Demyanov VF, Rubinov AM (1995) Introduction to constructive nonsmooth analysis. P. Lang, Frankfurt am Main
5. Demyanov VF, Rubinov AM (eds) (2000) Quasidifferentiability and related topics. Kluwer, Dordrecht
6. Demyanov VF, Stavroulakis GE, Polyakova LN, Panagiotopoulos PD (1996) Quasidifferentiability and nonsmooth modelling in mechanics, engineering and economics. Kluwer, Dordrecht
7. Demyanov VF, Vasiliev LN (1985) Nondifferentiable optimization. Optim. Software, New York
8. Moreau JJ (1968) La notion de sur-potentiel et les liaisons unilaté en élastostatique. CR Acad Sci Paris 267A:954–957
9. Panagiotopoulos PD (1988) Nonconvex superpotentials and hemivariational inequalities. Quasidiferentiability in mechanics. In: Moreau JJ, Panagiotopoulos PD (eds) Nonsmooth Mechanics and Applications. CISM Lect. Springer, Berlin, pp 83–176
10. Panagiotopoulos PD (1993) Hemivariational inequalities. Applications in mechanics and engineering. Springer, Berlin
11. Panagiotopoulos PD, Stavroulakis GE (1992) New types of variational principles based on the notion of quasidifferentiability. Acta Mechanics 94:171–194
12. Rockafellar RT (1970) Convex analysis. Math Ser, vol 28. Princeton Univ. Press, Princeton
13. Rockafellar RT (1981) The theory of subgradients and its applications to problems of optimization: Convex and nonconvex functions. Heldermann, Berlin

Quasidifferentiable Optimization: Algorithms for Hypodifferentiable Functions

GEORGIOS E. STAVROULAKIS
Carolo Wilhelmina Techn. University, Braunschweig, Germany

MSC2000: 49J52, 65K99

Article Outline

Keywords

Nonsmooth optimization; Codifferentiability

Quasidifferentiability and codifferentiability extend the notion of the subdifferential of convex analysis for a quite general class of nonconvex and nonsmooth functions. If for a directionally differentiable function f: $\mathbf{R}^n \to \mathbf{R}$ there exists an ordered pair of convex compact sets $[U, V]$ in $\mathbf{R}^n \times \mathbf{R}^n$ which produces the directional derivative of f at x in the direction g by the expression:

$$f'(x,g) = \max_{h\in U}(h,g) + \min_{h\in V}(h,g)\,, \tag{1}$$

this function is called *quasidifferentiable* in the sense of V.F. Demyanov and A.M. Rubinov.

If moreover the quasidifferential of the above function is of the form $[U, 0]$ (where 0 is considered as an element of the space $\mathbf{R}^n$), then function f is called *subdifferentiable*.

More details on this notion, the calculus rules for computing quasidifferentials, its connection to other notions of nonsmooth analysis and it applications can be found in ► Quasidifferentiable optimization; ► Quasidifferentiable optimization: Calculus of quasidifferentials; ► Quasidifferentiable optimization: Dini derivatives, Clarke derivatives; ► Quasidifferentiable optimization: Applications; as well as in [1,2,3].

The quasidifferential, as well as the subdifferential of *convex analysis*, are set-valued quantities which include discontinuities at the points of nondifferentiability. In numerical algorithms this may cause problems. A notion that takes into account neighboring information would be more appropriate. This led Demyanov to extend the notion of the quasidifferential and to define the notion of the codifferential.

Let X be an open subset of $\mathbf{R}^n$ and let a function f be defined and finite for every $x \in X$. A function f is called *codifferentiable* at x if there exist convex, compact sets $\underline{d}f(x) \subset \mathbb{R}^{n+1}$ and $\overline{d}f(x) \subset \mathbb{R}^{n+1}$ such that the function admits a first order approximation in a neighborhood of x of the form

$$f(x+\Delta) = f(x) + \max_{[\alpha,v]\in \underline{d}f(x)} [\alpha + (v,\Delta)] + \min_{[b,w]\in\overline{d}f(x)} [b + (w,\Delta)] + o_x(\Delta)\,, \tag{2}$$

where $o_x(\alpha\Delta)/\alpha \to 0$, as $\alpha \downarrow 0$, $\forall \Delta \in \mathbf{R}^n$. The ordered pair of convex, compact sets $Df(x) = [\underline{d}f(x), \overline{d}f(x)]$ is called a *codifferential* of f at x, where $\underline{d}f(x)$ is a *hypodifferential* and $\overline{d}f(x)$ is a *hyperdifferential*.

If there exists a codifferential of the form $Df(x) = [\overline{d}f(x), 0]$, where 0 is considered as an element of space $\mathbf{R}^{n+1}$, the function f is called *hypodifferentiable*.

One recalls that classical convex nondifferentiable functions are subdifferentiable (resp. hypodifferentiable) in the above outlined framework, since one may use the classical convex analysis subdifferential in the above definitions for the construction of the subdifferential (resp. the hypodifferential) at a given point.

More details about codifferentiability (including extensions to higher order codifferentials) can be found in ► Quasidifferentiable optimization: Codifferentiable functions.

Hypodifferentiable Optimization

Efficient nonsmooth optimization algorithms can be constructed for hypodifferentiable functions. In fact, the technique of replacing a nondifferentiable optimization problem by an enlarged, classical, inequality constrained optimization problem has been successfully used for convex or for composite optimization problems [4,13]. For hypodifferentiable functions a direction of descent at each given point can be determined and used in an iterative optimization procedure.

Let us consider a conceptual iterative steepest descent optimization algorithm and the form it takes for nondifferentiable (hypodifferentiable) functions. First, recall that a nondifferentiable function does not possess derivatives in the classical sense. One uses set-valued approximations of the derivative (cf., the subdifferential or the hypodifferential) at the points of the nondifferentiability instead.

Accordingly optimality conditions (which will also provide the stopping rules for an optimization algorithm) and the calculation of the steepest descent direction must appropriately be modified.

The first order necessary condition for a hypodifferentiable function f to attain a minimum at point x_0 reads:

$$0 \in \underline{d} f(x_0)\,. \tag{3}$$

Points x_0 for which relation (3) is satisfied are called *inf-stationary points*. Note that the previous relation hold in the space $\mathbf{R}^{n+1}$.

If at a given point x_k, at the kth iteration of an iterative optimization scheme, relation (3) is not satisfied, then one may always find the point $\overline{z}$ with minimum norm in the closed convex set $\underline{d} f(x_k)$, such that:

$$\overline{z}^*(x_k) = (\eta^*(x_k), z^*(x_k)) = \arg \min_{\overline{z} \in \underline{d} f(x_k)} \|\overline{z}\|\,. \tag{4}$$

Since (3) is not satisfied, one has $\left\|\overline{z}^*(x_k)\right\| > 0$. The direction

$$g_k(x_k) = -\frac{z^*, (x_k)}{\|z^*(x_k)\|} \tag{5}$$

can be used as a descent direction within an optimization algorithm.

In the conceptual manner used in this note, the next step of the iterative algorithm will have the form:

$$x_{k+1} = x_k + \alpha_k x_k\,,$$

where steplength α_k will be determined from the solution of the one-dimensional optimization problem (along the direction g_k):

$$\alpha_k = \arg \min_{\alpha \geq 0} \{f(x_k + \alpha g_k)\}\,.$$

For more general quasidifferentiable and codifferentiable functions one may construct appropriate solution algorithms, see ▶ Quasidifferentiable optimization: Algorithms for QD functions and in the original literature (see, e. g., [1]).

Comments

Nondifferentiable optimization procedures have attracted the attention of several researchers and practitioners in the last decade. The lost of information which is connected with smoothing approaches is, for several applications, critical for the quality of the results. Beyond the quasidifferentiable optimization literature, previously mentioned in this note, general methods and theory for descent type methods for nonsmooth functions can be found in [7,12]. In this respect, the bundle concept has been found useful (see, among others, [6,8,9,11]). An application of this method for the solution of hemivariational inequality problems arising in mechanics can be found in [10] and [5].

Closing one would like to mention again the additional requirements of nonsmooth optimization with respect to classical, smooth one. First, stopping criteria must take into account the set-valued nature of the nonsmooth optimality conditions. Otherwise cycling in an iterative scheme or premature exit at a noncritical point may occur. This is the more critical point. Moreover, the line search must take into account the nondifferentiability of the involved function. This requirement is, usually, easily taken into account (for instance, by means of a derivative-free technique).

See also

- ▶ Generalized Monotonicity: Applications to Variational Inequalities and Equilibrium Problems
- ▶ Hemivariational Inequalities: Applications in Mechanics
- ▶ Hemivariational Inequalities: Eigenvalue Problems
- ▶ Hemivariational Inequalities: Static Problems
- ▶ Nonconvex Energy Functions: Hemivariational Inequalities
- ▶ Nonconvex-Nonsmooth Calculus of Variations
- ▶ Quasidifferentiable Optimization
- ▶ Quasidifferentiable Optimization: Algorithms for QD Functions
- ▶ Quasidifferentiable Optimization: Applications
- ▶ Quasidifferentiable Optimization: Applications to Thermoelasticity
- ▶ Quasidifferentiable Optimization: Calculus of Quasidifferentials
- ▶ Quasidifferentiable Optimization: Codifferentiable Functions

- ▶ Quasidifferentiable Optimization: Dini Derivatives, Clarke Derivatives
- ▶ Quasidifferentiable Optimization: Exact Penalty Methods
- ▶ Quasidifferentiable Optimization: Optimality Conditions
- ▶ Quasidifferentiable Optimization: Stability of Dynamic Systems
- ▶ Quasidifferentiable Optimization: Variational Formulations
- ▶ Quasivariational Inequalities
- ▶ Sensitivity Analysis of Variational Inequality Problems
- ▶ Solving Hemivariational Inequalities by Nonsmooth Optimization Methods
- ▶ Variational Inequalities
- ▶ Variational Inequalities: F. E. Approach
- ▶ Variational Inequalities: Geometric Interpretation, Existence and Uniqueness
- ▶ Variational Inequalities: Projected Dynamical System
- ▶ Variational Principles

References

1. Demyanov VF, Rubinov AM (1995) Introduction to constructive nonsmooth analysis. P. Lang, Frankfurt am Main
2. Demyanov VF, Stavroulakis GE, Polyakova LN, Panagiotopoulos PD (1996) Quasidifferentiability and nonsmooth modelling in mechanics, engineering and economics. Kluwer, Dordrecht
3. Demyanov VF, Vasiliev LN (1985) Nondifferentiable optimization. Optim. Software, New York
4. Fletcher R (1990) Practical methods of optimization. Wiley, New York
5. Haslinger J, Miettinen M, Panagiotopoulos PD (1999) Finite element method for hemivariational inequalities. Kluwer, Dordrecht
6. Hiriart-Urruty J-B, Lemaréchal C (1993) Convex analysis and minimization algorithms. Springer, Berlin
7. Kiwiel KC (1985) Methods of descent for nondifferentiable optimization. Springer, Berlin
8. Lemarechal C, Mifflin R (eds) (1978) Bundle methods in nonsmooth optimization. Pergamon, Oxford
9. Mäkelä MM, Neittaanmäki P (1992) Nonsmooth optimization. World Sci., Singapore
10. Miettinen M, Mäkelä MM, Haslinger J (1995) On numerical solution of hemivariational inequalities by nonsmooth optimization methods. J Global Optim 8(4):401–425
11. Schramm H, Zowe J (1992) A version of the bundle idea for minimizing a nonsmooth function: conceptual idea, convergence analysis, numerical results. SIAM J Optim 2:121–152
12. Shor NZ (1985) Minimization methods for nondifferentiable functions. Springer, Berlin
13. Womersley RS, Fletcher R (1986) An algorithm for composite nonsmooth optimization problems. J Optim Th Appl 48:493–523

Quasidifferentiable Optimization: Algorithms for QD Functions

VLADIMIR F. DEMYANOV
St. Petersburg State University, St. Petersburg, Russia

MSC2000: 90Cxx, 65Kxx

Article Outline

Keywords

Quasidifferentiable function; Codifferentiable function; Method of codifferential descent; Method of hypodifferential descent; D.c. function

Codifferentiable (c.d.) Functions

$f\colon \mathbf{R}^n \to \mathbf{R}$ is called *quasidifferentiable* at $x \in \mathbf{R}^n$ if it is directionally differentiable (in the sense of Dini or Hadamard) and there exists a pair $\mathcal{D}f(x) = \left[\underline{\partial} f(x), \overline{\partial} f(x)\right]$ of compact convex sets of $\mathbf{R}^n$ such that

$$f'(x,g) = \max_{v \in \underline{\partial} f(x)} (v,g) + \min_{w \in \overline{\partial} f(x)} (w,g)\,. \tag{1}$$

Here $f'(x,g)$ is either the Dini or Hadamard derivative of f at x in a direction $g \in \mathbf{R}^n$. (See ▶ Quasidifferentiable optimization: Optimality conditions).

In the sequel we discuss only the problem of minimizing the function f. In ► Quasidifferentiable optimization: Optimality conditions, necessary conditions for a minimum of f were formulated in terms of quasidifferentials (q.d.) and a formula for computing steepest descent directions was derived. However, it is difficult to apply steepest descent directions for constructing numerical methods for minimizing the function f since the quasidifferential mapping $\mathcal{D}f$ is, in general, discontinuous in the Hausdorff metric. This is why we need some other tool to overcome the discontinuity of $\mathcal{D}f$.

A function $f\colon \mathbf{R}^n \to \mathbf{R}$ is called *Dini codifferentiable* (D.c.d.) at $x \in \mathbf{R}^n$ if there exists a pair $Df(x) = [\underline{d}f(x), \overline{d}f(x)]$ of compact convex sets of $\mathbf{R}^{n+1}$ such that

$$f(x+\Delta) = f(x) + \max_{[a,v]\in \underline{d}f(x)} \big[a + (v,\Delta)\big] + \min_{[b,w]\in \overline{d}f(x)} \big[b + (w,\Delta)\big] + o_x(\Delta)\,. \tag{2}$$

$$\frac{o_x(\alpha\Delta)}{\alpha} \to 0\,, \quad \forall \Delta \in \mathbf{R}^n\,. \tag{3}$$

Here $a, b \in \mathbf{R}$; $v, w \in \mathbf{R}^n$. If in (2)

$$\frac{o_x(\Delta)}{\|\Delta\|} \overset{\|\Delta\|\to 0}{\to} 0\,, \quad \forall \Delta \in \mathbf{R}^n\,, \tag{4}$$

then f is called *Hadamard codifferentiable* (H.c.d) at x.

Without loss of generality it may be assumed that

$$\max_{[a,v]\in \underline{d}f(x)} a = \min_{[b,w]\in \overline{d}f(x)} b = 0\,. \tag{5}$$

If it causes no misunderstanding, we shall use the term codifferentiable (c.d.) for both Dini and Hadamard codifferentiable functions.

The pair $Df(x) = [\underline{d}f(x), \overline{d}f(x)]$ is called a *codifferential* of f at x, $\underline{d}f(x)$ is a *hypodifferential*, $\overline{d}f(x)$ is a *hyperdifferential*. A codifferential (like quasidifferential) is not uniquely defined. If there exists a codifferential of the form $Df(x) = [\underline{d}f(x), \{0_{n+1}\}]$, the function f is called *hypodifferentiable* at x. If there exists a codifferential of the form $Df(x) = [\{0_{n+1}\}, \overline{d}f(x)]$, the function f is called *hyperdifferentiable* at x.

It is easy to see that the class of Dini (Hadamard) codifferentiable functions coincides with the class of Dini (Hadamard) quasidifferentiable functions.

For example, if $Df(x) = [\underline{d}f(x), \overline{d}f(x)]$ is a codifferential of f at x such that (5) holds, then the pair $\mathcal{D}f(x) = [\underline{\partial}f(x), \overline{\partial}f(x)]$, where

$$\underline{\partial}f(x) = \{v \in \mathbf{R}^n\colon\ [0,v] \in \underline{d}f(x)\}\,,$$
$$\overline{\partial}f(x) = \Big\{w \in \mathbf{R}^n\colon\ [0,w] \in \overline{d}f(x)\Big\}\,,$$

is a quasidifferential of f at x.

A function f is called *continuously codifferentiable* at x if it is codifferentiable in some neighborhood of x and there exists a codifferential mapping Df which is Hausdorff continuous at x.

Remark 1 Of course, it is possible to introduce the notion of continuously quasidifferentiable function; however, if f is continuously q.d. at x then it is just differentiable at x.

For a fixed Δ the functions (see (1) and (2))

$$\Phi_{1x}(\Delta) = f(x) + \max_{v\in\underline{\partial}f(x)} (v,\Delta) + \min_{w\in\overline{\partial}f(x)} (w,\Delta)$$

and

$$\Phi_{2x}(\Delta) = f(x) + \max_{[a,v]\in \underline{d}f(x)} \big[a + (v,\Delta)\big] + \min_{[b,w]\in \overline{d}f(x)} \big[b + (w,\Delta)\big]$$

are both first order approximations of f in a neighborhood of x. The function $F_1(\Delta) = \Phi_{1x}(\Delta) - f(x)$ is positively homogeneous (of degree one) in Δ while the function $F_2(\Delta) = \Phi_{2x}(\Delta) - f(x)$ is, in general, not positively homogeneous. The loss of homogeneity is the price to be paid for the continuity (if any) of the codifferential mapping.

Note again that the 'value' of the mapping $\mathcal{D}f$ at any x is a pair of convex compact sets in $\mathbf{R}^{n+1}$.

If turns out that most of the known functions are continuously codifferentiable (see [3,4]). For example, all smooth, convex, concave and concavo-convex functions are continuously codifferentiable. The class of c.d. functions enjoys a very rich calculus similar to that for q.d. functions (see ► Quasidifferentiable optimization: Optimality conditions) which is a generalization of the classical differential calculus. The class of c.d. functions was introduced in [3].

First we discuss the problem of minimizing a c.d. function on the entire space (in the absence of constraints).

For a c.d. function the following necessary condition holds:

Proposition 2 *For a point $x^* \in \boldsymbol{R}^n$ to be a minimizer of f on $\boldsymbol{R}^n$ it is necessary that*

$$\begin{aligned} &0_{n+1} \in \{\underline{d}f(x^*) + [0, w]\}\,, \\ &\forall [0, w] \in \overline{d}f(x^*) \end{aligned} \tag{6}$$

(it is assumed that condition (5) holds).

A point x^* satisfying (6) is called *inf-stationary*. Let x be not inf-stationary. Then there exists $\overline{w} = [0, w] \in \overline{d}f(x)$ such that

$$0_{n+1} \notin \{\underline{d}f(x^*) + \overline{w}\} = L_{\overline{w}}(x)\,. \tag{7}$$

Find

$$\min_{\overline{z} \in L_{\overline{w}}(x)} \|\overline{z}\| = \|\overline{z}_{\overline{w}}(x)\|\,.$$

(7) implies that

$$\begin{aligned} &\overline{z}_{\overline{w}}(x) = \left[\eta_{\overline{w}}(x), z_{\overline{w}}(x)\right] \neq 0_{n+1} \\ &(\eta_{\overline{w}}(x) \in \mathbf{R}\,, \quad z_{\overline{w}}(x) \in \mathbf{R}^n)\,. \end{aligned}$$

It is also possible to show that $z_{\overline{w}}(x) \neq 0_n$ and that for the direction

$$g_{\overline{w}}(x) = -\frac{z_{\overline{w}}(x)}{\|z_{\overline{w}}(x)\|}$$

the inequality $f'(x, g_{\overline{w}}(x)) \leq -\|z_{\overline{w}}(x)\|$ holds.

Method of Codifferential Descent (MCD)

Let a function f be defined, Lipschitz and continuously codifferentiable on $\mathbf{R}^n$. Fix any $\mu > 0$. Choose an arbitrary $x_0 \in \mathbf{R}^n$. Let x_k have already been found. If condition (6) holds at x_k, then x_k is inf-stationary and the process terminates. Otherwise, for every $\overline{w} \in \overline{d}_\mu f(x_k)$ where

$$\overline{d}_\mu(x) = \left\{\overline{w} \in \overline{d}f(x):\ \overline{w} = (\omega, w),\ 0 \leq \omega \leq \mu\right\} \tag{8}$$

we find

$$\min_{\overline{z} \in L_{\overline{w}}(x_k)} \|\overline{z}\| = \|\overline{z}_{k\overline{w}}\|$$

with

$$\overline{z}_{k\overline{w}} = [\eta_{k\overline{w}}, z_{k\overline{w}}]\,, \quad L_{\overline{w}}(x_k) = [\underline{d}f(x_k) + \overline{w}]\,.$$

Now, for every $\overline{w} \in \overline{d}_\mu f(x_k)$ we find

$$\min_{\alpha \geq 0} f(x_k - \alpha z_{k\overline{w}}) = f(x_k - \alpha_{k\overline{w}} z_{k\overline{w}}) \tag{9}$$

and then

$$\min_{\overline{w} \in \overline{d}_\mu f(x_k)} f(x_k - \alpha_{k\overline{w}} z_{k\overline{w}}) = f(x_k - \alpha_{k\overline{w}_k} z_{k\overline{w}_k})\,.$$

Put $x_{k+1} = x_k - \alpha_{k\overline{w}_k} z_{k\overline{w}_k}$. Continuing in the same manner we construct a sequence $\{x_k\}$ such that $f(x_{k+1}) < f(x_k)$.

Proposition 3 *(See [4, Thm. V.5.1].) Let the set $\{x \in \boldsymbol{R}^n: f(x) \leq f(x_0)\}$ be bounded, x^* be a limit point of the sequence $\{x_k\}$ and let relation (4) hold uniformly in x from some neighborhood of x^* and in Δ from $S = \{\Delta \in \boldsymbol{R}^n: \|\Delta\| = 1\}$.*

Then the point x^* is an inf-stationary point of f (i. e. condition (6) holds).

Remark 4 The above described MCD is a conceptual method (according to the terminology of E. Polak). It should be adjusted to a specific class of functions. The MCD is a generalization of the classical steepest descent method.

For example, if for every $x \in \mathbf{R}^n$ the set $\overline{d}f(x)$ is the convex hull of a finite number of points then in (8) one can take only points $\overline{w} = (w, \omega)$ which are 'vertices' of $\overline{d}f$ such that $0 \leq \omega \leq \mu$.

In this case at each step it is required to solve only a finite number of one-dimensional optimization problems (9).

Method of Hypodifferential Descent (MHD)

Let f be defined, Lipschitz and continuously hypodifferentiable on $\mathbf{R}^n$, i. e. there exists a codifferential mapping of the form $Df(x) = [\underline{d}f(x), \{0_{n+1}\}]$ which is Hausdorff continuous. Then the necessary condition for a minimum (6) takes the form

$$0_{n+1} \in \underline{d}f(x^*)\,. \tag{10}$$

If $x \in \mathbf{R}^n$ is not an inf-stationary point (i. e., (10) does not hold) then let us find

$$\min_{\overline{z} \in \underline{d}f(x)} \|\overline{z}\| = \|(\eta(x), z(x))\| = \rho(x) = \|\overline{z}(x)\|\,.$$

Since $\rho(x) > 0$ then $\overline{z}(x) \neq 0_{n+1}$. It is possible to show that $z(x) \neq 0_n$. The direction $g(x) = -z(x)/\|z(x)\|$

is a descent direction (not necessarily the steepest descent direction). The vector function $g(x)$ is continuous at any point which is not inf-stationary.

Take any $x_0 \in \mathbf{R}^n$. Let x_k have already been constructed. Let us find $\rho(x_k) = \|\overline{z}(x_k)\|$. If $\rho(x_k) = 0$ then x_k is inf-stationary and the process terminates. Otherwise, take $g_k = -z(x_k)/\parallel z(x_k) \parallel$ and find

$$\min_{\alpha \geq 0} f(x_k + \alpha g_k) = f(x_k + \alpha_k g_k) .$$

Put $x_{k+1} = x_k + \alpha_k x_k$. Continuing analogously we construct a sequence $\{x_k\}$ such that $f(x_{k+1}) < f(x_k)$.

Proposition 5 *Let x^* be a limit point of the sequence $\{x_k\}$ and the hypotheses of Proposition 3 hold. Then $0_{n+1} \in \mathrm{d}f(x^*)$ i. e. x^* is an inf-stationary point of f.*

Difference of Convex (d.c.) Functions

Let $f(x) = f_1(x) - f_2(x)$ where $f_1, f_2 : \mathbf{R}^n \to \mathbf{R}$ are convex. A d.c. function is quasidifferentiable with the quasidifferential $\mathcal{D}f(x) = [\underline{\partial} f(x), \overline{\partial} f(x)]$ where $\underline{\partial} f(x) = \partial f_1(x)$, $\overline{\partial} f(x) = -\partial f_2(x)$, $\partial f_1(x)$ and $\partial f_2(x)$ are *subdifferentials* (in the sense of convex analysis) of the functions f_1 and f_2 respectively:

$$\partial f_i(x) = \left\{ v \in \mathbf{R}^n : \quad \begin{array}{c} f_i(z) - f_i(x) \geq (v, z - x), \\ \forall z \in \mathbf{R}^n \end{array} \right\} .$$

The sets ∂f_i are convex and compact. The necessary condition for a minimum (6) takes the form

$$\partial f_2(x^*) \subset \partial f_1(x^*) . \tag{11}$$

If f_2 is a polyhedral function (i. e. $f_2(x) = \max_{i \in I}\{a_i + (v_i, x)\}$ where $a_i \in \mathbf{R}$, $v_i \in \mathbf{R}^n$, $I = 1, \ldots, N$) then condition (11) is sufficient for the point x^* to be a local minimizer of f.

Since the mappings ∂f_1 and ∂f_2 are discontinuous then $\mathcal{D}f$ is also discontinuous.

If F is a convex function, $\varepsilon \geq 0$ then the set

$$\partial_\varepsilon F(x) = \left\{ v \in \mathbf{R}^n : \quad \begin{array}{c} F(z) - F(x) \\ \geq (v, z - x) - \varepsilon, \\ \forall z \in \mathbf{R}^n \end{array} \right\}$$

is called the *ε-subdifferential* of F at x.

We shall use the following properties of a convex function (see, e. g., [7]):

1) $\partial_\varepsilon F(x)$ is a closed compact set.

2) The mapping $\partial_\varepsilon F$ is Hausdorff continuous jointly in ε and x on $(0, \infty) \times \mathbf{R}^n$.

3)

$$\max_{v \in \partial_\varepsilon F(x)} (v, g) = \inf_{\alpha \geq 0} \frac{1}{\alpha} \left[F(x + \alpha g) - F(x) + \varepsilon \right] := F'_\varepsilon(x, g) .$$

In [6] the following necessary and sufficient condition for a global minimum is stated:

For a point x^* to be a global minimizer of a d.c. function $f(x) = f_1(x) - f_2(x)$ it is necessary and sufficient that

$$\partial_\varepsilon f_2(x^*) \subset \partial_\varepsilon f_1(x^*) , \quad \forall \varepsilon \geq 0 . \tag{12}$$

Note that if $\varepsilon_1 > \varepsilon_2$ and

$$f'_{\varepsilon_1 \varepsilon_2}(x, g) := f'_{1\varepsilon_1}(x, g) - f'_{2\varepsilon_2}(x, g) \leq 0 \tag{13}$$

then

$$\inf_{\alpha \geq 0} f(x + \alpha g) \leq f(x) + \varepsilon_2 - \varepsilon_1 . \tag{14}$$

Let us construct the following method for finding an inf-stationary point of f (i. e. a point satisfying (11)).

Fix ε_0, $\mu_0 = \varepsilon_0/2$. Take an arbitrary $x_{00} \in \mathbf{R}^n$. Assume that the set

$$C = \{x \in \mathbf{R}^n : \ f(x) \leq f(x_{00})\}$$

is bounded (then it is closed since f is continuous). If

$$B_{00} := \partial_{\mu_0} f_2(x_{00}) \subset A_{00} := \partial_{\varepsilon_0} f_1(x_{00})$$

then we put $x_0 = x_{00}$. If

$$\partial_{\mu_0} f_2(x_{00}) \not\subset \partial_{\varepsilon_0} f_1(x_{00})$$

then let us find

$$\max_{w \in B_{00}} \min_{v \in A_{00}} \|v - w\| = \|v_{00} - w_{00}\| = \rho_{00}$$

and put $g_{00} = (w_{00} - v_{00})/\parallel w_{00} - v_{00} \parallel$. Since $\rho_{00} > 0$ then

$$f'_{\varepsilon_0 \mu_0}(x_{00}, g_{00}) < 0$$

and, by (13) and (14), we conclude that

$$\inf_{\alpha\geq 0} f(x_{00}+\alpha g_{00}) = f(x_{00}+\alpha_{00}g_{00}) \leq f(x_{00}) - \frac{\varepsilon_0}{2}\,. \tag{15}$$

Now take $x_{01} = x_{00} + \alpha_{00}g_{00}$ and check the condition

$$B_{01} := \partial_{\mu_0} f_2(x_{01}) \subset A_{01} := \partial_{\varepsilon_0} f_1(x_{01})\,.$$

Continuing in the same manner, in a finite number of steps we shall find a point x_{0s_0} such that

$$B_{0s_0} := \partial_{\mu_0} f_2(x_{0s_0}) \subset A_{0s_0} := \partial_{\varepsilon_0} f_1(x_{0s_0}) \tag{16}$$

(it is due to (15) and the boundedness of C).

Put $x_0 = x_{0s_0}$. By (16)

$$B_0 := \partial_{\mu_0} f_2(x_0) \subset A_0 := \partial_{\varepsilon_0} f_1(x_0)\,.$$

Let x_k be constructed such that

$$\partial_{\mu_i} f_2(x_k) \subset \partial_{\varepsilon_i} f_1(x_k)\,,\quad \forall i \in 0,\dots,k\,, \tag{17}$$

where $\mu_i = \mu_0/2^i$, $\varepsilon = \varepsilon_0/2^i$.

Put $x_{k+1,0} = x_k$. If

$$\partial_{\mu_{k+1}} f_2(x_{k+1,0}) \subset \partial_{\varepsilon_{k+1}} f_1(x_{k+1,0}) \tag{18}$$

then we take $x_{k+1} = x_{k+1,0}$. If (18) does not hold, we continue as above and in a finite number of steps a point $x_{k+1,s_{k+1}}$ will be found such that

$$\partial_{\mu_{k+1}} f_2(x_{k+1,s_{k+1}}) \subset \partial_{\varepsilon_{k+1}} f_1(x_{k+1,s_{k+1}})$$

and we put $x_{k+1} = x_{k+1,s_{k+1}}$.

As a result we construct a bounded sequence $\{x_k\}$ satisfying (17).

Proposition 6 *Any limit point of the sequence $\{x_k\}$ is an inf-stationary point of f.*

Difference of Max-Type (d.m.) Functions

Let

$$f(x) = f_1(x) - f_2(x)$$

where $f_1, f_2\colon \mathbf{R}^n \to \mathbf{R}$ are max-type functions:

$$f_1(x) = \max_{y\in G_1} \varphi_1(x,y)\,,$$

$$f_2(x) = \max_{y\in G_2} \varphi_2(x,y)$$

where φ_1 and φ_2 are continuous on $\mathbf{R}^n \times G_1$ and $\mathbf{R}^n \times G_2$, respectively, and there exist derivatives ${\varphi_1}_x'$ and ${\varphi_2}_x'$ which are continuous. The function f (called a *d.m. function*) is quasidifferentiable. It is also continuously codifferentiable:

$$f(x+\Delta) = f(x) + \max_{[a,v]\in \underline{d}f(x)} \big[a + (v,\Delta)\big] + \min_{[b,w]\in \overline{d}f(x)} \big[b + (w,\Delta)\big] + o_x(\Delta)\,,$$

where

$$\underline{d}f(x) = \operatorname{co}\left\{ [a,v]\colon \begin{array}{c} a = \varphi_1(x,y) - f_1(x), \\ v = \varphi_{1x}'(x,y), \\ y \in G_1 \end{array} \right\} \subset \mathbf{R}\times\mathbf{R}^n,$$

$$\overline{d}f(x) = \operatorname{co}\left\{ [b,w]\colon \begin{array}{c} b = f_2(x) - \varphi_2(x,y), \\ w = -\varphi_{2x}'(x,y), \\ y \in G_2 \end{array} \right\} \subset \mathbf{R}\times\mathbf{R}^n\,.$$

Here $a, b \in \mathbf{R}$; $v, w \in \mathbf{R}^n$.

Now it is possible to employ the MCD for finding inf-stationary points.

Twice Codifferentiable Functions

A function $f\colon \mathbf{R}^n \to \mathbf{R}$ is called *twice codifferentiable* at $x \in \mathbf{R}^n$ if there exist convex compact sets $d^2 f(x)$ and $\overline{d}^2 f(x) \subset \mathbf{R}\times\mathbf{R}^n\times\mathbf{R}^{n\times n}$ such that

$$f(x+\Delta) = f(x) + \max_{[a,v,A]\in \underline{d}^2 f(x)} \left[a + (v,\Delta) + \frac{1}{2}(A\Delta,\Delta)\right] + \min_{[b,w,B]\in \overline{d}^2 f(x)} \left[b + (w,\Delta) + \frac{1}{2}(B\Delta,\Delta)\right] + o(\Delta^2)$$

where

$$\frac{o((\alpha\Delta)^2)}{\alpha^2} \overset{\alpha\to 0}{\to} 0\,,\quad \forall \Delta \in \mathbf{R}^n\,.$$

Here $\mathbf{R}^{n\times n}$ is the space of real $(n\times n)$-matrices.

The pair of sets $D^2 f(x) = [\underline{d}^2 f(x), \overline{d}^2 f(x)]$ is called a *second order codifferential* of f at x. If f is twice c.d. in some neighborhood of x and the mapping $D^2 f$ is Hausdorff continuous at x, then the function is called *twice continuously codifferentiable* at x.

The class of twice c.d. functions is quite rich and enjoys a well-developed calculus (see [4]).

Let f be twice continuously c.d. on $\mathbf{R}^n$. Then the following second order Newton-type method can be employed to find inf-stationary points of f.

Take any $x_0 \in \mathbf{R}^n$. Let x_k have already been constructed. Put

$$F_k(\Delta) = \max_{[a,v,A]\in \underline{d}^2 f(x_k)} \left[a + (v,\Delta) + \frac{1}{2}(A\Delta,\Delta)\right] + \min_{[b,w,B]\in \overline{d}^2 f(x_k)} \left[b + (w,\Delta) + \frac{1}{2}(B\Delta,\Delta)\right],$$

find

$$\min_{\Delta\in\mathbf{R}^n} F_k(\Delta) = F_k(\Delta_k)$$

and take $x_{k+1} = x_k + \Delta_k$.

The sequence $\{x_k\}$ thus constructed converges (under some additional assumptions) to an inf-stationary point of f (see [1]).

Quasidifferentiable Programming Problems

Let functions f and h_i: $\mathbf{R}^n \to \mathbf{R}$ $(i \in I = 1, \ldots, N)$ be quasidifferentiable on $\mathbf{R}^n$ and let

$$\Omega = \{x \in \mathbf{R}^n : \ h_i(x) \le 0, \ \forall i \in I\} .$$

Assume that $\Omega \neq \emptyset$.

It is required to find

$$(\mathrm{P}) \min_{x\in\Omega} f(x) = f^* .$$

The set Ω is called *quasidifferentiable*, problem (P) is a quasidifferentiable (q.d.) programming problem. Necessary conditions for a minimum of f on Ω are stated in ► Quasidifferentiable optimization: Optimality conditions. If all the functions f and f_i' are, in addition, continuously codifferentiable then it is possible to extend the MCD to problem (P) (see [4]).

Another approach to problem (P) is based on the penalization technique.

We say that problem (P) is *calm* if

$$\limsup_{\varepsilon\downarrow 0} \frac{f^* - f_\varepsilon}{\varepsilon} \le B < \infty \tag{19}$$

where

$$f_\varepsilon = \inf_{x\in\Omega_\varepsilon} f(x) ,$$

$$\Omega_\varepsilon = \{x \in \mathbf{R}^n : \ h_i(x) \le \varepsilon, \ \forall i \in I\} ,$$

$$\varepsilon > 0 .$$

Proposition 7 *If the calmness condition (19) holds then there exists $A^* < \infty$ such that, for any $A > A^*$, the set of minimizers of the function f on Ω coincides with the set of minimizers of the function*

$$F(x, A) = f(x) + A\sum_{i\in I} h_i^+(x) \tag{20}$$

on $\mathbf{R}^n$. Here $h_i^+(x) = max\{0, h_i(x)\}$.

Remark 8 Thus, the constrained optimization problem (P) is reduced to the unconstrained one. Since the function $F(x, A)$ is again quasidifferentiable, one can use methods for unconstrained optimization. Another condition (different from (19)) under which Proposition 7 is valid was stated in [2].

Remark 9 Problem (P) is called a *d.c. programming problem* if all the functions f and h_i' $(i \in I)$ are d.c., i. e. $f = f_1 - f_2$, $h_i = h_{1i} - h_{2i}$ where f_1, f_2, h_{1i}, h_{2i} are convex. If the calmness condition (19) holds then, by Proposition 7, problem (P) is reduced to that of minimizing the function $F(x, A)$ (see (20)) (if A is sufficiently large). We have

$$h_i^+(x) = \max\{0, h_i(x)\} = \max\{0, h_{1i}(x) - h_{2i}(x)\} = \overline{h}_{1i}(x) - \overline{h}_{2i}(x) ,$$

where

$$\overline{h}_{1i}(x) = \max\{h_{1i}(x), h_{2i}(x)\} ,$$
$$\overline{h}_{2i}(x) = h_{1i}(x) + h_{2i}(x) .$$

The functions $\overline{h}_{1i}$ and $\overline{h}_{2i}$ are convex, therefore h_i^+ is d.c. and, hence, the function $F(x, A)$ is also d.c. and one may use the method described above for d.c. functions.

See also

- ► Generalized Monotonicity: Applications to Variational Inequalities and Equilibrium Problems
- ► Hemivariational Inequalities: Applications in Mechanics
- ► Hemivariational Inequalities: Eigenvalue Problems
- ► Hemivariational Inequalities: Static Problems
- ► Nonconvex Energy Functions: Hemivariational Inequalities
- ► Nonconvex-Nonsmooth Calculus of Variations

- ▶ Quasidifferentiable Optimization
- ▶ Quasidifferentiable Optimization: Algorithms for Hypodifferentiable Functions
- ▶ Quasidifferentiable Optimization: Applications
- ▶ Quasidifferentiable Optimization: Applications to Thermoelasticity
- ▶ Quasidifferentiable Optimization: Calculus of Quasidifferentials
- ▶ Quasidifferentiable Optimization: Codifferentiable Functions
- ▶ Quasidifferentiable Optimization: Dini Derivatives, Clarke Derivatives
- ▶ Quasidifferentiable Optimization: Exact Penalty Methods
- ▶ Quasidifferentiable Optimization: Optimality Conditions
- ▶ Quasidifferentiable Optimization: Stability of Dynamic Systems
- ▶ Quasidifferentiable Optimization: Variational Formulations
- ▶ Quasivariational Inequalities
- ▶ Sensitivity Analysis of Variational Inequality Problems
- ▶ Solving Hemivariational Inequalities by Nonsmooth Optimization Methods
- ▶ Variational Inequalities
- ▶ Variational Inequalities: F. E. Approach
- ▶ Variational Inequalities: Geometric Interpretation, Existence and Uniqueness
- ▶ Variational Inequalities: Projected Dynamical System
- ▶ Variational Principles

References

1. Demyanov VF (1995) Fixed point theorem in nonsmooth analysis and its applications. Numer Funct Anal Optim 16(1–2):53–110
2. Demyanov VF, Di Pillo G, Facchinei F (1998) Exact penalization via Dini and Hadamard conditional derivatives. Optim Methods Softw 8:19–36
3. Demyanov VF, Rubinov AM (1980) On quasidifferentiable functionals. Dokl USSR Acad Sci (2) 250(1):21–25
4. Demyanov VF, Rubinov AM (1995) Constructive nonsmooth analysis. P. Lang, Frankfurt am Main
5. Demyanov VF, Vasiliev LV (1986) Nondifferentiable optimization. Springer and Optim. Software, Berlin
6. Hiriart–Urruty J-B (1989) From convex optimization to nonconvex optimization. In: Clarke FH, Demyanov VF, Giannessi F (eds) Nonsmooth Optimization and Related Topics. Plenum, New York, pp 219–239
7. Rockafellar RT (1970) Convex analysis. Princeton Univ. Press, Princeton

Quasidifferentiable Optimization: Applications

GEORGIOS E. STAVROULAKIS
Carolo Wilhelmina Techn. University, Braunschweig, Germany

MSC2000: 74A55, 74M10, 74M15, 65K99, 90C26, 49J35

Article Outline

Keywords

Nonsmooth optimization; Quasidifferentiability; Nonsmooth modeling

Quasidifferentiability and the notion of the quasidifferential extend the subdifferential of convex analysis for a quite general class of nonconvex and nonsmooth, but directionally differentiable functions. By using an ordered pair of convex sets, the quasidifferential copes in a nice way with both nonsmoothness and nonconvexity issues. Since its introduction by V.F. Demyanov, a number of quasidifferential optimization problems have been studied. Moreover calculus rules have been developed and applications, among others in mechanics and engineering [5] have been considered. In addition, the related, more appropriate for numerical purposes notion of the codifferential has been introduced.

Let us consider a classical optimization algorithm, the (anti)gradient optimization, and how it is modified for quasidifferentiable functions. First, recall that a nondifferentiable function does not has derivatives in

the classical sense. One should use set-valued approximations at the points of the nondifferentiability instead. This replacement introduces the following two problems in a gradient optimization algorithm. First, one has to calculate an appropriate direction of descent, which should be followed at a given iteration step. Moreover, the optimality conditions for a nondifferentiable function have a form dictated by the set-valued approximation of the derivative (i. e., one should check, at a given point, if zero is element of a set, or if some relation between sets is satisfied). Several applications of the quasidifferentiability concept are briefly reviewed in this short article.

Nonsmooth Modeling

Let us recall first the notion of quasidifferentiability in the sense of Demyanov. A function f which is defined on an open set $X \subset \mathbf{R}^n$ and which is directionally differentiable at a point $x \in X$ is called *quasidifferentiable* if there exists an ordered pair of convex compact sets $[\underline{\partial} f(x), \overline{\partial} f(x)]$ in $\mathbf{R}^n \times \mathbf{R}^n$ which produces the directional derivative of the function by the following formula

$$f'(x,g) = \max_{w \in \underline{\partial} f(x)} \langle w, g \rangle + \min_{v \in \overline{\partial} f(x)} \langle v, g \rangle \ , \qquad (1)$$

for all directions $g \in \mathbf{R}^n$. More details are given in ► Quasidifferentiable optimization.

Relation (1) gives rise to a qualitative and quantitative nonsmooth approximation (quasilinearization) of a nonsmooth and nonconvex function f at point x. Theoretical results on nonsmooth modeling can be found, among others, in [8,9,14].

The notion of the quasidifferential gives rise to nonsmooth models, with applications in mechanics [5,11]. In particular, interesting nonconvex variational formulations can be written, as it is discussed in more detail in ► Quasidifferentiable optimization: Variational formulations. They extend the *variational inequalities*, which are valid for the convex, nondifferentiable case, and constitute a parallel development to the *hemivariational inequalities* in the sense of P.D. Panagiotopoulos (see also ► Nonconvex energy functions: Hemivariational inequalities; ► Hemivariational inequalities: Applications in mechanics as well as [12]). Furthermore, quasidifferential and codifferential optimization techniques can be used for the construction of numerical algorithms for problems of nonsmooth computational mechanics [5].

Nonsmooth and Nonconvex Optimization

The notion of the quasidifferential allows one to calculate one steepest descent direction of a quasidifferentiable function $f(x)$ at a given point x_0. Assume that at point x_0 one has the subdifferential $\underline{\partial} f(x_0)$ and the superdifferential $\overline{\partial} f(x_0)$. Then, a steepest descent direction h can be calculated by:

$$h = \frac{w_1^* + w_2^*}{\|w_1^* + w_2^*\|} \ , \qquad (2)$$

for $w_1^* \in \underline{\partial} f(x_0)$, $w_2^* \in \overline{\partial} f(x_0)$, such that

$$\|w_1^* + w_2^*\| = \max_{w_1 \in \underline{\partial} f(x_0)} \left\{ \min_{w_2 \in \overline{\partial} f(x_0)} \|w_1 + w_2\| \right\} \ .$$

Moreover, there exists necessary (and in some cases sufficient) set-valued optimality conditions for quasidifferentiable optimization problems (see ► Quasidifferentiable optimization). Thus one has whatever is needed for the construction of a numerical algorithm. Calculus rules exist for the construction of the quasidifferential (see ► Quasidifferentiable optimization: Calculus of quasidifferentials), if this is not obvious from the definition of the optimization problem. Stopping rules for an optimization algorithm can also be extracted. In fact, if the optimality criteria are satisfied, then (at least local) minimum point has been calculated. Otherwise, one can calculate a steepest descent direction by (2) and proceed with a (steepest descent like) numerical optimization scheme. In this respect the affiliated notion of the codifferentiability has certain advantages for the numerical implementation. More details can be found in ► Quasidifferentiable optimization: Codifferentiable functions and in [3,6].

It is worth noting to observe here that formula (2) may admit multiple solutions. This should be expected since one deals with nonconvex (global) optimization problems. This is actually one of the advantages of the quasidifferentiability concept since, theoretically, if one follows all possible directions of descent which may arise along an iterative algorithm one should be able to calculate multiple solutions (i. e., local minima).

More information on smooth (convex and nonconvex) optimization and appropriate algorithms devel-

oped for these problems can be found, among others, in [2,7]. Note also the multi-objective programming approach for the solution of systems of quasidifferentiable equations which has been developed in [13].

Multilevel and Marginal Function Optimization

Interesting results on the application of the quasidifferentiability concept for the sensitivity analysis and algorithms for multilevel optimization problems have been presented in [1,10].

Applications in nonsmooth mechanics

Quasidifferential modeling and optimization have been used for nonsmooth mechanics applications. As it is already mentioned these results can be found in [5,11]. A number of recent (2000) applications of quasidifferentiability can be found in [4].

See also

- ▶ Generalized Monotonicity: Applications to Variational Inequalities and Equilibrium Problems
- ▶ Hemivariational Inequalities: Applications in Mechanics
- ▶ Hemivariational Inequalities: Eigenvalue Problems
- ▶ Hemivariational Inequalities: Static Problems
- ▶ Nonconvex Energy Functions: Hemivariational Inequalities
- ▶ Nonconvex-Nonsmooth Calculus of Variations
- ▶ Quasidifferentiable Optimization
- ▶ Quasidifferentiable Optimization: Algorithms for Hypodifferentiable Functions
- ▶ Quasidifferentiable Optimization: Algorithms for QD Functions
- ▶ Quasidifferentiable Optimization: Applications to Thermoelasticity
- ▶ Quasidifferentiable Optimization: Calculus of Quasidifferentials
- ▶ Quasidifferentiable Optimization: Codifferentiable Functions
- ▶ Quasidifferentiable Optimization: Dini Derivatives, Clarke Derivatives
- ▶ Quasidifferentiable Optimization: Exact Penalty Methods
- ▶ Quasidifferentiable Optimization: Optimality Conditions
- ▶ Quasidifferentiable Optimization: Stability of Dynamic Systems
- ▶ Quasidifferentiable Optimization: Variational Formulations
- ▶ Quasivariational Inequalities
- ▶ Sensitivity Analysis of Variational Inequality Problems
- ▶ Solving Hemivariational Inequalities by Nonsmooth Optimization Methods
- ▶ Variational Inequalities
- ▶ Variational Inequalities: F. E. Approach
- ▶ Variational Inequalities: Geometric Interpretation, Existence and Uniqueness
- ▶ Variational Inequalities: Projected Dynamical System
- ▶ Variational Principles

References

1. Dempe S, Pallaschke D (1997) Quasidifferentiability of optimal solutions in parametric nonlinear optimization. Optim 40(1):1–24
2. Demyanov VF (ed) (1985) Nondifferentiable optimization: motivations and applications. Springer, Berlin
3. Demyanov VF, Rubinov AM (1995) Introduction to constructive nonsmooth analysis. P. Lang, Frankfurt am Main
4. Demyanov VF, Rubinov AM (eds) (2000) Quasidifferentiability and related topics. Kluwer, Dordrecht
5. Demyanov VF, Stavroulakis GE, Polyakova LN, Panagiotopoulos PD (1996) Quasidifferentiability and nonsmooth modelling in mechanics, engineering and economics. Kluwer, Dordrecht
6. Demyanov VF, Vasiliev LN (1985) Nondifferentiable optimization. Optim. Software, New York
7. Kiwiel KC (1985) Methods of descent for nondifferentiable optimization. Springer, Berlin
8. Kuntz L (1996) Topological aspects of nonsmooth optimization. Habilitation Thesis, Univ Karlsruhe
9. Kuntz L, Scholtes S (1994) Structural analysis of nonsmooth mappings, inverse functions and metric projections. J Math Anal Appl 188:346–386
10. Luderer B (1988) Primale Dekomposition quasidifferenzierbarer Optimierungsaufgaben. Diss, Fak Math Naturwissenschaft, Techn Univ Karl-Marx-Stadt
11. Panagiotopoulos PD (1988) Nonconvex superpotentials and hemivariational inequalities. Quasidiferentiability in mechanics. In: Moreau JJ, Panagiotopoulos PD (eds) Nonsmooth Mechanics and Applications. CISM Lecture Notes. Springer, Berlin, pp 211;176
12. Panagiotopoulos PD (1993) Hemivariational inequalities. Applications in mechanics and engineering. Springer, Berlin

13. Pielczyk A (1991) Numerical methods for solving systems of quasidifferentiable equations. PhD Thesis, Univ. Karlsruhe, published in: Math. Systems in Economics, vol 124. Anton Hain Verlag, Frankfurt am Main
14. Scholtes S (1994) Piecewise differentiable mappings: theory and applications. Habilitation Thesis, Univ Karlsruhe

Quasidifferentiable Optimization: Applications to Thermoelasticity

GEORGIOS E. STAVROULAKIS[1,2]
[1] Carolo Wilhelmina Techn. University, Braunschweig, Germany
[2] Technical University of Crete, Chania, Greece

MSC2000: 74B99, 74D99, 74G99, 74H99, 47S40, 35R70

Article Outline

Keywords

Thermoelasticity; Variational inequalities; Nonsmooth mechanics; Quasidifferentiability

Certain *semipermeability* or *temperature control* problems in *thermoelasticity*, which may be combined with analogous mechanical unilateral contact effects, can be formulated and studied in a unified framework by nonsmooth modeling techniques [7]. The theory of quasidifferentiable optimization, in the sense of V.F. Demyanov and A.M. Rubinov, provides a general framework for the treatment of both convex and nonconvex, nonsmooth modeling problems [1,2,3,4]. Coupled thermal and kinematical nonconvex unilateral effects will be modeled in the sequel by using the quasidifferentiable optimization approach. Analogous formulations which have been based on the notion of *hemivariational inequalities* have been proposed and studied for semipermeability and thermal problems by P.D. Panagiotopoulos et al. [6,7]. An extension to thermoviscoelasticity has recently been published in [5].

This short article is mainly based on the results presented in [4,7], where more details can be found.

Classical Thermoelastic Model

Let us consider a thermoelastic medium in the Euclidean space $\mathbf{R}^3$. A point is denoted by x and its coordinates with respect to a fixed Cartesian coordinate system $0x_1x_2x_3$ by x_i, $i = 1, 2, 3$. The time variable t takes values in the interval $[0, T] \subset \mathbf{R}$. Moreover, let $u = u(x, t)$ be the displacement of the material point x at time t with reference to the natural state of the body, which is characterized by zero stresses and a constant absolute temperature $\theta_0 > 0$. The density at point x of the natural state is denoted by $\rho = \rho(x)$ and the open, bounded, connected subset of $\mathbf{R}^3$ occupied by the body is denoted by Ω. As usual, the boundary Γ of Ω is assumed to be regular.

The behavior of a linear thermoelastic body is governed by the following constitutive equations for the stress tensor $\sigma = \{\sigma_{ij}\}$, $i = 1, 2, 3$, and the specific entropy deviation $\eta - \eta_0$ (where η_0 is the specific entropy of the natural state)

$$\begin{aligned}\sigma_{ij} &= t_{ij} - m_{ij}(\theta - \theta_0)\\ &= C_{ijhk}\varepsilon_{hk} - m_{ij}(\theta - \theta_0),\end{aligned} \tag{1}$$

$$\eta - \eta_0 = \frac{1}{\theta_0} c_D(\theta - \theta_0) + \frac{1}{\rho} m_{ij}\varepsilon_{ij}. \tag{2}$$

Here $\theta = \theta(x, t)$ is the absolute temperature, and $\varepsilon = \{\varepsilon_{ij}\}$ the strain tensor, which is related to the displacements by the small deformation elasticity relation

$$\varepsilon_{ij}(u) = \frac{1}{2}(u_{i,j} + u_{j,i}).$$

Here $C = \{C_{ijhk}\}$, $i, j, h, k = 1, 2, 3$, is the elasticity tensor, which satisfies the well-known symmetry and ellipticity conditions, $m = \{m_{ij}\}$ is the symmetry tensor of thermal expansion, and $c_D = c_D(x) > 0$ is the specific heat at zero strain of the body. $C(x)$, $m(x)$ and $c_D(x)$ are referred to the natural state of the body. The equations of motion read:

$$\rho u_i'' = \sigma_{ij,j} + f_i, \tag{3}$$

and the law of conservation of energy has the form

$$\rho\theta_0\eta' = -q_{i,i} + Q. \tag{4}$$

Here $f = \{f_i\}$, $f_i = f_i(x, t)$, is the volume force vector, $q = \{q_i\}$, $q_i = q_i(x, t)$, is the heat flux vector and $Q = Q(x, t)$ is the radiant heating per unit volume. *Fourier's law* of *heat conduction* reads:

$$q_i = -k_{ij}\theta_{,j}. \tag{5}$$

The symmetric tensor of thermal conductivity $k = \{k_{ij}\}$, $k_{ij} = k_{ij}(x)$, refers to the natural state of the body and satisfies the condition

$$k_{ij}a_i a_j \geq c a_i a_i, \quad \forall a = \{a_i\} \in \mathbb{R}^3,$$

where c is a positive constant. These relations lead to the following system of differential equations, which describe the *linear thermoelastic behavior of a generally nonhomogeneous and nonisotropic body*:

$$\rho u_i'' = f_i + (C_{ijhk}\varepsilon_{hk})_{,j} - \left(m_{ij}(\theta - \theta_0)\right)_{,j}, \quad \text{in } \Omega \times (0, T), \tag{6}$$

$$\rho c_D \theta' - (k_{ij}\theta_{,j})_{,i} + m_{ij}\theta_0 \varepsilon_{ij}' = Q, \quad \text{in } \Omega \times (0, T). \tag{7}$$

In the sequel the following initial conditions at $t = 0$ are assumed:

$$u_i = u_{0i}(x), \quad u_i' = u_{1i}(x) \quad \text{in } \Omega, \tag{8}$$

and

$$\theta = \overline{\theta}(x) \quad \text{in } \Omega. \tag{9}$$

Let the following bilinear forms be introduced:

$$\begin{aligned} a(u, v) &= \int_\Omega C_{ijhk}\varepsilon_{ij}(u)\varepsilon_{hk}(v)\, d\Omega, \\ (u, v) &= \int_\Omega u_i v_i \, d\Omega, \\ M_1(\theta, v) &= \int_\Omega (m_{ij}\theta)_{,j} v_i \, d\Omega, \\ M_2(u, \varphi) &= \int_\Omega m_{ij} u_{i,j} \varphi \, d\Omega, \\ K(\theta, \varphi) &= \int_\Omega k_{ij}\theta_{,j}\varphi_{,i} \, d\Omega, \\ (\theta, \varphi) &= \int_\Omega \theta\varphi \, d\Omega. \end{aligned} \tag{10}$$

Quasidifferential Thermal Boundary Conditions

In order to complete the description of the previous boundary value problem one needs to specify boundary conditions for the thermal and for the elasticity problem. First, let us assume that between the boundary temperature and the heat flux the following quasidifferential (QD) superpotential relation holds:

$$\begin{aligned} &q_i n_i = -k_{ij}\theta_{,j} n_i = w_1(\theta, t) + w_2(\theta, t), \\ &\text{with } \{w_1(\theta, t), w_2(\theta, t)\} \in Dj(\theta, t), \\ &\text{on } \Gamma_1 \times (0, T), \end{aligned} \tag{11}$$

where $\Gamma_1 \subset \Gamma$ and on the remaining part of the boundary one assumes, for simplicity, that:

$$\theta = 0 \quad \text{on } \Gamma - \Gamma_1. \tag{12}$$

For the displacements, a simple boundary condition is considered:

$$u_i = 0 \quad \text{on } \Gamma \times (0, T). \tag{13}$$

Here $n = \{n_i\}$ denotes, as usual, the unit normal to Γ directed towards the exterior of Ω.

Variational Formulation

One follows here the usual way for the construction of the variational or weak formulation of the previous boundary value problem (see also ▶ quasidifferentiable optimization: variational formulations; ▶ hemivariational inequalities: applications in mechanics). Let the virtual variations $v - u'$ and $\varphi - \theta$ are sufficiently smooth. Then, by multiplying (6) and (7) by $v - u'$ and $\varphi - \theta$ respectively, integrating over Ω, and using the Green—Gauss theorem, one obtains the variational equalities

$$\begin{aligned} &(\rho u'', v - u') + a(u, v - u') + M_1(\theta - \theta_0, v - u') \\ &= (f, v - u') + \int_\Gamma t_{ij} n_j (v_i - u_i')\, d\Gamma \\ &\text{in } \Omega \times (0, T) \end{aligned} \tag{14}$$

and

$$\begin{aligned} &(\rho c_D \theta', \varphi - \theta) + K(\theta, \varphi - \theta) + M_2(\theta_0 u', \varphi - \theta) \\ &= (Q, \varphi - \theta) + \int_\Gamma k_{ij}\theta_{,j} n_i (\varphi - \theta)\, d\Gamma \\ &\text{in } \Omega \times (0, T). \end{aligned} \tag{15}$$

Now let us assume that C_{ijhk}, k_{ij}, m_{ij}, $\rho > 0$ and $c_D > 0$ are elements of $L^\infty(\Omega)$, and that $f(t) \in [L^2(\Omega)]^3$ and $Q(t) \in L^2(\Omega)$. Moreover, the spaces $[\overset{0}{H}{}^1(\Omega)]^3$ for v, u′ and $H^1(\Omega)$ for φ, θ are introduced.

Let us recall that a QD boundary condition (for instance, the relation (11)), gives rise, due to the definition of the quasidifferential, to a min-max relation. This relation, is used for the formulation of nonconvex variational problems, as it is discussed in more details in ► quasidifferentiable optimization: variational formulations.

Thus, the variational equalities (14) and (15) are combined with the boundary conditions (11)–(13), and lead to the following variational problem: find functions u: $[0, T] \to [\overset{0}{H}{}^1(\Omega)]^3$ and θ: $[0, T] \to \Phi = H^1(\Omega)$: $\theta = 0$ on $\Gamma - \Gamma_1\}$, with $u'(t) \in [\overset{0}{H}{}^1(\Omega)]^3$, $u''(t) \in [L^2(\Omega)]^3$, $\theta'(t) \in L^2(\Omega)$, which satisfy the initial conditions and the variational expression

$$
\begin{aligned}
&(\rho u'', v - u') + a(u, v - u') + M_1(\theta - \theta_0, v - u') \\
&= (f, v - u'), \quad \forall v \in [\overset{0}{H}{}^1(\Omega)]^3,
\end{aligned}
\tag{16}
$$

and

$$
\begin{aligned}
&(\rho c_D \theta', \varphi - \theta) + K(\theta, \varphi - \theta) \\
&+ M_2(\theta_0 u', \varphi - \theta) \\
&+ \max\{\langle w_1^*, \phi - \theta\rangle : \ w_1^* \in \underline{\partial} J(\theta, t)\} \\
&+ \min\left\{\langle w_2^*, \phi - \theta\rangle : \ w_2^* \in \overline{\partial} J(\theta, t)\right\} \\
&= (Q, \varphi - \theta), \quad \forall \varphi \in \Phi.
\end{aligned}
\tag{17}
$$

Quasidifferential Elastic Boundary Conditions

Assume now simple thermal boundary conditions, i. e.,

$$
\theta = \theta_0 \quad \text{on } \Gamma \times (0, T). \tag{18}
$$

For the elasticity problem let a nonmonotone, possibly multivalued quasidifferential (QD) boundary law holds on a part Γ_S of the boundary Γ:

$$
\begin{aligned}
&-S = \{-S_i\} = \{-\sigma_{ij} n_i\} \\
&= S_1(u', x, t) + S_2(u', x, t), \\
&\{S_1(u', x, t), S_2(u', x, t)\} \in D\psi(u', x, t) \\
&\text{on } \Gamma_S \times (0, T).
\end{aligned}
\tag{19}
$$

On the remaining part of the boundary one assumes simply that:

$$
u_i = U_i \quad \text{on } \Gamma_U \times (0, T). \tag{20}
$$

Here $\Gamma = \overline{\Gamma}_U \cup \overline{\Gamma}_S$, where Γ_U and Γ_S are nonempty, disjoint, open sets, $U_i = U_i(x, t)$ is a prescribed displacement vector on Γ_U (which should be compatible with the initial conditions (8)–(9)).

In an analogous way one proceeds with the boundary value problem which is defined by the relations (6)–(9) and (18)–(19). Let v, $u' \in [H^1(\Omega)]^3$ be such that $v = u' = U'(t)$ on Γ_U and φ, $\theta \in H^1(\Omega)$ with $\varphi = \theta = \theta_0$ on Γ. In this case one gets the variational problem: find u: $[0, T] \to [H^1(\Omega)]^3$ with $u' = U$ on Γ_U and $\theta \in H^1(\Omega)$ with $\theta = \theta_0$ on Γ with $u'(t) \in [H^1(\Omega)]^3$, $u''(t) \in [L^2(\Omega)]^3$, $\theta'(t) \in L^2(\Omega)$, which satisfy the initial conditions and the variational expression

$$
\begin{aligned}
&(\rho u'', v - u') + a(u, v - u') \\
&+ M_1(\theta - \theta_0, v - u') \\
&+ \max\left\{\left\langle S_1^*, v - u'\right\rangle : \ S_1^* \in \underline{\partial} \Psi(u', t)\right\} \\
&+ \min\left\{\left\langle S_2^*, v - u'\right\rangle : \ S_2^* \in \partial \Psi(u', t)\right\} \\
&= (f, v - u'), \\
&\forall v \in [H^1(\Omega)]^3 \quad \text{with } v = U'(t) \text{ on } \Gamma_U
\end{aligned}
\tag{21}
$$

and

$$
\begin{aligned}
&(\rho c_D \theta', \varphi - \theta) + K(\theta, \varphi - \theta) \\
&+ M_2(\theta_0 u', \varphi - \theta) \\
&= (Q, \varphi - \theta), \\
&\forall \varphi \in H^1(\Omega) \quad \text{with } \varphi = \theta_0 \text{ on } \Gamma.
\end{aligned}
\tag{22}
$$

More general thermoelastic problems may be considered by considering QD laws for both the elasticity and the thermal part of the problem, or even mixed laws.

See also

- ► Generalized monotonicity: Applications to variational inequalities and equilibrium problems
- ► Hemivariational inequalities: Applications in mechanics
- ► Hemivariational inequalities: Eigenvalue problems
- ► Hemivariational inequalities: Static problems
- ► Nonconvex energy functions: hemivariational inequalities

- ▶ Nonconvex-nonsmooth calculus of variations
- ▶ Quasidifferentiable optimization
- ▶ Quasidifferentiable optimization: Algorithms for hypodifferentiable functions
- ▶ Quasidifferentiable optimization: Algorithms for QD functions
- ▶ Quasidifferentiable optimization: Applications
- ▶ Quasidifferentiable optimization: Calculus of quasidifferentials
- ▶ Quasidifferentiable optimization: Codifferentiable functions
- ▶ Quasidifferentiable optimization: Dini derivatives, clarke derivatives
- ▶ Quasidifferentiable optimization: Exact penalty methods
- ▶ Quasidifferentiable optimization: Optimality conditions
- ▶ Quasidifferentiable optimization: Stability of dynamic systems
- ▶ Quasidifferentiable optimization: Variational formulations
- ▶ Quasivariational inequalities
- ▶ Sensitivity analysis of variational inequality problems
- ▶ Solving hemivariational inequalities by nonsmooth optimization methods
- ▶ Variational inequalities
- ▶ Variational inequalities: F. E. approach
- ▶ Variational inequalities: Geometric interpretation, existence and uniqueness
- ▶ Variational inequalities: Projected dynamical system
- ▶ Variational principles

References

1. Demyanov VF, Rubinov AM (1983) On quasidifferentiable mappings. Math Operationsforsch Statist Ser Optim 14:3–21
2. Demyanov VF, Rubinov AM (1983) On quasidifferentiable mappings. Math Operationsforsch Statist Ser Optim 14:3–21
3. Demyanov VF, Rubinov AM (1995) Introduction to constructive nonsmooth analysis. P. Lang, Bern
4. Demyanov VF, Stavroulakis GE, Polyakova LN, Panagiotopoulos PD (1996) Quasidifferentiability and nonsmooth modelling in mechanics, engineering and economics. Kluwer, Dordrecht
5. Foutsitzi G, Stavroulakis GE (2004) A variational inequality approach to thermoviscoelasticity with monotone unilateral boundaries of kinematical and thermal type. Nonl Anal Theory Methods Appl 57:743–771
6. Haslinger J, Baniotopoulos CC, Panagiotopoulos PD (1993) A boundary multivalued integral "equation" approach to the semipermeability problem. Appl Mat 38:39–60
7. Panagiotopoulos PD (1985) Inequality problems in mechanics and applications. Convex and nonconvex energy functions. Birkhäuser, Basel

Quasidifferentiable Optimization: Calculus of Quasidifferentials

GEORGIOS E. STAVROULAKIS
Carolo Wilhelmina Techn. University, Braunschweig, Germany

MSC2000: 49J52, 65K99, 90C90

Article Outline

Keywords

Nonsmooth analysis; Nonsmooth modeling; Quasidifferentiability; Nonsmooth optimization

A function f defined on an open set $X \subset \mathbf{R}^n$ and directionally differentiable at a point $x \in X$ is called *quasidifferentiable* (in the sense of V.F. Demyanov) if there exists an ordered pair of convex compact sets $[U, V]$ in $\mathbf{R}^n \times \mathbf{R}^n$ which produces the directional derivative of the function by the following formula

$$f'(x, g) = \max_{h \in U}(h, g) + \min_{h \in V}(h, g)\,, \tag{1}$$

for all directions $g \in \mathbf{R}^n$.

Quasidifferentiability is a genuine generalization of the classical differentiability concept which is valid for smooth differentiable functions, and of the convex analysis subdifferential, which, in turn, is a set-valued differential valid for convex, nondifferentiable functions. An ordered pair of convex sets is used for the approximation of the directional derivative in (1). More

details can be found in the companion article ► Quasidifferentiable optimization, the links with other notions of nonsmooth analysis are discussed in ► Quasidifferentiable optimization: Dini derivatives, Clarke derivatives and applications are briefly presented in ► Quasidifferentiable optimization: Applications. One may also consult the original publications [2,3,4,5,6].

Using classical differential calculus and on the assumption of smooth (differentiable) functions the derivative of sums, of differences or of composite functions etc may easily be calculated. To this end one uses calculus rules and the derivatives of the involved functions (cf., the chain rule of differentiation). For quasidifferentiable functions there exist appropriate calculus rules [2]. The situation is more complicated here, since one manipulates ordered pairs of convex sets. Furthermore, calculus rules have been developed for composite functions which can be produced from a finite number of smooth constituents and from the application of a finite number of minimum or maximum operators. Moreover, as the quasidifferential of a given function is not uniquely determined (it is actually a class of equivalent ordered pairs of convex sets) one may wishes to simplify the results of such a calculus operation.

It is clear that, since quasidifferentials have found a number of applications, among others in optimization, in mechanics, in control theory and in economy, the need for refining the quasidifferential calculus and for incorporating it into automatic computational procedures (e.g, in computer algebra systems, in analogy to classical systems [1]) is obvious. For the latter task, which at the present remains open for future research efforts, use of results developed within the theory of interval analysis may be advantageous.

Calculus rules for one-dimensional functions (defined on $\mathbf{R}$) and for functions defined on $\mathbf{R}^n$ are given without proofs here. See [2,3] for more details.

One-Dimensional Case

A function in $\mathbf{R}^1$ and sets which are intervals of the real line $\mathbf{R}^1$ are considered first. Let D_1 and D_2 be two pairs of closed intervals: $D_1 = [A_1, B_1]$, $D_2 = [A_2, B_2]$, where $A_1 = [v_{11}, v_{12}]$, $B_1 = [w_{11}, w_{12}]$, $A_2 = [v_{21}, v_{22}]$, $B_2 = [w_{21}, w_{22}]$, with $v_{i1} \le v_{i2}$, $w_{i1} \le w_{i2}$, $\forall i \in \{1, 2\}$. Addition of intervals is defined as follows: $D = D_1 + D_2 = [A_1 + B_1, A_2 + B_2] = [A, B]$, where $A = [v_{11} + v_{21}, v_{12} + v_{22}]$ and $B = [w_{11} + w_{21}, w_{12} + w_{22}]$. Moreover, for $D = [A, B]$, $A = [v_1, v_2]$, $B = [w_1, w_2]$, $v_1 \le v_2$, $w_1 \le w_2$, multiplication by a scalar quantity λ is defined by:

$$\lambda D = \begin{cases} \lambda[A, B], & \lambda \ge 0, \\ \lambda[B, A], & \lambda < 0, \end{cases}$$

where, on the right-hand side one has $\lambda\,[A, B] = [[\lambda\, v_1, \lambda\, v_2], [\lambda\, w_1, \lambda\, w_2]]$, etc.

Based on these results concerning calculus of closed intervals one derives calculus rules for quasidifferentials in the one-dimensional case.

Let f_1 be a directionally differentiable function at a point x and let $Df_1(x) = [\underline{\partial} f_1(x), \overline{\partial} f_1(x)]$ be its quasidifferential at a point $x \in \mathbf{R}^1$: $\underline{\partial} f_1(x) = [v_{11}, v_{12}]$, $\overline{\partial} f_1(x) = [w_{11}, w_{12}]$, $v_{11} \le v_{12}$, $w_{11} \le w_{12}$. Then the function $f = \lambda\, f_1$ is also directionally differentiable at x and admits a quasidifferential of the form $Df(x) = [\underline{\partial} f(x), \overline{\partial} f(x)]$, where

$$\underline{\partial} f(x) = \begin{cases} [\lambda v_{11}, \lambda v_{12}], & \lambda \ge 0, \\ [\lambda w_{12}, \lambda w_{11}], & \lambda < 0, \end{cases}$$

$$\overline{\partial} f(x) = \begin{cases} [\lambda w_{11}, \lambda w_{12}], & \lambda \ge 0, \\ [\lambda v_{12}, \lambda v_{11}], & \lambda < 0. \end{cases}$$

If in addition, $f_1(x) \ne 0$ then the function $f = 1/f_1$ is also directionally differentiable at x and

$$Df(x) = -\frac{1}{f_1^2} Df_1(x) = [\underline{\partial} f(x), \overline{\partial} f(x)],$$

where

$$\underline{\partial} f(x) = \left[-\frac{1}{f_1^2} w_{12}, -\frac{1}{f_1^2} w_{11}\right],$$

$$\overline{\partial} f(x) = \left[-\frac{1}{f_1^2} v_{12}, -\frac{1}{f_1^2} v_{11}\right].$$

Let us consider two directionally differentiable functions f_1, f_2 at a point x and let $D f_1(x), D f_2(x)$ be their quasidifferentials

$$Df_1(x) = [\underline{\partial} f_1(x), \overline{\partial} f_1(x)],$$

$$Df_2(x) = [\underline{\partial} f_2(x), \overline{\partial} f_2(x)],$$

with the corresponding intervals denoted by:

$$\underline{\partial} f_1(x) = [v_{11}, v_{12}], \quad \overline{\partial} f_1(x) = [w_{11}, w_{12}],$$

$$\underline{\partial} f_2(x) = [v_{21}, v_{22}], \quad \overline{\partial} f_2(x) = [w_{21}, w_{22}],$$

$$v_{i1} \le v_{i2}, \quad w_{i1} \le w_{i2}, \qquad \forall i \in \{1, 2\}.$$

Then the function $f = f_1 + f_2$ is also directionally differentiable at x and one can take $Df(x) = [\underline{\partial} f(x), \overline{\partial} f(x)]$, where

$$\underline{\partial} f(x) = \underline{\partial} f_1(x) + \underline{\partial} f_2(x) = [v_1, v_2]$$
$$\overline{\partial} f(x) = \overline{\partial} f_1(x) + \overline{\partial} f_2(x) = [w_1, w_2] ,$$

with $v_1 = v_{11} + v_{21}$, $v_2 = v_{12} + v_{22}$, $w_1 = w_{11} + w_{21}$ and $w_2 = w_{12} + w_{22}$.

Analogously one proceeds with the product of two functions $f = f_1 f_2$, where

$$Df(x) = f_1(x)Df_2(x) + f_2(x)Df_1(x) = [\underline{\partial} f(x), \overline{\partial} f(x)] .$$

Furthermore, let $\varphi_i(x)$ $(i \in I = \{1, \ldots, N\})$ be directionally differentiable functions at a point x. The functions

$$f_1(x) = \max_{i \in I} \varphi_i(x) , \quad f_2(x) = \min_{i \in I} \varphi_i(x)$$

are also quasidifferentiable.

Finally, let $f(z_1, \ldots, z_m)$ be a smooth function and let $y_1, \ldots, y_m$ be quasidifferentiable functions at a point x_0. Then the function $F(x) = f(y_1(x), \ldots, y_m(x))$ is also quasidifferentiable at x_0.

One concludes that the family of quasidifferentiable functions is a linear space, closed with respect to all smooth operations, as well as the operations of taking the pointwise maximum and minimum over a finite number of functions.

Finite-Dimensional Case

In this case one needs calculus rules for pairs of convex sets of $\mathbf{R}^n$ (see, e. g. [4,6]).

Let the functions f, f_1, f_2 be quasidifferentiable at x and λ be a real number. Then the sum, the product, the function λf and the function $1/f(x)$ (or every point x such that $f(x) \neq 0$) are also quasidifferentiable and an element of their quasidifferential can be calculated as follows:

$$D(f_1 + f_2)(x) = Df_1(x) + Df_2(x) ,$$
$$D(f_1 \cdot f_2)(x) = f_1(x)Df_2(x) + f_2(x)Df_1(x) ,$$
$$D(\lambda f)(x) = \lambda Df(x) ,$$
$$D\left(\frac{1}{f}\right)(x) = -\frac{1}{f^2(x)} Df(x) .$$

Let moreover the functions $f_1, \ldots, f_m$ be defined on an open set $X \subset \mathbf{R}^n$ and be quasidifferentiable at $x \in X$. Then, the functions

$$\phi_1(x) = \max_{i \in \{1,\ldots,m\}} f_i(x), \phi_2(x) = \min_{i \in \{1,\ldots,m\}} f_i(x)$$

are quasidifferentiable at x as well. The following relations hold:

$$D\phi_j(x) = \left[\underline{\partial} \phi_j(x), \overline{\partial} \phi_j(x)\right] , \quad j = 1, 2 ,$$

with

$$\underline{\partial} \phi_1(x) = \operatorname{co} \bigcup_{k \in R(x)} \left(\underline{\partial} f_k(x) - \sum_{\substack{i \in R(x), \\ i \neq k}} \overline{\partial} f_i(x) \right) ,$$
$$\overline{\partial} \phi_1(x) = \sum_{k \in R(x)} \overline{\partial} f_k(x), \underline{\partial} \phi_2(x) = \sum_{k \in q(x)} \underline{\partial} f_k(x),$$
$$\overline{\partial} \phi_2(x) = \operatorname{co} \bigcup_{k \in Q(x)} \left(\overline{\partial} f_k(x) - \sum_{\substack{i \in Q(x), \\ i \neq k}} \underline{\partial} f_i(x) \right) .$$

Here, $[\underline{\partial} f_k(x), \overline{\partial} f_k(x)]$ is a quasidifferential of f_k at x and the following activity sets have been used:

$$R(x) = \{i \in I: \ f_i(x) = \phi_1(x)\} ,$$
$$Q(x) = \{i \in I: \ f_i(x) = \phi_2(x)\} ,$$

where $I = \{1, \ldots, n\}$.

Finally, consider the case of a composite function. Let a mapping $H(x) = (h_1(x), \ldots, h_m(x))$ be defined such that $H(x)$: $X \to Y$, where X is an open set in $\mathbf{R}^n$ and Y is an open set in $\mathbf{R}^m$ and every function h_i is quasidifferentiable at $x_0 \in X$. Let us assume that a function f is defined on Y and is Hadamard differentiable and quasidifferentiable at $y_0 = H(x_0)$. Then the composite function

$$\phi(x) = f(H(x))$$

is quasidifferentiable at x_0 and its quasidifferential $D\phi(x_0) = [\underline{\partial}\phi(x_0), \overline{\partial}\phi(x_0)]$ is expressed by the formulas:

$$\underline{\partial}\phi(x_0) = \left\{ p: \begin{array}{c} p = \sum_{i=1}^{m} (\nu^{(i)}(\lambda_i + \mu_i) - \underline{\nu}^{(i)}\lambda_i - \overline{\nu}^{(i)}\mu_i), \\ \nu = (\nu^{(1)}, \ldots, \nu^{(m)}) \in \underline{\partial} f(y_0), \\ \lambda_i \in \underline{\partial} h_i(x_0),\ \mu_i \in \overline{\partial} h_i(x_0) \end{array} \right\}$$

$$\overline{\partial}\phi(x_0) = \left\{ l: \begin{array}{c} l = \sum_{i=1}^{m} (\nu^{(i)}(\lambda_i + \mu_i) + \underline{\nu}^{(i)}\lambda_i + \overline{\nu}^{(i)}\mu_i), \\ \nu = (\nu^{(1)}, \ldots, \nu^{(m)}) \in \overline{\partial} f(y_0), \\ \lambda_i \in \underline{\partial} h_i(x_0),\ \mu_i \in \overline{\partial} h_i(x_0) \end{array} \right\},$$

where $\underline{\nu}$ and $\overline{\nu}$ are arbitrary vectors such that

$$\underline{\nu} \le \nu \le \overline{\nu}, \quad \forall \nu \in \underline{\partial} f(y_0) \cup \left(-\overline{\partial} f(y_0)\right).$$

Concrete examples and the derivation of the above rules can be found in the above given literature. One should only mention that if some of the involved sets (i. e., the subdifferential or the superdifferential) happens to be polyhedral, then certain of the previous rules can be simplified significantly (see, e. g., [7]). The latter case appears, among others, in the applications of quasidifferential calculus within a finite element method environment for applications in mechanics (see also ▶ Quasidifferentiable optimization: Variational formulations; ▶ Quasidifferentiable optimization: Applications to thermoelasticity, and [5]).

See also

- ▶ Generalized Monotonicity: Applications to Variational Inequalities and Equilibrium Problems
- ▶ Hemivariational Inequalities: Applications in Mechanics
- ▶ Hemivariational Inequalities: Eigenvalue Problems
- ▶ Hemivariational Inequalities: Static Problems
- ▶ Nonconvex Energy Functions: Hemivariational Inequalities
- ▶ Nonconvex-Nonsmooth Calculus of Variations
- ▶ Quasidifferentiable Optimization
- ▶ Quasidifferentiable Optimization: Algorithms for Hypodifferentiable Functions
- ▶ Quasidifferentiable Optimization: Algorithms for QD Functions
- ▶ Quasidifferentiable Optimization: Applications
- ▶ Quasidifferentiable Optimization: Applications to Thermoelasticity
- ▶ Quasidifferentiable Optimization: Codifferentiable Functions
- ▶ Quasidifferentiable Optimization: Dini Derivatives, Clarke Derivatives
- ▶ Quasidifferentiable Optimization: Exact Penalty Methods
- ▶ Quasidifferentiable Optimization: Optimality Conditions
- ▶ Quasidifferentiable Optimization: Stability of Dynamic Systems
- ▶ Quasidifferentiable Optimization: Variational Formulations
- ▶ Quasivariational Inequalities
- ▶ Sensitivity Analysis of Variational Inequality Problems
- ▶ Solving Hemivariational Inequalities by Nonsmooth Optimization Methods
- ▶ Variational Inequalities
- ▶ Variational Inequalities: F. E. Approach
- ▶ Variational Inequalities: Geometric Interpretation, Existence and Uniqueness
- ▶ Variational Inequalities: Projected Dynamical System
- ▶ Variational Principles

References

1. Beltzer AI (1990) Variational and finite element methods: A symbolic computation approach. Springer, Berlin
2. Demyanov VF, Dixon LCW (1986) Quasidifferential calculus. Math Program Stud, vol 29. North-Holland, Amsterdam
3. Demyanov VF, Rubinov AM (1985) Quasidifferentiable calculus. Optim. Software, New York
4. Demyanov VF, Rubinov AM (1995) Introduction to constructive nonsmooth analysis. P. Lang, Frankfurt am Main
5. Demyanov VF, Stavroulakis GE, Polyakova LN, Panagiotopoulos PD (1996) Quasidifferentiability and nonsmooth modelling in mechanics, engineering and economics. Kluwer, Dordrecht
6. Demyanov VF, Vasiliev LN (1985) Nondifferentiable optimization. Optim. Software, New York
7. Stavroulakis GE, Polyakova LN (1996) Difference convex optimization techniques in nonsmooth computational mechanics. Optim Methods Softw 7(1):55–87

Quasidifferentiable Optimization: Codifferentiable Functions

GEORGIOS E. STAVROULAKIS
Carolo Wilhelmina Techn. University, Braunschweig, Germany

MSC2000: 65K99, 70-08, 49J52, 90C25

Article Outline

Keywords

Nonsmooth optimization; Quasidifferentiability; Codifferentiability; Nonconvex optimization

If for a directionally differentiable function there exists an ordered pair of convex compact sets $[U, V]$ in $\mathbf{R}^n \times \mathbf{R}^n$ which produces the directional derivative of f at x in the direction g by the expression:

$$f'(x,g) = \max_{h\in U}(h,g) + \min_{h\in V}(h,g)\,; \tag{1}$$

this function is called *quasidifferentiable* in the sense of V.F. Demyanov and A.M. Rubinov. This notion covers a large number of structured nonconvex and nonsmooth functions, which can be used for the solution of nonconvex and global optimization problems. For instance, the class of difference convex functions is included.

More details on this notion, the calculus rules for computing quasidifferentials, its connection to other notions of nonsmooth analysis and it applications can be found in ► Quasidifferentiable optimization; ► Quasidifferentiable optimization: Calculus of quasidifferentials; ► Quasidifferentiable optimization: Dini derivatives, Clarke derivatives; ► Quasidifferentiable optimization: Applications, as well as in [1,2,3].

The quasidifferential, as well as the subdifferential of *convex analysis*, are set-valued quantities which include discontinuities at the points of nondifferentiability. In numerical algorithms this may cause problems. A notion that takes into account neighboring information would be more appropriate. This led Demyanov to extend the notion of the quasidifferential by introducing the *codifferential*. Accordingly, the notions of *subdifferential* and *superdifferential* are extended to the notions of *hypodifferential* and *hyperdifferential*. One should mention that all quasidifferentiable functions are codifferentiable as well. Moreover, calculus rules exists, in analogy to the quasidifferential calculus rules.

These notions, which are useful for the construction of numerical algorithms in nonsmooth optimization [1] and nonsmooth computational mechanics [2] are introduced in this short paper. More details are given in the cited literature and in the previously mentioned lemmas.

Codifferentiable Functions

Let X be an open subset of $\mathbf{R}^n$ and let a function f be defined and finite for every $x \in X$. A function f is called *codifferentiable* at x if there exist convex compact sets $\underline{d}f(x) \subset \mathbf{R}^{n+1}$ and $\overline{d}f(x) \subset \mathbf{R}^{n+1}$ such that the function admits a first order approximation in a neighborhood of x of the form

$$f(x+\Delta) = f(x) + \max_{[\alpha,v]\in \underline{d}f(x)} [\alpha + (v,\Delta)] + \min_{[b,w]\in\overline{d}f(x)} [b + (w,\Delta)] + o_x(\Delta)\,, \tag{2}$$

where $o_x(\alpha\ \Delta)/\alpha \to 0$ as $\alpha \downarrow 0$, $\forall \Delta \in \mathbf{R}^n$. The ordered pair of convex compact sets $Df(x) = [\underline{d}f(x), \overline{d}f(x)]$ is called a *codifferential* of f at x, where $df(x)$ is a *hypodifferential* and $\overline{d}f(x)$ is a *hyperdifferential*.

If moreover there exists a codifferential Df which is Hausdorff continuous in a neighborhood of x, the function f is called *continuously codifferentiable* at x.

If there exists a codifferential of the form $Df(x) = [df(\mathbf{x}), \{0\}]$, the function f is called *hypodifferentiable*, while if there exists a codifferential of the form $Df(x) = [\{0\}, \overline{d}f(x)]$ the function is called *hyperdifferentiable*.

Note here that for a continuously codifferentiable function the first order approximation which is based on (2) is a continuous function in both x and Δ (recall that the analogous approximation based on the quasidifferential is a continuous function of only Δ).

Twice Codifferentiable Functions

Twice codifferentiable functions present a suitable tool for constructing higher order approximations of nondifferentiable functions. They extend the notion of second order derivatives of classical smooth analysis.

Let a function f be defined on an open set $X \subset \mathbf{R}^n$ and let it be finite there. The function f is *twice codifferentiable* at $x \in X$ if there exist convex compact sets $d^2 f(x)$, and $\overline{d}^2 f(x)$, both subsets of $\mathbf{R} \times \mathbf{R}^n \times \mathbf{R}^{n+n}$ such that

$$f(x+\Delta) = f(x) + \max_{[\alpha,v,A]\in \underline{d}^2 f(x)} \left[\alpha + (v,\Delta) + \frac{1}{2}(A\Delta,\Delta)\right] + \min_{[b,w,B]\in \overline{d}^2 f(x)} \left[b + (w,\Delta) + \frac{1}{2}(B\Delta,\Delta)\right] + o(\Delta^2),$$

with $o((\alpha\ \Delta)^2)/\alpha^2 \to 0$ as $\alpha \downarrow 0$ and $\forall \Delta \in \mathbf{R}^n$. Here $\mathbf{R}^{n+n}$ is the space of real $(n \times n)$-matrices.

The ordered pair of convex sets $D^2 f(x) = [\underline{d}^2 f(x), \overline{d}^2 f(x)]$ is called a *second order codifferential* of f at x, the set $d^2 f(x)$ is a *second order hypodifferential* and the set $d^2 f(x)$ is a *second order hyperdifferential* of f at x. Moreover, if f is twice codifferentiable in some neighborhood of a point x and the mapping $D^2 f$ is Hausdorff continuous at x, then the function f is called *twice continuously codifferentiable* at x.

Analogously to the quasidifferentiable and codifferentiable functions, twice hypodifferentiable functions and twice hyperdifferentiable functions may be defined. Calculus rules do also exist for twice codifferentiable functions (see [1, p. 216]).

For example, let f be convex and finite on a convex set $X \subset \mathbf{R}^n$, $x \in X$, and let X_0 be an arbitrary closed convex and bounded subset of X with $x \in \operatorname{int} X_0$. In this case one may consider the second order codifferential $D^2 f(x) = [d^2 f(x), 0]>$, with

$$\underline{d}^2 f(x) = \operatorname{co}\left\{ [\alpha, v, A] : \begin{array}{c} \alpha = f(z) - f(x) \\ +(v(z), x - z), \\ v(z) \in \underline{\partial} f(z), \\ A = 0 \in \mathbf{R}^{n+n}, \\ z \in X_0 \end{array} \right\}.$$

Here $v(z) \in \partial f(z)$ is an arbitrary element of the set valued mapping, which is kept fixed for every $z \in X_0$ and $\partial f(z)$ is equal to the classical convex analysis subdifferential.

Moreover, for a twice continuously differentiable function f it is well-known that

$$f(x+\Delta) = f(x) + (f'(x), \Delta) + \frac{1}{2}(f''(x)\Delta, \Delta) + o(\Delta^2),$$

where $f''(x)$ is the matrix of second order derivatives (Hessian) of f at x. The function f is twice continuously codifferentiable and one may consider (among other choices) one of the following second order codifferentials of f:

$$\underline{d}^2 f(x) = \{[0, f'(x), f''(x)]\},$$
$$\overline{d}^2 f(x) = \{0, 0, 0\},$$

or

$$\underline{d}^2 f(x) = \{0, 0, 0\},$$
$$\overline{d}^2 f(x) = \{[0, f'(x), f'(x)]\}.$$

Applications

Efficient nonsmooth optimization algorithms can be constructed based on the notion of the codifferential, or, for hypodifferentiable functions, on the notion of the hypodifferential. In fact, the technique of replacing a nondifferentiable optimization problem by an enlarged, classical, inequality constrained optimization problem has been successfully used for convex or for composite optimization problems [4,15]. For hypodifferentiable functions a direction of descent at each given point can be determined and used in an iterative optimization procedure. For general, codifferentiable functions, several directions of descent can be determined. This can be expected, given that one deals with nonconvex, global optimization problems. Some details in this direction are given in ► Quasidifferentiable optimization: Applications and in the original publications [3,11].

Furthermore, twice (or higher order) quasidifferentials and codifferentials provide set-valued approximations of the higher order derivatives of a function. For numerical optimization tasks this information may lead to more efficient algorithms, in analogy to the use of Hessian matrices in classical, smooth optimization. Other attempts for generalized second order derivatives can be found, for convex functions in [5] and

for nonconvex functions in [10,12]. For more information in the area of nonsmooth optimization see, e. g., [6,7,8,9,13].

Another area of interest for practical applications will be the use of this information for the construction of necessary and sufficient (local)optimality conditions. Applications of these results include stability and sensitivity analysis for quasidifferentiable and codifferentiable optimization problems. In mechanics, this information can be used for the study of the stability of structures governed by quasidifferentiable superpotentials (cf. e. g., [14] and ► Quasidifferentiable optimization: Stability of dynamic systems). Applications in economics will be of interest as well. Much work remains to be done in this area, which is open for further investigations (as of 1999).

See also

- ► Generalized Monotonicity: Applications to Variational Inequalities and Equilibrium Problems
- ► Hemivariational Inequalities: Applications in Mechanics
- ► Hemivariational Inequalities: Eigenvalue Problems
- ► Hemivariational Inequalities: Static Problems
- ► Nonconvex Energy Functions: Hemivariational Inequalities
- ► Nonconvex-nonsmooth Calculus of Variations
- ► Quasidifferentiable Optimization
- ► Quasidifferentiable Optimization: Algorithms for Hypodifferentiable Functions
- ► Quasidifferentiable Optimization: Algorithms for QD Functions
- ► Quasidifferentiable Optimization: Applications
- ► Quasidifferentiable Optimization: Applications to Thermoelasticity
- ► Quasidifferentiable Optimization: Calculus of Quasidifferentials
- ► Quasidifferentiable Optimization: Dini Derivatives, Clarke Derivatives
- ► Quasidifferentiable Optimization: Exact Penalty Methods
- ► Quasidifferentiable Optimization: Optimality Conditions
- ► Quasidifferentiable Optimization: Stability of Dynamic Systems
- ► Quasidifferentiable Optimization: Variational Formulations
- ► Quasivariational Inequalities
- ► Sensitivity Analysis of Variational Inequality Problems
- ► Solving Hemivariational Inequalities by Nonsmooth Optimization Methods
- ► Variational Inequalities
- ► Variational Inequalities: F. E. Approach
- ► Variational Inequalities: Geometric Interpretation, Existence and Uniqueness
- ► Variational Inequalities: Projected Dynamical System
- ► Variational Principles

References

1. Demyanov VF, Rubinov AM (1995) Introduction to constructive nonsmooth analysis. P. Lang, Frankfurt am Main
2. Demyanov VF, Stavroulakis GE, Polyakova LN, Panagiotopoulos PD (1996) Quasidifferentiability and nonsmooth modelling in mechanics, engineering and economics. Kluwer, Dordrecht
3. Demyanov VF, Vasiliev LN (1985) Nondifferentiable optimization. Optim. Software, New York
4. Fletcher R (1990) Practical methods of optimization. Wiley, New York
5. Hiriart-Urruty J-B (1986) A new set-valued second-order derivative for convex functions. In: Fermat-Days 85: Mathematics for Optimization. Math Stud, vol 129. North-Holland, Amsterdam, pp 157–182
6. Hiriart-Urruty J-B, Lemaréchal C (1993) Convex analysis and minimization algorithms. Springer, Berlin
7. Kiwiel KC (1985) Methods of descent for nondifferentiable optimization. Springer, Berlin
8. Lemarechal C, Mifflin R (eds) (1978) Bundle methods in nonsmooth optimization. Pergamon, Oxford
9. Mäkelä MM, Neittaanmäki P (1992) Nonsmooth optimization. World Sci., Singapore
10. Pallaschke D, Recht P, Urbanski R (1987) On extensions of the second-order derivative. Bull Acad Polon Sci Ser Func Anal (1987):751–763
11. Polyakova LN (1986) On minimizing the sum of a convex function and a concave function. Math Program Stud 29:69–73
12. Recht P (1993) Generalized derivatives: An approach to a new gradient in non-smooth optimization. PhD Diss., Univ. Karlsruhe. Published in: Serie Math. Systems in Economics, vol 136. Anton Hain Verlag, Meisenheim
13. Schramm H, Zowe J (1992) A version of the bundle idea for minimizing a nonsmooth function: conceptual idea, con-

vergence analysis, numerical results. SIAM J Optim 2:121–152
14. Stavroulakis GE, Goeleven D, Panagiotopoulos PD (1995) Stability of elastic bodies with nonmonotone multivalued boundary conditions of the quasidifferentiable type. J Elasticity 41(2):137–149
15. Womersley RS, Fletcher R (1986) An algorithm for composite nonsmooth optimization problems. J Optim Th Appl 48:493–523

Quasidifferentiable Optimization: Dini Derivatives, Clarke Derivatives

GEORGIOS E. STAVROULAKIS
Carolo Wilhelmina Techn. University, Braunschweig, Germany

MSC2000: 49J52, 26E25, 52A27, 90C99

Article Outline

Keywords

Nonsmooth analysis; Clarke generalized gradient; Quasidifferentiability; Dini derivative

The notion of the *quasidifferential* in the sense of V.F. Demyanov and A.M. Rubinov [5] constitutes a set-valued extension of the classical differential, which is appropriate for nonsmooth and generally nonconvex but directionally differentiable functions. This class of functions covers a large number of applications in nonsmooth analysis and, among others, includes the popular in *global optimization* class of difference convex functions. The quasidifferential approximates the directional derivative of a function by using an ordered pair of convex sets, the subdifferential and the superdifferential. Definitions are given in ► Quasidifferentiable optimization. Information on the corresponding calculus can be found in ► Quasidifferentiable optimization: Calculus of quasidifferentials.

Here, the relation between quasidifferentials and more classical notions in nonsmooth analysis is briefly addressed. In particular, the Dini directional derivatives and the F.H. Clarke [2,3] derivatives are considered. Other notions of nonsmooth analysis may be found, among others, in the recent publications [1,3,11].

Dini Derivatives

The *Dini upper derivative* of a function $f\colon \mathbf{R}^n \to \overline{\mathbf{R}}$ at a point $x \in \operatorname{dom} f$ in a direction $g \in \mathbf{R}^n$ is defined by:

$$f_{\mathrm{D}}^{\uparrow} = \limsup_{\alpha \downarrow 0} \frac{1}{\alpha}[f(x, \alpha g) - f(x)] \,. \tag{1}$$

Note that the upper limit in (1) is not necessarily finite. Analogously, the *Dini lower derivative* of f at x is defined by the relation

$$f_{\mathrm{D}}^{\downarrow} = \liminf_{\alpha \downarrow 0} \frac{1}{\alpha}[f(x, \alpha g) - f(x)] \,.$$

Recall that if the limit

$$f'(x, g) = \lim_{\alpha \downarrow 0} \frac{1}{\alpha}[f(x + \alpha g) - f(x)]$$

exists it is called the *derivative* of a function f at a point x in a direction g, or the *Dini derivative* and it is denoted by ${f_D}'(x, g)$.

Since the Dini derivative (resp. the Dini upper or lower derivative) is just the one-sided (resp. the one-sided upper or lower) derivative of an ordinary real-valued function, one can uses the methods developed to study functions of one variable. Thus, for instance, calculus rules for directional derivatives can be constructed.

A function f defined on an open set Ω is called *Dini uniformly directionally differentiable* at a point $x \in \Omega$ if it is directionally differentiable at x and there exists a real number α_0 such that

$$\begin{aligned} &\frac{1}{\alpha}[f(x + \alpha g) - f(x) - \alpha f'(x, g)] < \epsilon \,, \\ &\forall \alpha \in (0, \alpha_0) \,, \quad \forall g \in S \,, \end{aligned} \tag{2}$$

where $S = \{g \,|g| = 1$ is the unit sphere. By setting $\alpha\, g = v$ in (2) one gets:

$$\begin{aligned} &\left|f(x + v) - f(x) - f'_x(v)\right| < \epsilon \,\|v\| \,, \\ &\forall v \text{ such that } \|v\| \le \alpha_0 \,. \end{aligned}$$

Thus the uniform directional differentiability means that $\frac{1}{\|v\|}\left|f(x+v)-f(x)-f'_x(v)\right|$ tends to zero, as $\|v\| \to 0$.

More details on Dini derivatives and their use in optimization can be found in [9].

Clarke Derivatives

Let us consider the upper and lower Dini derivatives of a function f for a fixed direction g, i. e. the functions $x \to f_D^{\uparrow}(x,g)$ and $x \to f_D^{\downarrow}(x,g)$. Let us also consider the upper (resp. the lower) regularizations of these functions:

$$\overline{f}_{\mathrm{D}}^{\uparrow}(x,g) = \max\left\{ f_{\mathrm{D}}^{\uparrow}(x,g), \limsup_{x' \to x} f_{\mathrm{D}}^{\uparrow}(x,g)\right\},$$

respectively

$$\overline{f}_{\mathrm{D}}^{\downarrow}(x,g) = \min\left\{ f_{\mathrm{D}}^{\downarrow}(x,g), \liminf_{x' \to x} f_{\mathrm{D}}^{\downarrow}(x,g)\right\}.$$

For a Lipschitz function f, the upper and lower Dini derivatives are bounded in some neighborhood of x, hence both previous limits are finite.

The Clarke upper and lower derivatives are defined as upper and lower regularizations of the Dini upper and lower derivatives, i. e.

$$f_{\mathrm{CL}}^{\uparrow}(x,g) = \overline{f}_{\mathrm{D}}^{\uparrow}(x,g),$$
$$f_{\mathrm{CL}}^{\downarrow}(x,g) = \overline{f}_{\mathrm{D}}^{\downarrow}(x,g).$$

For the initial, equivalent definition of these quantities, see [2,6 p. 69]. Here the approach of [8] has been followed. For every fixed direction g, the function $x \to f_{CL}^{\uparrow}(x,g)$ is upper semicontinuous and the function $x \to f_{CL}^{\downarrow}(x,g)$ is lower semicontinuous.

It is appropriate to recall here some properties of the Clarke derivatives. For every fixed point x, the function $g \to f_{CL}^{\uparrow}(x,g)$ is sublinear and the function $g \to f_{CL}^{\downarrow}(x,g)$ is superlinear, thus the subdifferential $\partial f_{CL}^{\uparrow}(x,g)$ and the superdifferential $\overline{\partial} f_{\mathrm{CL}}^{\downarrow}(x,g)$ can be determined, such that

$$f_{\mathrm{CL}}^{\uparrow}(x,g) = \max_{l \in \underline{\partial} f_{\mathrm{CL}}^{\uparrow}(x,g)} (l,g),$$
$$f_{\mathrm{CL}}^{\downarrow}(x,g) = \min_{w \in \overline{\partial} f_{\mathrm{CL}}^{\downarrow}(x,g)} (w,g).$$

Moreover, the following relations hold

$$f_{\mathrm{CL}}^{\uparrow}(x,-g) = (-f)_{\mathrm{CL}}^{\uparrow}(x,-g),$$
$$f_{\mathrm{CL}}^{\uparrow}(x,g) = -f_{\mathrm{CL}}^{\uparrow}(x,-g).$$

From the above properties it results that

$$\max_{l \in \underline{\partial} f_{\mathrm{CL}}^{\uparrow}(x,g)} (l,g) = \max_{w \in \overline{\partial} f_{\mathrm{CL}}^{\downarrow}(x,g)} (w,g);$$

thus the two compact convex sets coincide. The Clarke subdifferential is thus defined as

$$\partial_{\mathrm{CL}} f(x) = \underline{\partial} f_{\mathrm{CL}}^{\uparrow}(x,g) = \overline{\partial} f_{\mathrm{CL}}^{\downarrow}(x,g).$$

The mapping $x \to \partial_{\mathrm{CL}} f(x)$ is upper semicontinuous. An element of the Clarke subdifferential is called a *generalized gradient* of f at x.

Concerning the relation between the directional derivative of the function (if it is directionally differentiable) and the Clarke upper and lower derivatives one has, in general,

$$f_{\mathrm{CL}}^{\downarrow}(x,g) \le f'(x,g) \le f_{\mathrm{CL}}^{\uparrow}(x,g). \tag{3}$$

Thus Clarke upper and lower derivatives are a sublinear majorant and a superlinear minorant of $f'(x,g)$ respectively. Only in the case of an u.s.c. (resp. l.s.c.) directional derivative $f'(x,g)$ the second (resp. the first) inequality in (3) holds as an equality. The latter property is considered to be the major drawback of the Clarke subdifferential in nonsmooth analysis applications, because it does not always gives rise to an approximation of the directional derivative at the points of nondifferentiability.

For further reference we recall here the necessary optimality conditions for a locally Lipschitz function f at a point x:

$$0 \in \partial_{\mathrm{CL}} f(x).$$

Note also that since approximations of sets and functions are linked, the notion of Clarke subdifferential gives rise to a notion for the generalized tangent cone (and respectively a generalized normal cone). The reader is referred to [2,3] [6, p. 83], [10] for more details.

Quasidifferential and Clarke Subdifferential

Before giving some information on the links between the quasidifferential and the Clarke subdifferential, some elements on the several definitions of the difference of convex compact sets are in order.

Differences of Convex Sets

Before stating the definition, some introductory material must be given. The max-face of a compact set U generated by $x \in \mathbf{R}^n$ is defined by

$$G_x(U) = \left\{ h \in U \colon (h, x) = \max_{g \in U} (x, g) \right\}.$$

Recall that the max-face set coincides with the subdifferential of the *support function* of U, i. e. $G_x(U) = \underline{\partial} p_U(x)$. Recall also that for a convex function defined on $\mathbf{R}^n$ the set of points of $T \subset \mathbf{R}^n$ where max-face is a singleton is of measure zero is a set of full measure (with respect to $\mathbf{R}^n$, i. e. $\mathbf{R}^n \setminus T$ is a set of measure zero).

The difference of two sets U and V, $U \dot{-} V$ is defined on the set of full measure T where both $G_x(U)$ and $G_x(V)$ are singletons by:

$$U \dot{-} V = \operatorname{clco} \{ \nabla p_U(x) - \nabla p_V(x) \colon x \in T \},$$

where ∇g denotes the gradient of function g. One may observe here that if $U = V + W$ then $U \dot{-} V = W$.

An equivalent definition of $U \dot{-} V$ is given by

$$U \dot{-} V = \operatorname{clco} \left(\bigcup_{x \in T_{U,V}} [G_x(U) - G_x(V)] \right), \tag{4}$$

where the dependence of T on both U and V is explicitly indicated.

An extension of (4) leads to the quasidifference operation $\ddot{-}$, defined by

$$U \ddot{-} V = \operatorname{clco} \left(\bigcup_{x \neq 0} [G_x(U) - G_x(V)] \right). \tag{5}$$

Unfortunately $\ddot{-}$ is not invariant with respect to the equivalence relation $\approx$. Nevertheless an estimate of the form $U \ddot{-} V \supset U \dot{-} V$, for every sets U and V always holds and in some cases there exist conditions under which the inclusion holds as an equality (see e. g. [6, p. 117]).

Estimation Results

The Clarke subdifferential can also be generated, in some cases, by the set operators difference $\dot{-}$ and quasidifference $\ddot{-}$ applied on the subdifferential $\overline{\partial} f(x)$ and the superdifferential $\underline{\partial} f(x)$ of a quasidifferentiable function f(x).

Under appropriate assumptions on f, and for appropriate choice of the elements of the subdifferential and the subdifferential of f at x, an estimate of the following form can be extracted:

$$A \subset \partial_{\mathrm{CL}} f(x) \subset B.$$

with set $A = \underline{\partial} f(x) \dot{-} (-\overline{\partial} f(x))$ and set $B = \underline{\partial} f(x) \ddot{-} (-\overline{\partial} f(x))$, as it is discussed in [6, pp. 143–155]. A different approach to the study of the relationship between the Clarke subdifferential and the quasidifferential is followed in [7] (see also [6, pp. 156–159]). More details in this direction can also be found in [4].

See also

- ▶ Generalized Monotonicity: Applications to Variational Inequalities and Equilibrium Problems
- ▶ Hemivariational Inequalities: Applications in Mechanics
- ▶ Hemivariational Inequalities: Eigenvalue Problems
- ▶ Hemivariational Inequalities: Static Problems
- ▶ Nonconvex Energy Functions: Hemivariational Inequalities
- ▶ Nonconvex-Nonsmooth Calculus of Variations
- ▶ Quasidifferentiable Optimization
- ▶ Quasidifferentiable Optimization: Algorithms for Hypodifferentiable Functions
- ▶ Quasidifferentiable Optimization: Algorithms for QD Functions
- ▶ Quasidifferentiable Optimization: Applications
- ▶ Quasidifferentiable Optimization: Applications to Thermoelasticity
- ▶ Quasidifferentiable Optimization: Calculus of Quasidifferentials
- ▶ Quasidifferentiable Optimization: Codifferentiable Functions
- ▶ Quasidifferentiable Optimization: Exact Penalty Methods
- ▶ Quasidifferentiable Optimization: Optimality Conditions
- ▶ Quasidifferentiable Optimization: Stability of Dynamic Systems
- ▶ Quasidifferentiable Optimization: Variational Formulations
- ▶ Quasivariational Inequalities
- ▶ Sensitivity Analysis of Variational Inequality Problems

- ▶ Solving Hemivariational Inequalities by Nonsmooth Optimization Methods
- ▶ Variational Inequalities
- ▶ Variational Inequalities: F. E. Approach
- ▶ Variational Inequalities: Geometric Interpretation, Existence and Uniqueness
- ▶ Variational Inequalities: Projected Dynamical System
- ▶ Variational Principles

References

1. Aubin P, Frankowska H (1991) Set–valued analysis. Birkhäuser, Basel
2. Clarke FH (1983) Optimization and nonsmooth analysis. Wiley/Interscience, New York. Reprinted as: Classics Appl. Math., vol 5. SIAM, 1990
3. Clarke FH, Ledyaev SYu, Stern RJ, Wolenski PR (1998) Nonsmooth analysis and control theory. Springer, Berlin
4. Demyanov VF, Jeyakumar V (1997) Hunting for a smaller convex subdifferential. J Global Optim 13(10):305–326
5. Demyanov VF, Rubinov AM (1980) On quasidifferentiable functionals. Soviet Math Dokl 21:14–17
6. Demyanov VF, Rubinov AM (1995) Introduction to constructive nonsmooth analysis. P. Lang, Frankfurt am Main
7. Demyanov VF, Sutti C (1995) Representation of the Clarke subdifferential for a regular quasidifferentiable function. J Optim Th Appl 87(3):553–562
8. Demyanov VF, Vasiliev LN (1985) Nondifferentiable optimization. Optim. Software, New York
9. Ioffe AD (1984) Calculus of Dini subdifferentials of functions and contingent coderivatives of set-valued maps. Nonlinear Anal Th Methods Appl 8(5):517–539
10. Panagiotopoulos PD (1985) Inequality problems in mechanics and applications. Convex and nonconvex energy functions. Birkhäuser, Basel
11. Rockafellar RT, WetsRJ-B (1998) Variational analysis. Springer, Berlin

Quasidifferentiable Optimization: Exact Penalty Methods

LYUDMILA N. POLYAKOVA
St. Petersburg State University, St. Petersburg, Russia

MSC2000: 90Cxx

Article Outline

Keywords

Quasidifferential functions; Quasidifferential; Exact penalty function; Exact penalty parameter; Regularity

Penalty function methods are used for solving many constrained optimization problems of the form: Find

$$\inf_{x\in X} f(x) = f^* \,, \tag{1}$$

where f is a locally Lipschitz quasidifferentiable function defined on $\mathbf{R}^n$, $\mathcal{D}f(x) = [\overline{\partial} f(x), \underline{\partial} f(x)]$ is its quasidifferential at a point $x \in \mathbf{R}^n$, $X \subset \mathbf{R}^n$ is a closed set.

It is always possible to define X in the form (see [2])

$$X = \{x \in \mathbf{R}^n : \ \varphi(x) = 0\} \,, \tag{2}$$

where φ is also a locally Lipschitz quasidifferentiable function defined on $\mathbf{R}^n$, the pair of sets $\mathcal{D}\varphi(x) = [\overline{\partial}\varphi(x), \underline{\partial}\varphi(x)]$ is a quasidifferential of φ at $x \in \mathbf{R}^n$ and

$$\varphi(x) > 0 \,, \quad \forall x \notin X \,.$$

Thus the set X is the set of global minimum points of the function φ on $\mathbf{R}^n$. Hence, it is closed. We shall assume that the set $X \subset \mathbf{R}^n$ is not empty and bounded.

As the function φ is quasidifferentiable then the following expansion holds:

$$\varphi(x + \alpha g) = \varphi(x) + \alpha\varphi'(x, g) + o(\alpha, x, g) \,,$$

where

$$\frac{o(\alpha, x, g)}{\alpha} \overset{\alpha\downarrow 0}{\to} 0 \,.$$

We shall assume that in this expansion at each point $x \in \mathbf{R}^n$ the convergence to 0 is uniform with respect to $g \in \mathbf{R}^n$, $\|g\| = 1$.

The idea of penalty function methods consists in reducing the problem (1) to a problem of the unconstrained optimization. Among the different approaches existing for such reduction we shall consider the method of exact penalty functions.

For solving the problem (1) a penalty function

$$F(c,x) = f(x) + c\varphi(x)$$

is introduced, where c is a nonnegative number, and then the problem

$$\inf_{x\in\mathbf{R}^n} F(c,x) \tag{3}$$

is considered.

We assume that $\inf_x \in \mathbf{R}^n F(c,x)$ is attained for every $c \geq 0$. In practice it would be useful to find conditions which guarantee that there exists an exact penalty parameter $c^* \geq 0$ such that the set

$$\left\{ x \in \mathbf{R}^n : \; x = \arg\min_{x\in\mathbf{R}^n} F(c^*,x) \right\}$$

coincides with the set

$$\left\{ x \in \mathbf{R}^n : \; x = \arg\min_{x\in X} f(x) \right\} .$$

At first such a problem was investigated in [5,10] for the problem of convex programming. Now there are many works in this field of mathematics. (See, for example, [1,4,6,8,9]).

The implementation of exact penalty function methods first of all depends on the properties of the function φ. Therefore various conditions are imposed on φ to make it possible to solve our problem. We shall consider some of them.

Regularity Condition 1

(See [3].)

We say that a *regularity condition* is satisfied for the function φ if for any boundary point $x^* \in \mathrm{bd}X$ there exist positive real numbers $\varepsilon(x^*)$ and $\beta(x^*)$, such that

$$\frac{o(\alpha,x,g)}{\alpha} > -\varphi'(x,g) + \beta(x^*)$$
$$= -\left[\max_{v\in\bar{\partial}\varphi(x)} \langle v,g\rangle + \min_{w\in\bar{\partial}\varphi(x)} \langle w,g\rangle \right] + \beta(x^*) ,$$
$$\forall x \in \mathcal{A}(X) \cap S_{\varepsilon(x^*)}(x^*) ,$$
$$\forall \alpha \in (0, \varepsilon(x^*)] ,$$
$$\forall g \in N(X,x) : \; \|g\| = 1 ,$$

where bd X is the set of boundary points of X,

$$S_{\varepsilon(x^*)}(x^*) = \{ x \in \mathbf{R}^n : \; \|x - x^*\| \leq \varepsilon(x^*) \} ,$$

$$\mathcal{A}(X) = \left\{ x \in \mathrm{bd}\, X : \begin{array}{c} \exists z \notin X : \\ x \text{ is a projection of } z \end{array} \right\} ,$$

$N(X,x)$ is the *normal cone* to the set X at the point $x \in X$:

$$N(X,x) = \left\{ g \in \mathbf{R}^n : \begin{array}{c} \langle g_1, g\rangle \leq 0, \\ \forall g_1 \in \Gamma(X,x) \end{array} \right\}$$

and

$$\Gamma(X,x) = \left\{ g \in \mathbf{R}^n : \begin{array}{c} \exists g_k \in \mathbf{R}^n, \alpha_k \geq 0, \\ g_k \to g, \; \alpha_k \downarrow 0, \\ x + \alpha_k g_k \in X \end{array} \right\} .$$

The regularity condition 1 is a condition about the behavior of the function φ only at the boundary points of the set X.

If for the function φ the regularity condition 1 holds, then there exists an exact penalty parameter c^*.

Since in practice the exact penalty parameter is a priori unknown, a sequence of real values c_k is constructed, satisfying the conditions

$$0 = c_0 < \cdots < c_k < \cdots ,$$
$$\lim_{k\to+\infty} c_k = +\infty .$$

Let us find

$$x(\cdot_k) = \arg\min_{x\in\mathbf{R}^n} F(\cdot_k, x) .$$

As a result, a decreasing sequence of real values $\{\varphi(x(c_k))\}$ is constructed. There exists an integer $K > 0$ such that $x(c_k) \in X$, $\forall k > K$. Thus, for $k > K$, the points $x(c_k)$ will be global minimum points of a function $F(c_k, x)$ on $\mathbf{R}^n$, i. e. will be solutions of problem (1). The value of the penalty parameter c^* is directly proportional to the Lipschitz constant of the function f on the set

$$\mathcal{L}(x^{**}) = \{ x \in \mathbf{R}^n : \; \varphi(x) \leq \varphi(x^{**}) \} ,$$

where

$$x^{**} = \min_{x\in\mathbf{R}^n} f(x) ,$$

and inversely proportional to the number $\beta(x^*)$, where the point x^* is a limit point of the sequence $\{x(c_k)\}$. In this method the regularity condition 1 is used only in a neighborhood of the point x^*.

Note that the function φ is essentially nondifferentiable at the boundary points of the set X.

For differentiable functions at the boundary points of the set X, the regularity condition 1 does not hold. For this reason, if the function φ at the boundary points of the set X is superdifferentiable, then it cannot be used for constructing a sequence of exact penalty functions $F(c, x)$.

An example of a function, which can be used for constructing a family of exact penalty functions (even not requiring the set X to be bounded) is the Euclidean distance function. For it as an exact penalty parameter it is possible to take the Lipschitz constant of the function f on the set $\mathcal{L}(x^{**})$. However this function is not suitable for practical use due to computational problems.

We shall notice, that the regularity condition 1 is not constructive. Sometimes instead of it one uses another regularity condition.

Regularity Condition 2

(See [3].)

We shall assume that there exists a real number $\beta > 0$, such that the following inequality holds

$$\inf_{x\in\mathcal{A}(X)} \min_{\substack{\|g\|=1,\\ g\in N(X,x)}} \varphi'(x,g) = \inf_{x\in\mathcal{A}(X)} \min_{\substack{\|g\|=1,\\ g\in N(X,x)}} \left[\max_{v\in\underline{\partial}\varphi(x)} \langle v, g\rangle + \min_{w\in\overline{\partial}\varphi(x)} \langle w, g\rangle\right] \geq \beta\,. \tag{4}$$

If the set X is bounded and does not consist of isolated points, and the regularity condition 2 is fulfilled for the family of penalty functions $\{F(c_k, \mathrm{x})\}$ then there exists an exact penalty parameter.

Under some assumptions on the set X it is possible to get an analytical representation of the normal cone for this set at each boundary point and then, having calculated the constant β, it is possible to evaluate the exact penalty parameter c^*.

We shall assume, that for the function φ at each point $x \in \mathrm{bd}\, X$ the regularity condition 2 holds. Then the representation of the normal cone $N(X, x)$ to the given set X at the point $x \in \mathrm{bd}\, X$,

$$N(X,x) = \bigcap_{w\in\overline{\partial}\varphi(x)} \mathrm{cl\,cone}(\underline{\partial}\varphi(x) + w)\,,$$

holds. Here, $\mathrm{cl}\, A$ is the closure of A, $\mathrm{cone}\, A$ is the conical hull of A.

For example, if the function φ is subdifferentiable at each boundary point of the set X, then

$$N(X,x) = \mathrm{cl\,cone}(\underline{\partial}\varphi(x))\,,$$

and (4) can be rewritten as

$$\inf_{x\in\mathcal{A}(X)} \min_{\substack{\|g\|=1,\\ g\in\mathrm{cl\,cone}(\underline{\partial}\varphi(x))}} \max_{v\in\underline{\partial}\varphi(x)} \langle v, g\rangle \geq \beta > 0\,. \tag{5}$$

Example 1 Let

$$f_0(x) = \frac{1}{2}\langle A_0 x, x\rangle + \langle b_0, x\rangle\,, \quad x, b_0 \in \mathbf{R}^n\,,$$
$$f_i(x) = \frac{1}{2}\langle A_i x, x\rangle + \langle b_i, x\rangle + c_i\,,$$
$$x, b_i \in \mathbf{R}^n, \quad c_i \in \mathbf{R}^1, \quad i \in I = 1, \ldots, p\,,$$

where f_i, $i \in (0, \ldots, p)$ are strongly convex functions. All the matrices A_i, $i = 0, \ldots, p$, are positive definite.

Consider the problem

$$\min_{x\in X} f_0(x)\,, \tag{6}$$

where

$$X = \{x \in \mathbf{R}^n :\ f_i(x) \leq 0,\ i \in I\}\,.$$

Let

$$\varphi(x) = \max\{0, \varphi_1(x)\}\,,$$
$$\varphi_1(x) = \max_{i\in I} f_i(x)\,,$$
$$x \in \mathbf{R}^n\,.$$

Then

$$X = \{x \in \mathbf{R}^n :\ \varphi(x) = 0\}\,.$$

Let

$$\overline{x} = \arg\min_{x\in\mathbf{R}^n} \varphi_1(x)\,.$$

We shall assume that $\varphi_1(\overline{x}) < 0$.

Problem (6) is a convex programming problem. For this problem the regularity condition 1 is valid.

Let d be the radius of the maximal ball centered at the point $\overline{x}$ and inscribed into the set X. Then the number $c^* = L/2\ md$ is an exact penalty parameter for the

problem (6). In this equality L is a Lipschitz constant of the function f_0 on the set

$$X_1 = \{x \in \mathbf{R}^n : \varphi(x) \le \varphi(x^{**})\} ,$$
$$x^{**} = \arg\min_{x \in \mathbf{R}^n} f_0(x) ,$$

i. e. the number

$$L = \max_{x \in X_1} \|A_0 x + b_0\|$$

can be taken as a Lipschitz constant of f on X_1, m is a strong convexity constant of the function φ_1 $m = \min_{i \in I} \overline{m}_i$, where $\overline{m}_i$ is constant of strong convexity of the function f_i, $i \in I$).

Example 2 Consider the optimization problem

$$\min_{x \in X} f(x) , \tag{7}$$

where

$$f(x) = f_1(x) - f_2(x) ,$$
$$X = \{x \in \mathbf{R}^n : \varphi_1(x) - \varphi_2(x) \le 0\} ,$$

$f_1, f_2, \varphi_1, \varphi_2$ are convex functions defined on $\mathbf{R}^n$.

Let the set X be bounded. It can be defined in the form

$$X = \{x \in \mathbf{R}^n : \varphi(x) = 0\} ,$$

where $\varphi(x) = \max\{0, \varphi_1(x) - \varphi_2(x)\}$.

The function φ can be represented as the difference of convex (d.c.) functions

$$\varphi(x) = \max\{\varphi_1(x), \varphi_2(x)\} - \varphi_2(x) .$$

We shall consider only points $x \in X$ where $\varphi_1(x) = \varphi_2(x)$. Then the pair of convex sets

$$\mathcal{D}\varphi(x) = \left[\operatorname{co}\{\partial\varphi_1(x), \partial\varphi_2(x)\}, -\partial\varphi_2(x)\right]$$

is a quasidifferential of the function φ at a point x, where $\partial\varphi_i(x)$, $i = 1, 2$, is the subdifferential of the convex function φ_i at x in the sense of convex analysis. Here co A is the convex hull of A.

If the regularity condition 2 is valid for the function φ, i. e. there exists a real value $\beta > 0$ such that

$$\inf_{x \in \mathcal{A}(X)} \min_{\substack{\|g\| = 1 \\ g \in N(X, x)}} \left\{ \max_{v \in \operatorname{co}\{\partial\varphi_1(x), \partial\varphi_2(x)\}} \langle v, g\rangle - \max_{w \in \partial\varphi_2(x)} \langle w, g\rangle \right\} \ge \beta ,$$

then there exists an exact penalty parameter c^* for the sequence of penalty functions

$$F(c_k, x) = f(x) + c_k\varphi(x) = \left(f_1(x) + c_k \max\{\varphi_1(x), \varphi_2(x)\}\right) - \left(f_2(x) + c_k\varphi_2(x)\right) .$$

Let the set X be defined as

$$X = \{x \in \mathbf{R}^n : \varphi_1(x) - \varphi_2(x) = 0\} ;$$

then it can be rewritten as

$$X = \{x \in \mathbf{R}^n : \varphi(x) = 0\} ,$$

where $\varphi(x) = \max\{0, |;\varphi_1(x) - \varphi_2(x)|\}$.

In this case the function φ can be represented as the difference of convex functions, namely

$$\varphi(x) = \max\{2\varphi_1(x), 2\varphi_2(x), \varphi_1(x) + \varphi_2(x)\} - (\varphi_1(x) + \varphi_2(x)) .$$

If the regularity condition 2 is valid for the function φ, i. e. there exists a real value $\beta > 0$ such that

$$\inf_{x \in \mathcal{A}(X)} \min_{\substack{\|g\| = 1 \\ g \in N(X, x)}} \left\{ \max_{v \in \operatorname{co}\{2\partial\varphi_1(x), 2\partial\varphi_2(x), \partial\varphi_1(x) + \varphi_2(x)\}} \langle v, g\rangle - \max_{w \in \partial[\varphi_1(x) + \varphi_2(x)]} \langle w, g\rangle \right\} \ge \beta,$$

then there exists an exact penalty parameter c^* for the sequence of penalty functions

$$F(c_k, x) = f(x) + c_k\varphi(x) = f_1(x) + c_k \max\{2\varphi_1(x), 2\varphi_2(x), \varphi_1(x) + \varphi_2(x)\} - \left(f_2(x) + c_k\varphi_1(x) + c_k\varphi_2(x)\right) .$$

Thus the solution of the problem (7) can be obtained as the result of unconstrained optimization of d.c.functions.

V.F. Demyanov [2] considers the following condition for constructing a family of exact penalty functions. Put

$$d(x) = \min_{\|g\| = 1} \varphi'(x, g) = \min_{\|g\| = 1} \left[\max_{v \in \underline{\partial}\varphi(x)} \langle v, g\rangle + \min_{w \in \overline{\partial}\varphi(x)} \langle w, g\rangle \right] ,$$
$$\Psi_X(x) = \limsup_{x' \to x, x' \notin X} d(x') .$$

Regularity Condition 3

If for some $\varepsilon > 0$ the set

$$X_\varepsilon = \{x \in \mathbf{R}^n : \ \varphi(x) \le \varepsilon\}$$

is bounded and

$$\Psi_X(x) < 0\,, \quad \forall x \in \mathrm{bd}\, X\,,$$

then for the family of penalty functions $F(c_k, x)$ there exists an exact penalty parameter $c^* < \infty$.

To use the regularity condition 3 it is necessary to know the behavior of the function φ in the neighborhood of the set X.

Sometimes the following regularity condition is used (see [7]).

Regularity Condition 4

(Condition of *ρ-regularity*). We say that the problem (1) with the set X is *ρ-regular* if there exists a positive number β such that the inequality

$$\varphi(x) \ge \beta \rho_X(x)\,, \quad \forall x \in \mathbf{R}^n \backslash X\,,$$

holds, where ρ_X is the Euclidean distance function.

It is not difficult to observe that the regularity condition 4 is not constructive. In [7] the existence of an exact penalty parameter for a family of penalty functions is proved for problems of nonlinear programming if the condition of ρ-regularity is satisfied.

See also

- ▶ Generalized Monotonicity: Applications to Variational Inequalities and Equilibrium Problems
- ▶ Hemivariational Inequalities: Applications in Mechanics
- ▶ Hemivariational Inequalities: Eigenvalue Problems
- ▶ Hemivariational Inequalities: Static Problems
- ▶ Nonconvex Energy Functions: Hemivariational Inequalities
- ▶ Nonconvex-Nonsmooth Calculus of Variations
- ▶ Quasidifferentiable Optimization
- ▶ Quasidifferentiable Optimization: Algorithms for Hypodifferentiable Functions
- ▶ Quasidifferentiable Optimization: Algorithms for QD Functions
- ▶ Quasidifferentiable Optimization: Applications
- ▶ Quasidifferentiable Optimization: Applications to Thermoelasticity
- ▶ Quasidifferentiable Optimization: Calculus of Quasidifferentials
- ▶ Quasidifferentiable Optimization: Codifferentiable Functions
- ▶ Quasidifferentiable Optimization: Dini Derivatives, Clarke Derivatives
- ▶ Quasidifferentiable Optimization: Optimality Conditions
- ▶ Quasidifferentiable Optimization: Stability of Dynamic Systems
- ▶ Quasidifferentiable Optimization: Variational Formulations
- ▶ Quasivariational Inequalities
- ▶ Sensitivity Analysis of Variational Inequality Problems
- ▶ Solving Hemivariational Inequalities by Nonsmooth Optimization Methods
- ▶ Variational Inequalities
- ▶ Variational Inequalities: F. E. Approach
- ▶ Variational Inequalities: Geometric Interpretation, Existence and Uniqueness
- ▶ Variational Inequalities: Projected Dynamical System
- ▶ Variational Principles

References

1. Charalambous C (1978) A lower bound for the controlling parameters of the exact penalty functions. Math Program 15:278–290
2. Demyanov VF, Rubinov AM (1995) Constructive nonsmooth analysis. P. Lang, Frankfurt am Main
3. Demyanov VF, Stavroulakis GE, Polyakova LN, Panagiotopoulos PD (1996) Quasidifferentiability and nonsmooth modelling in mechanics, engineering and economics. Kluwer, Dordrecht
4. Di Pillo G, Facchinei F (1992) Regularity conditions and exact penalty functions in Lipschitz programming problems. Nonsmooth optimization metods and applications (Amsterdam, 1992). In: Giannessi F (ed). Gordon and Breach, New York, pp 107–120
5. Eremin II (1967) Method of 'penalties' in convex programming. Dokl USSR Acad Sci 173(4):748–751
6. Evtushenko Y, Zhadan V (1990) Exact auxiliary functions. Informatica 1(1):40–55
7. Fedorov VV (1979) Numerical methods of a maximin. Nauka, Moscow

8. Han S, Mangasarian O (1979) Exact penalty functions in nonlinear programming. Math Program 17:251–269
9. Pietrzykowski T (1969) An exact potential method for constrained maxima. SIAM J Numer Anal 16:299–304
10. Zangwill W (1967) Non-linear programming via penalty functions. Managem Sci 13(5):344–358

Quasidifferentiable Optimization: Optimality Conditions

VLADIMIR F. DEMYANOV
St. Petersburg State University, St. Petersburg, Russia

MSC2000: 90Cxx

Article Outline

Keywords

Quasidifferential; Quasidifferential calculus; Necessary and sufficient conditions; Directional derivative

Directionally Differentiable Functions

Let f be a real-valued function defined on an open set $X \subset \mathbf{R}^n$, $x \in X$. The function f is called *Dini differentiable* at the point x in a direction $g \in \mathbf{R}^n$ if there exists the finite limit

$$f'_{\mathrm{D}}(x,g) = \lim_{\alpha \downarrow 0} \frac{1}{\alpha}[f(x+\alpha g) - f(x)]\,. \tag{1}$$

Here $\alpha \downarrow 0$ means that $\alpha \to 0, \alpha > 0$. The quantity $f'_{\mathrm{D}}(x, g)$ is called the *Dini derivative* of f at x in a direction g.

The function f is called *Hadamard differentiable* at the point x in a direction $g \in \mathbf{R}^n$ if there exists the finite limit

$$f'_{\mathrm{H}}(x,g) = \lim_{[\alpha,g'] \to [+0,g]} \frac{1}{\alpha}[f(x+\alpha g') - f(x)]\,. \tag{2}$$

Clearly, if f is Hadamard differentiable at x in a direction g then it is Dini differentiable as well and

$$f'_{\mathrm{H}}(x,g) = f'_{\mathrm{D}}(x,g)\,. \tag{3}$$

If the limit in (1) exists and is finite for every $g \in \mathbf{R}^n$ then the function f is called *Dini directionally differentiable* (D-d.d) at x. The quantity $f'_{\mathrm{D}}(x, g)$ is called the *Hadamard derivative* of f at x in a direction g.

If the limit in (2) exists and is finite for every $g \in \mathbf{R}^n$ then the function f is called the *Hadamard directionally differentiable* (H-d.d) at x. Of course, every H-d.d. function at x is D-d.d., the converse is not necessarily true.

The directional (and generalized directional) derivatives may be used to describe optimality conditions (see ▶ Dini and Hadamard derivatives in optimization). However, using properties of special classes of functions one can expect to get more 'constructive' conditions. One of such classes is the family of quasidifferentiable functions.

Quasidifferentiable Functions

Let f be a real-valued function defined on an open set $X \subset \mathbf{R}^n$, $x \in X$. The function f is called *Dini* (*Hadamard*) *quasidifferentiable* (q.d) at x if it is Dini (Hadamard) directionally differentiable at x and if its directional derivative $f'_{\mathrm{D}}(x, g)$ ($f'_{\mathrm{H}}(x, g)$) can be represented in the form

$$f'_{\mathrm{D}}(x,g) = \max_{v \in \underline{\partial} f_{\mathrm{D}}(x)} (v,g) + \min_{w \in \overline{\partial} f_{\mathrm{D}}(x)} (w,g),$$

$$\left(f'_{\mathrm{H}}(x,g) = \max_{v \in \underline{\partial} f_{\mathrm{H}}(x)} (v,g) + \min_{w \in \overline{\partial} f_{\mathrm{H}}(x)} (w,g) \right),$$

where the sets $\underline{\partial} f_{\mathrm{D}}(x), \overline{\partial} f_{\mathrm{D}}(x), \underline{\partial} f_{\mathrm{H}}(x), \overline{\partial} f_{\mathrm{H}}(x)$ are convex compact sets of $\mathbf{R}^n$. The pair

$$\mathrm{D} f_{\mathrm{D}}(x) = [\underline{\partial} f_{\mathrm{D}}(x), \overline{\partial} f_{\mathrm{D}}(x)]\,,$$

$$\left(\mathrm{D} f_{\mathrm{H}}(x) = [\underline{\partial} f_{\mathrm{H}}(x), \overline{\partial} f_{\mathrm{H}}(x)] \right)$$

is called a *Dini* (*Hadamard*) *quasidifferential* of f at x. Most of the results stated below are valid for both Dini

and Hadamard q.d. functions, therefore we shall use the notation $Df(x) = [\underline{\partial} f(x), \overline{\partial} f(x)$ for both $D_D f(x)$ and $D_H f(x)$ and the pair $Df(x)$ will just be called a *quasidifferential* of f at x. Analogously, the notation $f'(x, g)$ is used for both $f'_D(x, g)$ and $f'_H(x, g)$.

The directional derivative $f'(x, g)$ is positively homogeneous (in g) of degree one:

$$f'(x, \lambda g) = \lambda f'(x, g)\,, \quad \forall \lambda > 0\,. \tag{4}$$

Note that Hadamard quasidifferentiability implies Dini quasidifferentiability, the converse not necessarily being true.

Thus for a quasidifferentiable (q.d) function

$$f'(x, g) = \max_{v \in \underline{\partial} f(x)} (v, g) + \min_{w \in \overline{\partial} f(x)} (w, g)\,, \quad \forall g \in \mathbf{R}^n. \tag{5}$$

The set $\underline{\partial} f(x)$ is called a *subdifferential* of f at x, and the set $\overline{\partial} f(x)$ is called a *superdifferential* of f at x. Note that a quasidifferential is not uniquely defined: If a pair $D = [A, B]$ is a quasidifferential of f at x then, e. g., for any convex compact set $C \subset \mathbf{R}^n$ the pair $D_1 = [A + C, B - C]$ is a quasidifferential of f at x (since, by (5), both these pairs produce the same function $f'(x, g)$). The equivalence class of pairs of convex compact sets $Df(x) = [\underline{\partial} f(x), \overline{\partial} f(x)$ producing the function $f'(x, g)$ by formula (5) is called the *quasidifferential* of f at x (we shall use the same notation $Df(x)$ for the quasidifferential of f at x).

If there exists a quasidifferential $Df(x)$ of the form $Df(x) = [\underline{\partial} f(x), \{0_n\}]$ then f is called *subdifferentiable* at x. If there exists a quasidifferential $Df(x)$ of the form $Df(x) = [\{0_n\}, \overline{\partial} f(x)]$ then f is called *superdifferentiable* at x. Here $0_n = (0, \ldots, 0) \in \mathbf{R}^n$.

Examples of q.d. Functions

1) If f is a smooth function on X then

$$f'(x, g) = (f'(x), g)\,, \tag{6}$$

where $f'(x)$ is the gradient of f at x. It is clear that

$$f'(x, g) = \max_{v \in \underline{\partial} f(x)} (v, g) + \min_{w \in \overline{\partial} f(x)} (w, g)\,, \tag{7}$$

with

$$\underline{\partial} f(x) = \{f'(x)\}\,, \quad \overline{\partial} f(x) = \{0_n\}\,.$$

Hence, f is Hadamard quasidifferentiable and even subdifferentiable. Since in (7) one can also take

$$\underline{\partial} f(x) = \{0_n\}\,, \quad \overline{\partial} f(x) = \{f'(x)\}\,,$$

then f is superdifferentiable as well.

2) If f is a convex function on a convex open set $X \subset \mathbf{R}^n$ then (as it is known from convex analysis) f is H-d.d. on X and

$$f'(x, g) = \max_{v \in \partial f(x)} (v, g)\,,$$

where $\partial f(x)$ is the subdifferential of f (in the sense of convex analysis):

$$\partial f(x) = \left\{ v \in \mathbf{R}^n : \begin{array}{c} f(z) - f(x) \geq (v, z - x) \\ \forall z \in X \end{array} \right\}.$$

Therefore f is Hadamard quasidifferentiable and one can take the pair $Df(x) = [\,\partial f(x), \{0_n\}]$ as its quasidifferential. Thus, f is even subdifferentiable.

3) If f is concave on a convex set X then f is H-d.d. and

$$f'(x, g) = \min_{w \in \overline{\partial} f(x)} (w, g),$$

where

$$\overline{\partial} f(x) = \left\{ w \in \mathbf{R}^n : \begin{array}{c} f(z) - f(x) \leq (w, z - x) \\ \forall z \in X \end{array} \right\}.$$

Hence, f is Hadamard quasidifferentiable and one can take the pair

$$Df(x) = [\{0_n\}, \quad \overline{\partial} f(x)]$$

as its quasidifferential. Thus, f is even superdifferentiable.

Calculus of Quasidifferentials

The family of q.d. functions enjoys a well-developed calculus: First let us define the operation of addition of two pairs of compact convex sets and the operation of multiplication of a pair by a real number.

If $D_1 = [A_1, B_1]$, $D_2 = [A_2, B_2]$ are pairs of convex compact sets in $\mathbf{R}^n$ then

$$D_1 + D_2 = [A, B]$$

with

$$A = A_1 + A_2\,, \quad B = B_1 + B_2\,.$$

If $D = [A, B]$ where A and B are convex compact sets, $\lambda \in \mathbf{R}$ then

$$\lambda D = \begin{cases} [\lambda A, \lambda B], & \lambda \geq 0, \\ [\lambda B, \lambda A], & \lambda < 0. \end{cases}$$

Let $X \subset \mathbf{R}^n$ be an open set.

Proposition 1 ([1, Chap. III])

1) If functions $f_1, \ldots, f_N$ are quasidifferentiable at a point $x \in X$, and $\lambda_1, \ldots, \lambda_N$ are real numbers then the function

$$f = \sum_{i=1}^{N} \lambda_i f_i$$

is also quasidifferentiable at x with a quasidifferential $Df(x) = [\underline{\partial} f(x), \overline{\partial} f(x)]$ where

$$Df(x) = \sum_{i=1}^{N} \lambda_i Df_i(x), \tag{8}$$

$Df_i(x)$ being a quasidifferential of f_i at x.

2) If f_1 and f_2 are quasidifferentiable at a point $x \in X$ then the function $f = f_1 \cdot f_2$ is also q.d. at x and

$$Df(x) = f_1(x)Df_2(x) + f_2(x)Df_1(x). \tag{9}$$

3) If f_1 and f_2 are quasidifferentiable at a point $x \in X$ and $f_2(x) \neq 0$ then the function $f = \{f_1 f_2\}$ is also q.d. at x and

$$Df(x) = \frac{1}{f_2^2(x)} \left[f_2(x)Df_1(x) - f_1(x)Df_2(x) \right]. \tag{10}$$

4) Let functions $f_1, \ldots, f_N$ be quasidifferentiable at a point $x \in X$. Construct the functions

$$\varphi_1(x) = \max_{i \in 1,\ldots,N} f_i(x),$$
$$\varphi_2(x) = \min_{i \in 1,\ldots,N} f_i(x).$$

Then the functions φ_1 and φ_2 are q.d. at x and

$$D\varphi_1(x) = [\underline{\partial}\varphi_1(x), \overline{\partial}\varphi_1(x)],$$
$$D\varphi_2(x) = [\underline{\partial}\varphi_2(x), \overline{\partial}\varphi_2(x)], \tag{11}$$

where

$$\underline{\partial}\varphi_1(x) = \operatorname{co}\left\{ \underline{\partial} f_k(x) - \sum_{\substack{i \in R(x) \\ i \neq k}} \overline{\partial} f_i(x) \colon\ k \in R(x) \right\},$$

$$\overline{\partial}\varphi_1(x) = \sum_{k \in R(x)} \overline{\partial} f_k,$$

$$\underline{\partial}\varphi_2(x) = \sum_{k \in Q(x)} \underline{\partial} f_k,$$

$$\overline{\partial}\varphi_2(x) = \operatorname{co}\left\{ \overline{\partial} f_k(x) - \sum_{\substack{i \in Q(x) \\ i \neq k}} \underline{\partial} f_i(x) \colon\ k \in Q(x) \right\}.$$

Here $[\underline{\partial} f_k(x), \overline{\partial} f_k(x)]$ is a quasidifferential of the function f_k at the point x,

$$R(x) = \{i \in 1, \ldots, N \colon\ f_i(x) = \varphi_1(x)\},$$
$$Q(x) = \{i \in 1, \ldots, N \colon\ f_i(x) = \varphi_2(x)\}.$$

The following composition theorem holds.

Proposition 2 [1, Chap. III]) *Let X be an open set in $\mathbf{R}^n$, Y be an open set in $\mathbf{R}^m$ and let a mapping $H(x) = (h_1(x), \ldots, h_m(x))$ be defined on X, take its values in Y and its coordinate functions h_i be quasidifferentiable at a point $x_0 \in X$. Assume also that a function f is defined on Y and is Hadamard quasidifferentiable at the point $y_0 = H(x_0)$. Then the function*

$$\varphi(x) = f(H(x))$$

is quasidifferentiable at the point x_0.

The corresponding formula for the quasidifferential of φ at x_0 is presented in [Thm. III.2.3]

Remark 3 Thus, the family of quasidifferentiable functions is a linear space closed with respect to all 'smooth' operations and, what is most important, the operations of taking the pointwise maximum and minimum. Formulas (8)–(10) are just generalizations of the rules of classical differential calculus. Most problems and results of classical differential calculus may be formulated for nonsmooth functions in terms of quasidifferentials (see, e. g., [1,3]). For example, a mean value theorem is valid [5].

Necessary and Sufficient Conditions for an Unconstrained Optimum

The following results hold due to the properties of directionally differentiable functions.

Let $X \subset \mathbf{R}^n$ be an open set, f be a real-valued function defined and directionally differentiable on X.

Proposition 4 *For a point $x* \in X$ to be a local or global minimizer of f on X it is necessary that*

$$f'(x^*, g) \geq 0\,, \quad \forall g \in \mathbf{R}^n\,. \tag{12}$$

If f is Hadamard d.d. at $x*$ and

$$f'_H(x^*, g) > 0\,, \quad \forall g \in \mathbf{R}^n\,,\ g \neq 0_n\,, \tag{13}$$

then $x*$ is a strict local minimizer of f.

For a point $x** \in X$ to be a local or global maximizer of f on X it is necessary that

$$f'(x^{**}, g) \leq 0\,, \quad \forall g \in \mathbf{R}^n\,. \tag{14}$$

If f is Hadamard d.d. at $x**$ and

$$f'_H(x^{**}, g) < 0\,, \quad \forall g \in \mathbf{R}^n\,,\ g \neq 0_n\,, \tag{15}$$

then $x**$ is a strict local maximizer of f.

These conditions may be restated in terms of quasidifferentials. Let f be quasidifferentiable on an open set $X \subset \mathbf{R}^n$.

Proposition 5 (see [1,3,5]) *For a point $x* \in X$ to be a local or global minimizer of f on X it is necessary that*

$$-\overline{\partial} f(x^*) \subset \underline{\partial} f(x^*)\,. \tag{16}$$

If f is Hadamard quasidifferentiable at $x*$ and

$$-\overline{\partial} f(x^*) \subset \operatorname{int} \underline{\partial} f(x^*), \tag{17}$$

then $x*$ is a strict local minimizer of f.

For a point $x^{**} \in X$ to be a local or global maximizer of f on X it is necessary that

$$-\underline{\partial} f(x^{**}) \subset \overline{\partial} f(x^{**})\,. \tag{18}$$

If f is Hadamard quasidifferentiable at x^{**} and

$$-\underline{\partial} f(x^{**}) \subset \operatorname{int} \overline{\partial} f(x^{**})\,, \tag{19}$$

then x^{**} is a strict local maximizer of f.

Remark 6 The quasidifferential represents a generalization of the notion of gradient to the nonsmooth case and therefore conditions (16)–(19) are first order optimality conditions.

In the smooth case one can take $\mathrm{D}f(x) = [\underline{\partial} f(x), \overline{\partial} f(x)]$ where $\underline{\partial} f(x) = \{f'(x)\}$, $\overline{\partial} f(x) = \{0_n\}$; therefore condition (16) is equivalent to

$$f'(x^*) = 0_n\,, \tag{20}$$

condition (18) is equivalent to

$$f'(x^{**}) = 0_n\,, \tag{21}$$

and, since both sets $\underline{\partial} f$ and $\overline{\partial} f$ are singletons, the conditions (17) and (19) are impossible. Thus, conditions (17) and (19) are essentially 'nonsmooth'.

A point $x^* \in X$ satisfying (16) is called an inf-stationary point, a point $x^{**} \in X$ satisfying (18) is called a sup-stationary point of f. In the smooth case the necessary condition for a minimum (20) is the same as the necessary condition for a maximum (21).

Directions of Steepest Descent and Ascent

Let $x \in X$ be not an inf-stationary point of f (i. e. condition (16) is not satisfied). Take $w \in \overline{\partial} f(x)$ and find

$$\min_{v \in \underline{\partial} f(x)} \|v + w\| = \|v(w) + w\| = \rho_1(w).$$

Since $\underline{\partial} f(x)$ is a convex compact set, the point $v(w)$ is unique. Find now

$$\max_{w \in \overline{\partial} f(x)} \rho_1(w) = \rho_1(w(x)).$$

The point $w(x)$ is not necessarily unique. As x is not an inf-stationary point, then $\rho_1(w(x)) > 0$. The direction

$$g_1(x) = -\frac{v(w(x)) + w(x)}{\|v(w(x)) + w(x)\|} = -\frac{v(w(x)) + w(x)}{\rho_1(w(x))} \tag{22}$$

is a *steepest descent direction* of the function f at the point x, i. e.

$$f'(x, g_1(x))) = \min_{\|g\|=1} f'(x, g)\,.$$

Here $\|\cdot\|$ is the Euclidean norm. The quantity $f'(x, g_1(x)) = -\rho_1(w(x))$ is the *rate of steepest descent* of f

at x. It may happen that the set of steepest descent directions is not a singleton (and it need not be convex too). Recall that in the smooth case the steepest descent direction is always unique (if x is not a stationary point).

Similarly, if $x \in X$ is not a sup-stationary point of f (i. e. condition (18) does not hold) then let us take $v \in \partial f(x)$ and find

$$\min_{w \in \overline{\partial} f(x)} \|v + w\| = \|v + w(v)\| = \rho_2(v)$$

and

$$\max_{v \in \underline{\partial} f(x)} \rho_2(v) = \rho_2(v(x)) \,.$$

The direction

$$g_2(x) = \frac{v(x) + w(v(x))}{\|v(x) + w(v(x))\|} = \frac{v(x) + w(v(x))}{\rho_2(v(x))} \tag{23}$$

is a *steepest ascent direction* of the function f at x, i. e.

$$f'(x, g_2(x))) = \max_{\|g\|=1} f'(x, g) \,.$$

The quantity $f'(x, g_2(x)) = \rho_2(v(x))$ is the *rate of steepest ascent* of f at x. As above it may happen that there exist many steepest ascent directions.

Remark 7 Thus, the necessary conditions (16) and (18) are 'constructive': in the case where one of these conditions is violated we are able to find steepest descent or ascent directions.

The condition for a minimum (16) can be rewritten in the equivalent form

$$0_n \in \bigcap_{w \in \overline{\partial} f(x^*)} [\underline{\partial} f(x^*) + w] := L_1(x^*) \,, \tag{24}$$

and the condition for a maximum (18) can also be represented in the equivalent form

$$0_n \in \bigcap_{v \in \underline{\partial} f(x^{**})} [\overline{\partial} f(x^{**}) + v] := L_2(x^{**}) \,. \tag{25}$$

However, if, for example, (24) is violated at a point x, we are unable to recover steepest descent directions, it may even happen that the set $L_1(x)$ is empty (see [1, Sects. V.2 and V.3]).

Therefore, condition (24) is not 'constructive': if a point x is not inf-stationary then condition (24) supplies no information about the behavior of the function in a neighborhood of x and we are unable to get a 'better' point (e. g., to decrease the value of the function). The same is true for the condition for a maximum (25). Nevertheless conditions (25) and (25) may be useful for some other purposes.

Example 8 Let $x = (x^{(1)}, x^{(2)}) \in \mathbf{R}^2$, $x_0 = (0, 0)$, $f(x) = |x^{(1)}| - |x^{(2)}|$. We have $f(x) = f_1(x) - f_2(x)$, where $f_1(x) = \max\{f_3(x), f_4(x)\}$, $f_2(x) = \{f_5(x), f_6(x)\}$, $f_3(x) = x^{(1)}$, $f_4(x) = -x^{(1)}$, $f_5(x) = x^{(2)}$, $f_6(x) = -x^{(2)}$. The functions f_3–f_6 are smooth therefore (see (7))

$$\begin{aligned}
&\mathrm{D}f_3(x) = [\underline{\partial} f_3(x), \overline{\partial} f_3(x)] \,, \\
&\text{with } \underline{\partial} f_3(x) = \{(1, 0)\},\ \overline{\partial} f_3(x) = \{(0, 0)\} \,, \\
&\mathrm{D}f_4(x) = [\underline{\partial} f_4(x), \overline{\partial} f_4(x)] \,, \\
&\text{with } \underline{\partial} f_4(x) = \{(-1, 0)\},\ \overline{\partial} f_4(x) = \{(0, 0)\} \,, \\
&\mathrm{D}f_5(x) = [\underline{\partial} f_5(x), \overline{\partial} f_5(x)] \,, \\
&\text{with } \underline{\partial} f_5(x) = \{(0, 1)\},\ \overline{\partial} f_5(x) = \{(0, 0)\} \,, \\
&\mathrm{D}f_6(x) = [\underline{\partial} f_6(x), \overline{\partial} f_6(x)] \,, \\
&\text{with } \underline{\partial} f_6(x) = \{(0, -1)\},\ \overline{\partial} f_6(x) = \{(0, 0)\} \,,
\end{aligned}$$

Applying (11) one gets $\mathrm{D}f_1(x_0) = [\underline{\partial} f_1(x_0), \overline{\partial} f_1(x_0)]$, where

$$\begin{aligned}
\underline{\partial} f_1(x_0) &= eratornameco\{\underline{\partial} f_3(x_0) - \overline{\partial} f_4(x_0) \\
\underline{\partial} f_4(x_0) - \overline{\partial} f_3(x_0)\} &= \mathrm{co}\{(1, 0), (-1, 0)\} \,, \\
\overline{\partial} f_1(x_0) &= \{(0, 0)\} \,, \\
\mathrm{D}f_2(x_0) &= [\underline{\partial} f_2(x_0), \overline{\partial} f_2(x_0)] \,,
\end{aligned}$$

where

$$\begin{aligned}
\underline{\partial} f_2(x_0) &= \mathrm{co}\{\underline{\partial} f_5(x_0) - \overline{\partial} f_6(x_0), \underline{\partial} f_6(x_0) - \overline{\partial} f_5(x_0)\} \\
&= \mathrm{co}\{(0, 1), (0, -1)\} \,, \\
\overline{\partial} f_2(x_0) &= \{(0, 0)\} \,.
\end{aligned}$$

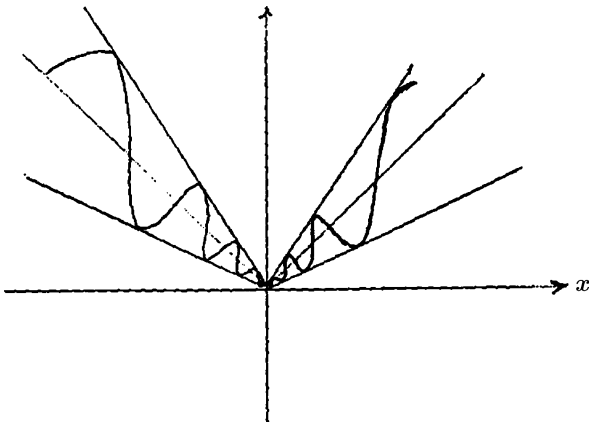
Finally, formula (8) yields

$$\mathrm{D}f(x_0) = [\underline{\partial} f(x_0), \overline{\partial} f(x_0)] \,,$$

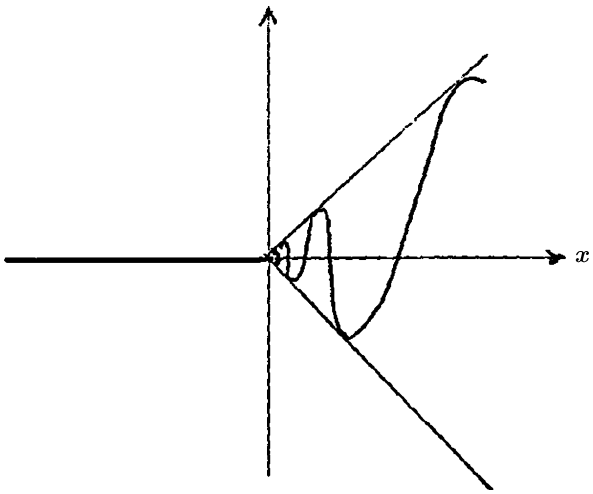
where

$$\begin{aligned}
\underline{\partial} f(x_0) &= \mathrm{co}\{(1, 0), (-1, 0)\} \,, \\
\overline{\partial} f(x_0) &= \mathrm{co}\{(0, 1), (0, -1)\} \,.
\end{aligned}$$

Since (see Fig. 1) conditions (16) and (18) are not satisfied, the point x_0 is neither inf-stationary nor sup-stationary.



Quasidifferentiable Optimization: Optimality Conditions, Figure 1



Quasidifferentiable Optimization: Optimality Conditions, Figure 2

Applying (22) and (23) we conclude that there exist two directions of steepest descent: $g_1 = (0, 1)$, $g_1{}' = (0, -1)$ and two directions of steepest ascent: $g_2 = (1, 0)$, $g_2{}' = (-1, 0)$.

It is also clear that the sets (see (24), (25))

$$L_1(x_0) = \bigcap_{w \in \overline{\partial} f(x_0)} [\underline{\partial} f(x_0) + w]$$

and

$$L_2(x_0) = \bigcap_{v \in \underline{\partial} f(x_0)} [\overline{\partial} f(x_0) + v]$$

are both empty.

Remark 9 If a function f is directionally differentiable but not quasidifferentiable, and if its directional derivative $f'(x, g)$ is continuous as a function of direction (this is the case, e. g., if f is directionally differentiable and Lipschitz) then (by the Stone–Weierstrass theorem) its directional derivative may be approximated (to within any given accuracy) by the difference of two positively homogeneous convex functions, i. e.

$$f'(x, g) \approx \max_{v \in A}(v, g) + \min_{w \in B}(w, g)\,, \tag{26}$$

where A and B are convex compact sets in $\mathbf{R}^n$. Relation (26) shows that f' can be studied by means of quasidifferential calculus (e. g., one is able to find an approximation of a steepest descent direction etc.). Corresponding results can be found in [1,4].

Remark 10 In many cases of practical importance the quasidifferential of a function f is a pair of sets each of them being the convex hull of a finite number of points or/and balls. If this happens it is easy to store and operate with the quasidifferential, to check necessary conditions, to find directions of descent or ascent, to construct numerical methods.

Necessary and Sufficient Conditions for a Constrained Optimum

Let a function f be defined and finite on some open set $X \subset \mathbf{R}^n$ and let $\Omega \subset X$. Consider the problem of finding a minimum or a maximum of f on Ω. For the definiteness in the sequel we shall consider only the problem of minimizing f on Ω since the problem of maximizing f is the problem of minimizing the function $f_1 = -f$.

Let $x \in \Omega$. The set

$$\Gamma(x, \Omega) = \left\{ g \in \mathbf{R}^n : \begin{array}{c} \exists\{[\alpha_k, g_k]\}: \\ [\alpha_k, g_k] \to [+0, g] \\ x + \alpha_k g_k \in \Omega \\ \forall k \end{array} \right\} \tag{27}$$

is called the *Bouligand cone* to the set Ω at the point x (or the *cone of feasible directions*). It is nonempty and closed. If $x \in \operatorname{int} \Omega$ then $\Gamma(x, \Omega) = \mathbf{R}^n$.

Proposition 11 *Let f be Hadamard directionally differentiable at a point $x^* \in \Omega$. For the point x^* to be a local or global minimizer of f on Ω it is necessary that*

$$f'_H(x^*, g) \geq 0\,, \quad \forall g \in \Gamma(x^*, \Omega)\,. \tag{28}$$

If

$$f'_H(x^*, g) > 0\,, \quad \forall g \in \Gamma(x^*, \Omega),\ g \neq 0_n\,, \tag{29}$$

then x^* is a strict local minimizer of f on Ω, i. e. there exists $\delta > 0$ such that

$$f(x) > f(x^*)\,,$$
$$\forall x \in \Omega,\ \|x - x^*\| < \delta,\ x \neq x^*\,.$$

The set $\Omega \subset \mathbf{R}^n$ is called *quasidifferentiable* if it can be represented in the form

$$\Omega = \{x \in \mathbf{R}^n :\ h(x) \leq 0\}\,, \tag{30}$$

where h is a quasidifferentiable function.

Take $x \in \Omega$ and consider the cones

$$\gamma_1(x) = \{g \in \mathbf{R}^n :\ h'(x, g) < 0\}\,,$$
$$\Gamma_1(x) = \{g \in \mathbf{R}^n :\ h'(x, g) \leq 0\}\,.$$

Let $h(x) = 0$. We say that the 'regularity condition is satisfied at x' if

$$\operatorname{cl}\gamma_1(x) = \Gamma_1(x)\,. \tag{31}$$

If $h(x) = 0$ (i. e., by (30), $x \in \Omega$) and the regularity condition (31) holds then

$$\Gamma(x, \Omega) = \Gamma_1(x)\,. \tag{32}$$

Now we are able to express condition (28) and (29) in terms of quasidifferentials of the functions f and h.

If $h(x) < 0$ then $x \in \operatorname{int}\Omega$, $\Gamma(x, \Omega) = \mathbf{R}^n$ and, by Proposition 5, conditions (16) and (17) hold. Therefore let us consider the case $h(x) = 0$.

Proposition 12 *Let functions f and h be Hadamard quasidifferentiable at a point $x^* \in \Omega$ and $h(x^*) = 0$. Assume also that the regularity condition (31) is satisfied at x^*. For the point x^* to be a local or global minimizer of f on Ω it is necessary that*

$$\left(\underline{\partial} f(x^*) + w\right) \bigcap \left[-\operatorname{cl}(\operatorname{cone}(\underline{\partial} h(x^*) + w'))\right] \neq \emptyset \tag{33}$$

for all $w \in \overline{\partial} f(x^*)$, $w' \in \overline{\partial} h(x^*)$.

Condition (33) is equivalent to the condition

$$-\overline{\partial} f(x^*) \subset L(x^*)\,, \tag{34}$$

where

$$L(x) = \bigcap_{w \in \overline{\partial} h(x)} \left[\underline{\partial} f(x) + \operatorname{cl}(\operatorname{cone}(\underline{\partial} h(x) + w))\right]\,.$$

The set $L(x)$ is nonempty and convex.

If $h(x^*) = 0$ and

$$-\overline{\partial} f(x^*) \subset \operatorname{int} L(x^*)\,, \tag{35}$$

then x^* is a strict local minimizer of f on Ω.

A point $x^* \in \Omega$ is called an inf-*stationary point* of f on Ω if condition (28) holds.

Let $x \in \Omega$, $h(x) = 0$. Assume that x is not an inf-stationary point and find

$$\min_{z \in [\underline{\partial} f(x) + w]} \|z + z'\| = \|z(w, w') + z'(w, w')\|$$
$$= \|v(w, w') + w + v'(w, w') + w'\|$$
$$= \|q(w, w')\| = d(w, w')$$

and

$$\rho(x) = \max_{\substack{w \in \overline{\partial} f(x) \\ w' \in \overline{\partial} h(x)}} d(w, w') \tag{36}$$
$$= d(w_0, w'_0) = \|q(w_0, w'_0)\|\,.$$

Since x is not inf-stationary then $\rho(x) > 0$.

Proposition 13 *If $h(x) = 0$ and the regularity condition (31) holds then the direction*

$$g_0 = -\frac{q(w_0, w'_0)}{\rho(x)} \tag{37}$$

is a steepest descent direction of f on Ω at x and $g_0 \in \Gamma(x, \Omega)$,

$$f'(x, g_0) = \min_{\|g\| = 1, g \in \Gamma(x)} f'(x, g)$$
$$= -\|q(w_0, w'_0)\| = -\rho(x)$$

i. e. $-\rho(x)$ is the rate of steepest descent.

Remark 14 If there exist several pairs $[w_0, w'_0](w_0 \in \overline{\partial} f(x), w'_0 \in \overline{\partial} h(x))$ satisfying (36), then (by (37)) there are several steepest descent directions.

Remark 15 Condition (33) is also equivalent to

$$0_n \in \bigcap_{\substack{w \in \overline{\partial} f(x') \\ w' \in \overline{\partial} h(x^*)}} \left[\underline{\partial} f(x^*) + w + \operatorname{cl}(\operatorname{cone}(\underline{\partial} h(x^*) + w'))\right]$$
$$= L'(x^*)\,. \tag{38}$$

However, condition (38) is not 'constructive' since the set $L'(x)$ may happen to be empty if x is not a stationary point (we consider the case $h(x) = 0$).

Proposition 16 *Let $x^* \in \Omega$ and $h(x^*) = 0$. Assume that the functions f and h are quasidifferentiable at x^*. For the point x^* to be a local or global minimizer of f on Ω it is necessary that*

$$L_1(x^*) \subset L_2(x^*)\,, \tag{39}$$

where

$$L_1(x) = -[\overline{\partial} f(x) + \overline{\partial} h(x)]\,,$$
$$L_2(x) = \mathrm{co}\{\underline{\partial} f(x) - \overline{\partial} h(x)\,, \underline{\partial} h(x) - \overline{\partial} f(x)\}.$$

If, in addition, f and h are Hadamard q.d. at x^*, $h(x^*) = 0$ and $L_1(x^*) \subset \mathrm{int}\, L_2(x^*)$ then x^* is a strict local minimizer of f on Ω.

Proposition 17 *Let $h(x^*) = 0$, f and h be Hadamard q.d. at x^*. If the regularity condition (31) holds at x^* then condition (39) is equivalent to condition (28).*

Let $x \in \Omega$, $h(x) = 0$. Assume that (39) does not hold. Find

$$d(x) = \max_{v \in L_1(x)} \rho(v) = \rho(v(x)),$$

where

$$\rho(v) = \min_{w \in L_2(x)} \|v - w\| = \|v - w(v)\|\,.$$

Since (39) is not satisfied then $\rho(v(x)) > 0$.

Proposition 18 *The direction*

$$g_0' = \frac{v(x) - w(v(x))}{\rho(v(x))} \tag{40}$$

is a descent direction of f on Ω at x.

Remark 19 While the steepest descent direction g_0 (see (37)) may be not admissible, the direction $g_0{}'$ (see (40)) is always admissible, i. e. for sufficiently small $\alpha > 0$ we have $x + \alpha\, g_0{}' \in \Omega$.

Recent results and the state-of-the-art in quasidifferential calculus can be found in [2].

See also

- ▶ Generalized Monotonicity: Applications to Variational Inequalities and Equilibrium Problems
- ▶ Hemivariational Inequalities: Applications in Mechanics
- ▶ Hemivariational Inequalities: Eigenvalue Problems
- ▶ Hemivariational Inequalities: Static Problems
- ▶ Nonconvex Energy Functions: Hemivariational Inequalities
- ▶ Nonconvex-nonsmooth Calculus of Variations
- ▶ Quasidifferentiable Optimization
- ▶ Quasidifferentiable Optimization: Algorithms for Hypodifferentiable Functions
- ▶ Quasidifferentiable Optimization: Algorithms for QD Functions
- ▶ Quasidifferentiable Optimization: Applications
- ▶ Quasidifferentiable Optimization: Applications to Thermoelasticity
- ▶ Quasidifferentiable Optimization: Calculus of Quasidifferentials
- ▶ Quasidifferentiable Optimization: Codifferentiable Functions
- ▶ Quasidifferentiable Optimization: Dini Derivatives, Clarke Derivatives
- ▶ Quasidifferentiable Optimization: Exact Penalty Methods
- ▶ Quasidifferentiable Optimization: Stability of Dynamic Systems
- ▶ Quasidifferentiable Optimization: Variational Formulations
- ▶ Quasivariational Inequalities
- ▶ Sensitivity Analysis of Variational Inequality Problems
- ▶ Solving Hemivariational Inequalities by Nonsmooth Optimization Methods
- ▶ Variational Inequalities
- ▶ Variational Inequalities: F. E. Approach
- ▶ Variational Inequalities: Geometric Interpretation, Existence and Uniqueness
- ▶ Variational Inequalities: Projected Dynamical System
- ▶ Variational Principles

References

1. Demyanov VF, Rubinov AM (1995) Constructive nonsmooth analysis. P. Lang, Frankfurt am Main

2. Demyanov VF, Rubinov AM (eds) (2000) Quasidifferentiablity and related topics. Kluwer, Dordrecht
3. Demyanov VF, Vasiliev LV (1986) Nondifferentiable optimization. Springer and Optim. Software, Berlin
4. Gorokhovik VV (1986) ε-quasidifferentiability of real-valued functions and optimality conditions in extremal problems. Math Program Stud 29:203–218
5. Xia ZQ (1987) On mean-value theorems in quasidifferential calculus. J Math Res Exp Dalian, China, DIT 4:681–684

Quasidifferentiable Optimization: Stability of Dynamic Systems

GEORGIOS E. STAVROULAKIS
Carolo Wilhelmina Techn. University, Braunschweig, Germany

MSC2000: 74G60, 74H99, 49J52, 49Q10, 74K99, 74Pxx, 90C90

Article Outline

Keywords

Quasidifferentiability; Nonsmooth mechanics; Elastic stability

Problems in mechanics whose governing relations can be obtained from a generally nondifferentiable and nonconvex, but quasidifferentiable (in the sense of V.F. Demyanov and A.M. Rubinov) potential function are considered. They consider a fairly general form for the modeling and the study of nonsmooth problems in mechanics [4] and they cover certain classes of variational and hemivariational inequality problems of mechanics [14,15]. The notion of hemivariational inequalities has been introduced and thoroughly studied in mechanics by P.D. Panagiotopoulos (see also ► Nonconvex energy functions: Hemivariational inequalities; ► Hemivariational inequalities: Applications in mechanics). Moreover, there exists extensive theoretical support for the use of quasidifferentiable calculus and optimization techniques, see, e. g., [3,4]. For methods and heuristic algorithms of nonconvex optimization in computational mechanics, see [12]. In this short note some techniques for treating stability problems for nonsmooth structures are outlined. This way results for classical, smooth structures (e. g., [1,10,11]) can be extended to cover nonsmooth ones (cf., also [8,9]). This work and the preliminary results outlined here are based on [18,19].

All previously mentioned potentials are piecewise-differentiable and may be described, in general, as continuous selections of differentiable functions. In turn, the structural analysis problem results from minimality or in general critical point conditions of the potential (see examples in [2,6,7,14,15]).

Results from stability analysis of parametric optimization problems for nondifferentiable functions are used for the study of a stepwise holonomic, incremental structural analysis problem. In particular the systematic first and second order linearizations proposed in respectively, and the arising normal forms are adopted for the potential energy function.

The techniques outlined here may be useful both for the analysis of the stability of structures which involve nonmonotone and possibly multivalued nonlinearities (in a holonomic or a stepwise holonomic setting) and for the design of incremental-iterative algorithms for structural analysis purposes.

Smooth Potentials and Stability in Mechanics

Let a discretized elastostatic analysis problem be formulated as a potential energy minimization problem:

$$\min_{u \in U_{\text{ad}}} \left\{ \overline{\Pi}(u, \lambda) = \Pi(e(u)) - p(u, \lambda) \right\} . \tag{1}$$

Here u is the n-vector of displacement degrees of freedom, e is the m-vector of discrete element deformations, $\Pi\ (e) \in \mathbf{R}$ is the internal energy density, $p\ (u, \lambda) \in \mathbf{R}$ is the external loading potential, parametrized by a loading scalar $\lambda \in \mathbf{R}^1$ and U ad $\subseteq \mathbf{R}^n$ is the space of admissible displacements. Displacement u and deformation e vectors are connected by the geometric compatibility operator $A(u)$: $\mathbf{R}^n \to \mathbf{R}^m$ such that $e\ (u) = A\ (u)$ holds.

On the assumption that $\overline{\Pi}(u,\lambda)$ is smooth, the equilibrium configurations for the structural system are critical points of this potential, i. e. for fixed $\lambda = \overline{\lambda}$ one has:

$$\Sigma_{\text{crit}} = \left\{ u \in \mathbf{R}^n \colon \ \nabla_u \overline{\Pi}(u,\overline{\lambda}) = 0 \right\} . \tag{2}$$

Moreover, inspection of the second order derivatives (the Hessian matrix of $\overline{\Pi}(u,\overline{\lambda})$) gives us stability information [1]. If u is a nondegenerate critical point, i. e. $u \in \Sigma_{\text{crit}}$ and $\nabla_u^2 \overline{\Pi}(u,\overline{\lambda})$ is regular, then a positive or negative definite Hessian $\nabla_u^2 \overline{\Pi}(u,\overline{\lambda})$ indicates that the point u is a local minimum or maximum, respectively. Only local minima correspond to stable equilibrium configurations. If $\nabla_u^2 \overline{\Pi}(u,\overline{\lambda})$ is singular in $u \in \Sigma_{\text{crit}}$, then higher order derivatives of $\overline{\Pi}(u,\overline{\lambda})$ must be examined for stability [1].

If u_0 is either a noncritical or a nondegenerate critical point of $\overline{\Pi}(u,\overline{\lambda})$, which is assumed here to be at least a C^2-function, i. e. two times continuously differentiable, then $\overline{\Pi}(u,\overline{\lambda})$ is C^1-equivalent to its second order approximation around u_0, i. e.

$$\begin{aligned}(\overline{\Pi} \circ \Phi)(u) = {} & \overline{\Pi}(u_0,\overline{\lambda}) + \nabla_u \overline{\Pi}(u_0,\overline{\lambda})^{\top}(u-u_0) \\ & + \frac{1}{2}(u-u_0)^{\top} \nabla_u^2 \overline{\Pi}(u_0,\overline{\lambda})(u-u_0) ,\end{aligned} \tag{3}$$

where Φ is a C^1-coordinate transformation (diffeomorphism). In the vicinity of a nondegenerate critical point the behavior of $\overline{\Pi}(u,\overline{\lambda})$ is characterized by the number of negative eigenvalues of $\nabla_u^2 \overline{\Pi}(u,\overline{\lambda})$ (the *quadratic index*).

In the coordinates $\Delta u = u - u_0$ and the notation $\Delta\overline{\Pi}(\Delta u,\overline{\lambda}) = \overline{\Pi}(u,\overline{\lambda}) - \overline{\Pi}(u_0,\overline{\lambda})$ we can determine (cf. [5, p. 21]) a local C^1-coordinate transformation Φ: $U \to V$, where U, V are neighborhoods of the origin, such that:

$$\begin{aligned}\Delta\overline{\Pi} \circ \Phi^{-1}(y) = {} & -y_1^2 - \cdots - y_k^2 + y_{k+1}^2 + \cdots + y_n^2 , \\ & \forall y \in V .\end{aligned} \tag{4}$$

Qualitative stability results for fixed load $\lambda = \overline{\lambda}$ are recognized in the normal form (4).

Incremental Algorithm

Incremental-iterative solution algorithms are based on appropriate approximations of (1). Let us consider the one-parametric load incrementation on the following case of (1) (cf. [10]):

$$\min_{u \in \mathbf{R}^n} \{ \overline{\Pi}(u,\lambda) = \Pi(u) - \lambda p^{\top} u \} . \tag{5}$$

For equilibrium we have

$$\nabla_u \overline{\Pi}(u,\lambda) = 0 \quad \Rightarrow \quad \nabla_u \Pi(u) - \lambda p = 0 \tag{6}$$

For the examination of the stability of a solution we study the following relation in terms of $\lambda - \lambda_0 = \Delta\lambda$, (defining Δu as a function of $\Delta\lambda$, if $\nabla_u^2 \Pi(u_0)$ is regular)

$$\nabla_u^2 \Pi(u_0)\Delta u + \Delta\lambda p = 0 , \tag{7}$$

which connects the incremental displacement Δu for a change of loading equal to $\Delta\lambda p$. Relation (7) can be produced by subtraction of the Taylor expansions of the equilibrium equation (6) in $(u_0 + \Delta u, \lambda_0 + \Delta\lambda)$ and (u_0, λ_0), respectively, and by using the approximation (up to higher order terms)

$$\nabla_u \Pi(u_0 + \Delta u) = \nabla_u \Pi(u_0) + \nabla_u^2 \Pi(u_0)\Delta u . \tag{8}$$

Consider the coordinate transformation: $\Delta u = \Phi^{-1}(y) = \phi_i y_i = F y$ where ϕ_i are the eigenvectors of $\nabla^2 \Pi(u_0)$ and the summation convention over repeated indices is used. Then equation (7) is written in the new coordinate system as:

$$\begin{aligned}& \nabla^2 \Pi(u_0) F y - p\Delta\lambda = 0 \\ & \quad \Rightarrow \quad F^{\top} \nabla^2 \Pi(u_0) F y - F^{\top} p \Delta\lambda = 0 \\ & \quad \Rightarrow \quad \left[\omega_i y_i\right] - F^{\top} p \Delta\lambda = 0 .\end{aligned} \tag{9}$$

Here ω_i are the eigenvalues of the local tangential stiffness matrix $K(u_0) = \nabla^2 \Pi(u_0)$, which act as stability coefficients for the linearized equation of equilibrium (7) [10,11].

Nonsmooth Superpotentials

Let us assume problem (1) with a nonsmooth potential energy function. For simplicity let only the internal energy function $\Pi(u)$ be nonsmooth and $U_{\text{ad}} = \mathbf{R}^n$ in (1).

Let V denote an open subset of $\mathbf{R}^n$. We call a function f: $V \to \mathbf{R}$ a *continuous selection* of the C^r-functions g_i: $V \to \mathbf{R}$, $1 \le i \le k$, (briefly, $f \in \text{CS}\{g_1, \ldots, g_k\}$), if f is continuous and $\forall u \in V\ \exists i \in \{1, \ldots, k\}$: $f(u) = g_i(u)$. Let

$\Pi(u)$ in (1) be a piecewise differentiable PC^r) function of appropriate order $r > 1$, defined on an open set $U \subseteq \mathbf{R}^n$. This means that (cf. [7]) at every point $u_0 \in U$ there exists an open neighborhood $V \subseteq U$ and a finite collection of C^r-functions $\{\Pi_1, \ldots, \Pi_k\}$ defined on V such that $\Pi|_V \in \mathrm{CS}\{\Pi_1, \ldots, \Pi_k\}$.

Let $I(u)$ be the *active index set* set$\{i\colon \Pi(u) = \Pi_k(u)\}$ One considers a smooth external loading potential $p(u, \lambda)$, which depends on the one-dimensional loading parameter λ (cf. (1)).

The assumption of a PC^r-potential energy function is very general and covers a large number of nonsmooth mechanics applications (see, also, [13, Chap. 8]). More detailed analysis of the requirements which are necessary in order for a PC^r-function to be the potential of a certain structural analysis problem must be investigated on a case-by-case basis.

Any PC^r-potential is locally Lipschitz continuous and Bouligand differentiable with the B-derivative at a point $u_0 \in \mathbf{R}^n$ in the direction $d \in \mathbf{R}^n$ being a continuous selection of the functions $\nabla\Pi_i(u_0)^\top d, i \in \widehat{I}(u_0)$. Here $\widehat{I}(u_0)$ denotes the *essentially active index set* $\widehat{I}(u_0) = \{i \in I(u_0)\colon\ u_0 \in \mathrm{cl}(\mathrm{int}(\{u \in U\colon \Pi(u) = \Pi_i(u)\}))\}$, with cl (resp. int) abbreviating the closure (resp. the interior) of a set. For completeness, recall that *Clarke's generalized subdifferential* is given by [7] $\partial_{\mathrm{Cl}}\Pi(u_0) = \mathrm{conv}\left\{\nabla\Pi_i(u_0)\colon\ i \in \widehat{I}(u_0)\right\}$ where conv stands for the convex hull.

Nonsmooth Local Approximations

For the needs of the applications in mechanics the first and the second order differentiation, or the appropriate analogous nonsmooth notions, and suitable local nonsmooth approximations which generalize the (second order) Taylor expansion of a smooth function are used. A local coordinate transformation will provide us with a simple formulation of the energy minimization problem, cf. (4), which, in turn, will be used for the extraction of stability information analogous to (9).

Following [6], a critical point u_0 of a PC^2-potential function $\overline{\Pi}(u)$ is called a *nondegenerate critical point* if $\overline{\Pi}$ is locally representable as a continuous selection of functions $\overline{\Pi}_1, \ldots, \overline{\Pi}_k$ such that the following properties are true:

ND1) the vectors $\nabla\overline{\Pi}_j(u_0), j \in I(u_O)\backslash\{i\}$ are linearly independent $\forall \mathrm{i} \in I(u_0)$,

ND2) the restricted Hessian of the Lagrangian, the matrix $\nabla^2 L(u_0)\,|_{V(u_0)}$, is invertible.

Here $V(u\}_0)$ denotes the space

$$\left\{ y \in \mathbf{R}^n\colon \begin{array}{c} [\nabla\overline{\Pi}_i(u_0) - \nabla\overline{\Pi}_j(u_0)]^\top y = 0, \\ i, j \in I(u_0) \end{array} \right\}.$$

For the Lagrangian

$$L(u) = \sum_{i \in I(u_0)} \lambda_i \overline{\Pi}_i(u)$$

holds

$$\begin{aligned} &\sum_{i \in I(u_0)} \lambda_i \nabla\overline{\Pi}_i(u_0) = 0\,, \\ &\sum_{i \in I(u_0)} \lambda_i = 1\,, \quad \lambda_i \geq 0\,. \end{aligned} \tag{10}$$

The qualitative behavior of the potential energy function, the link to the stability of the described mechanical system, can be shown if one considers the normal form (cf. (4))). In this context, the following result of [6] is of importance.

Let $\overline{\Pi} \in \mathrm{CS}(\overline{\Pi}_i, i \in I)$ and let $u_0 \in \mathbf{R}^n$ be a nondegenerate critical point for $\overline{\Pi}$ with quadratic index equal to q. Suppose moreover that $|I_0(u_0)| = k + 1$. Then at u_0, the potential is topologically equivalent to $g(y)$, where

$$\begin{aligned} g(y_1, \ldots, y_n) = \overline{\Pi}(u_0) + \mathrm{CS}\left(y_1, \ldots, y_k, -\sum_{i=1}^{k} y_i\right) \\ - \sum_{j=k+1}^{k+q} y_j^2 + \sum_{r=k+q+1}^{n} y_j^2\,. \end{aligned} \tag{11}$$

One observes that the second term in the right-hand side of (11) is sufficiently rich to describe locally every type of nonsmooth, finite-dimensional functions.

Furthermore, following [7] one notes that a PC^2-function can always be transformed into the min-max normal form:

$$\overline{\Pi}(u) = \max_{1 \leq i \leq k} \min_{j \in M_i} \overline{\Pi}_j(u)\,, \tag{12}$$

where (12) is considered as a local representation of the potential in a neighborhood of u_0, $M_i \subseteq \{1, \ldots, \mathrm{m}\}$ and the functions $\overline{\Pi}_j\colon\ U \to \mathbf{R}$, $j \in \{1, \ldots, \mathrm{m}\}$, are C^2-functions.

In this case a consistent nonsmooth second order approximation of the PC^2-potential, expressed by the normal form (12), is given by:

$$\max_{1\le i\le k}\min_{j\in M_i}\left\{\begin{array}{c}\overline{\Pi}_j(u_0)+\nabla\overline{\Pi}_j(u_0)^\top(u-u_0)\\ +\frac{1}{2}(u-u_0)^\top\nabla^2\overline{\Pi}_j(u_0)(u-u_0)\end{array}\right\}. \tag{13}$$

Note here that the previously denoted min-max form is not defined in an unique way.

Stability Results. Discussion

For a structural analysis system with a structured nonsmooth PC^2-potential with (11) and for a nondegenerate critical point $u_0 \in \mathbf{R}^n$, the local approximation (11) is available. Let us assume a potential of the external loading equal to $\lambda\, p^\intercal\, u$, as in (5) and let for the present the load parameter λ be fixed to a given value. From (11) the following complete subdivision of the coordinate space $\mathbf{R}^n$ arises:

$$\mathbf{R}^n = \mathbf{R}^k \oplus \mathbf{R}^q \oplus \mathbf{R}^{n-k-q} = \mathbf{R}^{\mathrm{non}} \oplus \mathbf{R}^{\mathrm{un}} \oplus \mathbf{R}^{\mathrm{st}}, \tag{14}$$

where $\mathbf{R}^{\mathrm{non}}$ stands for the essentially nondifferentiable subspace, $\mathbf{R}^{\mathrm{un}}$ for the unstable subspace and $\mathbf{R}^{\mathrm{st}}$ for the stable subspace. Let, moreover, the local coordinate transformation that leads to (11) be traced for the components of the vector $\lambda\, p$ (cf. (6)–(9)). Let the components of the last vector in the three subspaces of ((14)) be $\widetilde{p}^{\mathrm{non}}$, $\widetilde{p}^{\mathrm{un}}$ and $\widetilde{p}^{\mathrm{st}}$, respectively.

Further one considers the type of the CS in the linear term of the right-hand side in (11) in comparison with the three above defined components of the loading vector. This information can be used for stability analysis. Smooth and nonsmooth contributions should be treated separately. For the nonsmooth part, for example, if one has a max-type function and $q = 0$, then only stable local minima of the potential energy function arise.

The above outlined scheme can be followed for the derivation of stability considerations for a structure at a given point and for a given loading level (λ is constant). For the examination of the stability question along a given loading path (one-parametric change of λ) one should take into account that the local representation (11) may change as λ changes. The results are qualitative of the same nature, but, for practical applications, a combinatorial problem arises, which concerns the way of possible changes of the subdivision (14) as loading changes. Further work in this direction will generalize the computational mechanics techniques for the tracing of post-buckling equilibria in nonsmooth mechanics' applications. Theoretical support will be provided by the theory of parametric optimization (cf., e. g., [5] and the applications in contact mechanics [16,17]).

See also

- ▶ Generalized Monotonicity: Applications to Variational Inequalities and Equilibrium Problems
- ▶ Hemivariational Inequalities: Applications in Mechanics
- ▶ Hemivariational Inequalities: Eigenvalue Problems
- ▶ Hemivariational Inequalities: Static Problems
- ▶ Nonconvex Energy Functions: Hemivariational Inequalities
- ▶ Nonconvex-Nonsmooth Calculus of Variations
- ▶ Optimization Strategies for Dynamic Systems
- ▶ Quasidifferentiable Optimization
- ▶ Quasidifferentiable Optimization: Algorithms for Hypodifferentiable Functions
- ▶ Quasidifferentiable Optimization: Algorithms for QD Functions
- ▶ Quasidifferentiable Optimization: Applications
- ▶ Quasidifferentiable Optimization: Applications to Thermoelasticity
- ▶ Quasidifferentiable Optimization: Calculus of Quasidifferentials
- ▶ Quasidifferentiable Optimization: Codifferentiable Functions
- ▶ Quasidifferentiable Optimization: Dini Derivatives, Clarke Derivatives
- ▶ Quasidifferentiable Optimization: Exact Penalty Methods
- ▶ Quasidifferentiable Optimization: Optimality Conditions
- ▶ Quasidifferentiable Optimization: Variational Formulations
- ▶ Quasivariational Inequalities
- ▶ Sensitivity Analysis of Variational Inequality Problems
- ▶ Solving Hemivariational Inequalities by Nonsmooth Optimization Methods
- ▶ Variational Inequalities

- ▶ Variational Inequalities: F. E. Approach
- ▶ Variational Inequalities: Geometric Interpretation, Existence and Uniqueness
- ▶ Variational Inequalities: Projected Dynamical System
- ▶ Variational Principles

References

1. Bazant ZP, Cedolin L (1991) Stability of structures. Elastic, inelastic, fracture and damage theories. Oxford Univ. Press, Oxford
2. Curnier A, He Q-C, Zysset Ph (1995) Conewise linear elastic materials. J Elasticity 37:1–38
3. Demyanov VF, Rubinov AM (1985) Quasidifferentiable calculus. Optim. Software, New York
4. Demyanov VF, Stavroulakis GE, Polyakova LN, Panagiotopoulos PD (1996) Quasidifferentiability and nonsmooth modelling in mechanics, engineering and economics. Kluwer, Dordrecht
5. Guddat J, Guerra Vasquez F, Jongen ThH (1990) Parametric optimization: Singularities, path following and jumps. Teubner and Wiley, New York
6. Jongen ThH, Pallaschke D (1988) On linearization and continuous selection of functions. Optim 19(3):343–353
7. Kuntz L, Scholtes S (1995) Qualitative aspects of the local approximation of a piecewise differentiable function. Nonlinear Anal Th Methods Appl 25(2):197–215
8. Kurutz M (1993) Stability of structures with nonsmooth nonconvex energy functionals. The one dimensional case. Europ J Mechanics A Solids 12(3):347–385
9. Rohde A, Stavroulakis GE (1997) Genericity analysis for path-following methods. Application in unilateral contact elastostatics. Z Angew Math Mechanics (ZAMM) 77(10):777–790
10. Kurutz M (1994) Equilibrium paths of polygonally elastic structures. Mechanics of Structures and Machines 22(2):181–210
11. Li L-Y (1991) The criteria of identifying the type of critical points. Arch Appl Mechanics 61:231–235
12. Li L-Y (1994) Determination of stability in nonlinear analysis of structures. Arch Appl Mechanics 64:119–126
13. Mistakidis ES, Stavroulakis GE (1997) Nonconvex optimization in mechanics. Algorithms, heuristics and engineering applications by the F.E.M. Kluwer, Dordrecht
14. Pallaschke D, Rolewicz S (1997) Foundations of mathematical optimization. Convex analysis without linearity. Kluwer, Dordrecht
15. Panagiotopoulos PD (1985) Inequality problems in mechanics and applications. Convex and nonconvex energy functions. Birkhäuser, Basel
16. Panagiotopoulos PD (1993) Hemivariational inequalities. Applications in mechanics and engineering. Springer, Berlin
17. Rohde A, Stavroulakis GE (1995) Path–following energy optimization in unilateral contact problems. J Global Optim 6:347–365
18. Stavroulakis GE, Rohde A (1996) Stability of structures with quasidifferentiable energy functions. In: Sotiropoulos D, Beskos DE (eds) 2nd Greec Conf. Computational Mechanics, Chania, June 1996. GRACAM, pp 406–413
19. Stavroulakis GE, Rohde A (1999) Normal forms and stability in nonsmooth potential elastostatics. Mechanics Res Comm 26(2):185–190

Quasidifferentiable Optimization: Variational Formulations

GEORGIOS E. STAVROULAKIS
Carolo Wilhelmina Techn. University, Braunschweig, Germany

MSC2000: 74G99, 74H99, 74Pxx, 49J40, 49M05, 49S05

Article Outline

Keywords

Variational problems; Variational inequalities; Nonsmooth optimization; Nonsmooth mechanics; Quasidifferentiability

In science and, especially in engineering, the variational or weak formulation of a given boundary value problem has certain advantages. Instead of writing pointwise relations (for example, partial differential equations) which hold for each point of the considered system, one multiplies the governing relation with an arbitrary virtual variation, integrates over the entire area and requires that the latter integral be equal to zero. This is a weak or a variational formulation of the problem. Since the considered virtual variation is arbitrary

one gets back, on the assumption of sufficient regularity, the initial pointwise relations.

Variational formulations provide the basis for the development of numerical approximation methods (for example, by the *finite element method*). One of the advantages is that by performing partial integration one transfers differentiability requirements from the actual variables of the problem to the virtual ones, which, in turn, results in less demanding requirements on the complexity of the required finite element basis approximation functions. The literature on variational problems is very large, so that every selection of references would be incomplete. In this sense, let us mention here the publications [1,3,7,8,14].

In the language of smooth optimization, instead of considering the first order optimality condition that the derivative of a function at a given point is equal to zero, one proceeds as follows. The latter equation is multiplied by a virtual change of the variables along an arbitrary direction. Then, one considers the equivalent relation that the directional derivative of the function is equal to zero for all directions emanating from the assumed point.

In mechanics the arising quantities have a physical meaning (for instance, they correspond to the virtual work of a system). For historical reasons one speaks about *variational principles*. Moreover, on adequate smoothness assumptions one writes variational equalities. Finally, for engineering applications, and depending on the nature of the studied problem, one has to solve, after numerical discretization, systems of linear or nonlinear equations.

In connection with convex, nondifferentiable potentials or for convex problems with inequality constraints it is intuitively conceivable that not all virtual variations are allowed for. The theory of variational inequalities has been developed for the study of this class of problems. It is connected with the subdifferential of convex analysis and it is appropriate for the study of monotone operators [5,9]. In simple cases, or after appropriate reformulations one gets linear or nonlinear complementarity problems (see, e. g. [6] for a recent review).

For general nonconvex and nonsmooth problems a nonconvex extension of the notion of the variational inequality is required. For potential operators and by using the generalized subdifferential in the sense of F.H. Clarke, this class of variational problems have been developed and studied by P.D. Panagiotopoulos, who called them *hemivariational inequalities*. See ► Nonconvex energy functions: Hemivariational inequalities; ► Hemivariational inequalities: Applications in mechanics or [9,11] for more details.

The notion of quasidifferentiability, in the sense of V.F. Demyanov and A.M. Rubinov, provides an elegant way for the formulation and study of nonconvex variational inequality problems. By taking advantage of the ability of the quasidifferentials to provide a qualitative and quantitative nonsmooth approximation of a nonsmooth function one arrives at a very powerful variational description of the problem. This link has been studied for several applications in mechanics in [4,10,12]. One should mention that the author's understanding of this theory and their first attempts have been based on previous theoretical results of C.A. Stuart and J.F. Toland [13] and G. Auchmuty [2] concerning difference convex energy functions. Of course, the class of *difference convex functions* is included in the class of quasidifferentiable functions, so that the here presented approach is sufficiently general.

Variational Formulation of Subdifferential Laws

Let us assume a monotone possibly multivalued (i. e., with complete vertical branches) relation (a law) between the quantities u and $-f$. To be more precise, one may think about a nonlinear boundary law which connects boundary reactions $-f$ with boundary displacements u in mechanics. Let a convex l.s.c. and proper function Φ exists, the convex superpotential in the sense of J.-J. Moreau, and that the previously mentioned law is written in the subdifferential form:

$$-f \in \partial\Phi(u)\,. \tag{1}$$

Here ∂ denotes the subdifferential of convex analysis. Function $\Phi(u)$ can be considered as the potential energy corresponding to the mechanical law (1).

By definition, (1) is equivalent to the following variational inequality:

$$\Phi(u^*) - \Phi(u) \geq -\langle f, u^* - u\rangle\,, \quad \forall u^* \in \mathbf{R}\,. \tag{2}$$

For example, if Φ is the indicator I_K of a convex closed interval K of $\mathbf{R}$, then one has

$$-f \in \partial I_K(u)\,. \tag{3}$$

This is a unilateral constraint as one easily recognizes by considering the equivalent variational inequality (for $u \in K$):

$$\langle f, u^* - u\rangle \geq 0\,, \quad \forall u^* \in K\,. \tag{4}$$

Indeed, if $u_\star - u$ is an admissible variation of u (in the sense that it satisfies (4)), then the same does not hold for the variation $u - u_\star$. Of course in the one-dimensional case $\langle\cdot,\cdot\rangle$ is a simple multiplication. For multidimensional problems it will be an inner product.

Analogous subdifferential relations can be written for multidimensional laws (for example, for constitutive laws in elastoplasticity [9]).

Variational Formulation of Quasidifferential Laws

Let us assume now a nonmonotone possibly multivalued relation. By means of a real-valued, *quasidifferentiable superpotential* energy function Φ, one may expresses this relation in the form:

$$-f = w_1 + w_2\,, \tag{5}$$

with $\{w_1, w_2\} \in D\Phi(u) = [\underline{\partial}\Phi(u), \overline{\partial}\Phi(u)]$.

By definition, (5) is equivalent to the relation:

$$\begin{aligned}\langle -f, u^* - u\rangle = &\max_{w_1^* \in \underline{\partial}\Phi(u)} \langle w_1^*, u^* - u\rangle \\ &+ \min_{w_2^* \in \overline{\partial}\Phi(u)} \langle w_2^*, u^* - u\rangle\,, \\ &\forall u^* \in U\,,\end{aligned} \tag{6}$$

for $u \in U$, or with the *system of variational inequalities*

$$\begin{aligned}&\langle -f, u^* - u\rangle - \langle w_2^*, u^* - u\rangle \leq \max_{w_1^* \in \underline{\partial}\Phi(u)} \langle w_1^*, u^* - u\rangle\,, \\ &\forall u^* \in U\,, \quad \forall w_2^* \in \overline{\partial}\Phi(u)\,,\end{aligned} \tag{7}$$

and

$$\begin{aligned}&\langle -f, u^* - u\rangle - \langle w_1^*, u^* - u\rangle \geq \min_{w_2^* \in \overline{\partial}\Phi(u)} \langle w_2^*, u^* - u\rangle\,, \\ &\forall u^* \in U\,, \quad \forall w_1^* \in \underline{\partial}\Phi(u)\,.\end{aligned} \tag{8}$$

Space U is in general a subspace of $\mathbf{R}^n$ and depends on the considered application.

Analogously one treats multidimensional relations (for example, boundary adhesive layers) or constitutive laws (e. g., materials with softening effects). A number of concrete examples have been given in [4, Chap. 3].

Example: an Elastostatic Problem Involving QD-Superpotentials

Let $\Omega \subset \mathbf{R}^3$ be an open bounded subset occupied by a deformable body in its undeformed state. On the assumption of small deformations one writes the virtual work relation

$$\begin{aligned}&\int_\Omega \sigma_{ij}(u)\varepsilon_{ij}(v-u)\,d\Omega \\ &\quad = \int_\Omega f_i(v_i - u_i)\,d\Omega + \int_\Gamma \sigma_{ij} n_j (v_i - u_i) d\Gamma\,, \\ &\forall v \in V\,,\end{aligned} \tag{9}$$

for $u \in V$. Here V denotes the function space of the displacements which will be defined further. As it has been outlined previously, for the derivation of (9) one multiplies the equilibrium equation:

$$\sigma_{ij,j} + f_i = 0\,, \tag{10}$$

where f_i is the volume force vector, by a virtual displacement $v_i - u_i$ and then we have integrated over Ω. On the assumption of appropriately regular functions, one applies the Green–Gauss theorem by taking into account the strain-displacement relation

$$\varepsilon_{ij} = \frac{1}{2}(u_{i,j} + u_{j,i}). \tag{11}$$

Let us assume further that the body is linearly elastic, i. e. that

$$\sigma_{ij} = C_{ijhk}\varepsilon_{hk}\,, \tag{12}$$

where $C = \{C_{ijhk}\}$, $i, j, h, k = 1, 2, 3$, is the elasticity tensor which obeys to the well-known symmetry and ellipticity conditions. The energy bilinear form of linear elasticity is further denoted by $\alpha\,(u, v) = \int_\Omega C_{ijhk}\varepsilon_{ij}(u)\varepsilon_{hk}(v)d\Omega$.

Variational Equality

For example, let us assume first that on the boundary Γ of the structure the classical boundary conditions $S_N = 0$ and $u_{T_i} = 0$, $i = 1, 2, 3$, hold. Then one gets the classical

variational equality: Find $u \in V_0 = \{v\colon v \in V, v_{T_i} = 0$ on $\Gamma\}$ such that

$$\alpha(u, v) = \int_\Omega f_i v_i \, d\Omega \,, \quad \forall v \in V_0 \,.$$

Convex Variational Inequality

Furthermore, let us assume now that on Γ the general monotone multivalued subdifferential boundary condition (1) holds. Using (2) and (9) one obtains the following variational inequality: find $u \in V$ with $\Phi(u) < \infty$, such that

$$\alpha(u, v-u) + \int_\Gamma (\Phi(v) - \Phi(u)) \, d\Gamma \geq \int_\Omega f_i(v_i - u_i) \, d\Omega \,,$$
$$\forall v \in V : \; j(v) < \infty \,.$$

QD Laws and Systems of Variational Inequalities

Let us assume that on Γ the nonmonotone, possibly multivalued boundary condition (5) holds, where Φ is a quasidifferentiable functional. It has the form

$$-S = w_1 + w_2 \,,$$

with $\{w_1, w_2\} \in Dj(u) = \{\underline{\partial}\Phi(u), \overline{\partial}\Phi(u)\}$.

Then one has, by definition, the relation (6), where $\Phi'(u, v) = \langle -S, v\rangle$. Finally, by an analogous way, one has the variational problem: find $u \in V$, $w_1, w_2 \in W$ such as to satisfy the relation

$$\begin{aligned} &\alpha(u, v-u) - \int_\Omega f_i(v_i - u_i) \, d\Omega \\ &+ \max_{\substack{w_1^*(x) \in \underline{\partial}\Phi(u(x)) \\ \text{a.e. on } \Gamma}} \langle w_1^*, v-u\rangle \\ &+ \min_{\substack{w_2^*(x) \in \overline{\partial}\Phi(u(x)) \\ \text{a.e. on } \Gamma}} \langle w_2^*, v-u\rangle = 0 \,, \\ &\forall v \in V \,. \end{aligned} \tag{13}$$

The function spaces V and W depend on the studied application. For instance, for three-dimensional elastostatics the following choice has been proposed in [4]: $V = [H^1(\Omega)]^3$, $W = [L^2(\Gamma)]^3$. A more general formulation, also proposed in the previously given original publication, would be to assume that $w_1, w_2 \in [H^{-1/2}(\Gamma)]^3$.

Then in the left-hand side of (13) on should replace $w_1(x) \in \overline{\partial}\Phi(u(x))$ a.e. on Γ by $w_1 \in \overline{\partial}F(u)$ and $w_2(x) \in \underline{\partial}\,\Phi(u(x))$ a.e. on Γ by $w_2 \in \underline{\partial}F(u)$, where one assumes that

$$F(u) = \begin{cases} \displaystyle\int_\Gamma \Phi(u(x)) \, d\Gamma & \text{if } \Phi(\cdot) \in L^2(\Gamma), \\ \infty & \text{otherwise.} \end{cases}$$

Then instead of (13) one has the following problem: find $u \in [H^1(\Omega)]^3$, $w_1, w_2 \in [H^{-1/2}(\Gamma)]^3$ such that

$$\begin{aligned} &\alpha(u, v-u) - \int_\Omega f_i(v_i - u_i) \, d\Omega \\ &+ \max_{w_1^*} \{\langle w_1^* + w_2^*, v-u\rangle : \; w_1^* \in \underline{\partial}F(u)\} \\ &+ \min_{w_2^*} \{\langle w_1^* + w_2^*, v-u\rangle : \; w_1^* \in \underline{\partial}F(u)\} = 0 \,, \\ &\forall v \in [H^1(\Omega)]^3 \,. \end{aligned}$$

One should mention that the related questions concerning the extension of *QD*-superpotentials to function spaces remain still open.

Moreover we can write the min-max form which reads: find $u \in [H^1(\Omega)]^3$ such as to satisfy the relation:

$$\begin{aligned} &\alpha(u, v-u) - \int_\Omega f_i(v_i - u_i) \, d\Omega \\ &+ \min_{w_2^* \in \underline{\partial}F(u)} \max_{w_1^* \in \overline{\partial}F(u)} \{\langle w_1^* + w_2^*, v-u\rangle = 0 \,, \\ &\forall v \in [H^1(\Omega)]^3 \,. \end{aligned}$$

If in particular the superpotential F can be expressed as the difference of two convex functions, i. e. if $F = \Phi_1 - \Phi_2$, with Φ_1 and Φ_2 convex, then one has

$$\underline{\partial}F = \partial\Phi_1 \,, \quad \overline{\partial}F = -\partial\Phi_2 \,,$$

where $\partial$is the subdifferential of the convex analysis. In this case the following system of variational inequalities results, as it can easily be shown by using the definition of the subdifferential: find $u \in [H^1(\Omega)]^3$, such as to satisfy

$$\begin{aligned} &\alpha(u, v-u) - \int_\Omega f_i(v_i - u_i) \, d\Omega \\ &\quad - \langle w_2^*, v-u\rangle + \Phi_1(v) - \Phi_1(u) \geq 0 \,, \\ &\quad \forall v \in [H^1(\Omega)]^3 \end{aligned}$$

for all $w_2^\star \in [H^{-1/2}(\Gamma)]^3$ such that

$$\langle w_2^*, v-u\rangle \leq \Phi_2(v) - \Phi_2(u), \qquad \forall v \in [H^1(\Omega)]^3 \,.$$

Further information on variational formulations in elastostatics can be found in ► Hemivariational inequalities: Applications in mechanics.

See also

- ► Generalized Monotonicity: Applications to Variational Inequalities and Equilibrium Problems
- ► Hemivariational Inequalities: Eigenvalue Problems
- ► Hemivariational Inequalities: Static Problems
- ► Nonconvex Energy Functions: Hemivariational Inequalities
- ► Nonconvex-Nonsmooth Calculus of Variations
- ► Quasidifferentiable Optimization
- ► Quasidifferentiable Optimization: Algorithms for Hypodifferentiable Functions
- ► Quasidifferentiable Optimization: Algorithms for QD Functions
- ► Quasidifferentiable Optimization: Applications
- ► Quasidifferentiable Optimization: Applications to Thermoelasticity
- ► Quasidifferentiable Optimization: Calculus of Quasidifferentials
- ► Quasidifferentiable Optimization: Codifferentiable Functions
- ► Quasidifferentiable Optimization: Dini Derivatives, Clarke Derivatives
- ► Quasidifferentiable Optimization: Exact Penalty Methods
- ► Quasidifferentiable Optimization: Optimality Conditions
- ► Quasidifferentiable Optimization: Stability of Dynamic Systems
- ► Quasivariational Inequalities
- ► Sensitivity Analysis of Variational Inequality Problems
- ► Solving Hemivariational Inequalities by Nonsmooth Optimization Methods
- ► Variational Inequalities
- ► Variational Inequalities: F. E. Approach
- ► Variational Inequalities: Geometric Interpretation, Existence and Uniqueness
- ► Variational Inequalities: Projected Dynamical System
- ► Variational Principles

References

1. Argyris JH, Kelsey S (1960) Energy theorems and structural analysis. Butterworths, London
2. Auchmuty G (1989) Duality algorithms for nonconvex variational principles. Numer Funct Anal Optim 10:211–264
3. Dautray R, Lions J-L (1998) Functional and variational methods. Math Anal and Numer Methods for Sci and Tech, vol 2. Springer, Berlin
4. Demyanov VF, Stavroulakis GE, Polyakova LN, Panagiotopoulos PD (1996) Quasidifferentiability and nonsmooth modelling in mechanics, engineering and economics. Kluwer, Dordrecht
5. Ekeland I, Temam R (1976) Convex analysis and variational problems. North-Holland, Amsterdam
6. Ferris MC, Pang JS (1997) Engineering and economic applications of complementarity problems. SIAM Rev 39(4):669–713
7. Lippmann H (1960) Extremum and variational principles in mechanics. CISM Courses and Lectures, vol 54. Springer, Berlin
8. Oden JT, Reddy JN (1982) Variational methods in theoretical mechanics. Springer, Berlin
9. Panagiotopoulos PD (1985) Inequality problems in mechanics and applications. Convex and nonconvex energy functions. Birkhäuser, Basel
10. Panagiotopoulos PD (1988) Nonconvex superpotentials and hemivariational inequalities. Quasidiferentiability in mechanics. In: Moreau JJ, Panagiotopoulos PD (eds) Nonsmooth Mechanics and Applications. CISM Lecture Notes. Springer, Berlin, pp 83–176
11. Panagiotopoulos PD (1993) Hemivariational inequalities. Applications in mechanics and engineering. Springer, Berlin
12. Panagiotopoulos PD, Stavroulakis GE (1992) New types of variational principles based on the notion of quasidifferentiability. Acta Mechanics 94:171–194
13. Stuart CA, Toland JF (1980) A variational method for boundary value problems with discontinuous nonlinearities. J London Math Soc (2) 21:319–328
14. Washizu K (1968) Variational methods in elasticity and plasticity. Pergamon, Oxford

Quasivariational Inequalities

GEORGIOS E. STAVROULAKIS
Carolo Wilhelmina Techn. University,
Braunschweig, Germany

MSC2000: 49J40, 70-08, 49Q10, 74K99, 74Pxx

Article Outline

Keywords

Variational inequalities; Nonsmooth mechanics; Implicit variational problems

Variational or weak formulations of boundary value problems in science and, in particular, in engineering are integral, energetic expressions of all involved quantities (involving differential equations and boundary conditions). Usually, under differentiability (smoothness) assumptions of the involved variables and equalities throughout the considered model one gets variational equality problems. The strong formulation of the initial problem (i. e., constitutive relations, boundary conditions, etc) can be reconstructed if one considers certain values for the (otherwise arbitrary) variations in the weak form, i. e., in the variational equality. Variational formulations provide the basis for modern computational mechanics techniques (e. g., the *finite element method*) and for this reason they have been extensively studied in the affiliated literature (see, among others, [22]). In terms of optimization they can be considered as stationary point statements for the total differential of an appropriately constructed (convex or nonconvex) potential energy function, provided that the studied problem admits a potential. Namely, the weak formulation expresses the fact that the variation of a function for every small variation of the involved independent variables is equal to zero, which, due to the arbitrariness of the variations, is equivalent to the more classical requirement that the first derivative of the function vanishes at a critical point.

Due to inequality-type constraints or due to lack of differentiability in the involved functions one is sometimes obliged to consider one-sided (unilateral) variations of the problem's variables. A systematic way of doing so is provided by the theory of *variational inequalities* [13]. They are related to monotone operators, to convex, nondifferentiable optimization problems and to complementarity problems. Variational inequalities have been applied for the study of problems in engineering [17,20], economics, transportation planning and flow in networks (see also [6,8,10]).

Extensions for nonconvex variational inequalities, which are based on the generalized gradient approach in the sense of F.H. Clarke, have been proposed and studied by P.D. Panagiotopoulos who named them *hemivariational inequalities*. Details are given in ► Nonconvex energy functions: Hemivariational inequalities and ► Hemivariational inequalities: Applications in mechanics. Parallel developments which are based on the notion of the quasidifferentiability in the sense of V.F. Demyanov and A.M. Rubinov are described in ► Quasidifferentiable optimization: Variational formulations.

Furthermore, there exist problems where the admissible space (for the variables and their variations) or the involved potentials depend on the solution of the problem. This class of implicit variational inequality problems are called quasivariational inequalities. They have been used for the modeling of stochastic impulsive control problems, in free boundary problems, in mechanics and in economy. The interested reader may find more information in the references [1,2,7,9,15]. Here a short outline of quasivariational inequality problems is given. A model application arising in unilateral contact problems with Coulomb friction in engineering mechanics demonstrates the discussed ideas. This approach is based on early theoretical and numerical studied of [19] (see also numerical applications in [11,12]) and, among others, have recently been tested for several convex and nonconvex problems of mechanics in [14].

Variational Inequalities

Let us first consider abstract variational formulations of a boundary value problem which is defined in a subset Ω of $\mathbf{R}^n$, $n = 1, \ldots, n$, with boundary Γ. Let V be a real Hilbert space and V' be its dual space. Let $a(\cdot, \cdot)$: $V \times V \to \mathbf{R}$ be a symmetric, continuous and coercive bilinear form and (l, ·) be a continuous linear form on V. An

abstract variational problem reads: find $u \in V$ such that

$$a(u, v-u) = (l, v-u)\,, \quad \forall v \in V\,. \tag{1}$$

Let moreover K be a closed convex subset of V and assume that a solution of the boundary value problem within the set K is sought. It can be shown that this solution is characterized by the following abstract variational inequality (of the G. Fichera type, see [20, p. 188]):

$$\begin{cases} \text{Find} & u \in K \subset V \\ \text{s.t.} & a(u, v-u) \geq (l, v-u), \\ & \forall v \in K. \end{cases} \tag{2}$$

For a convex, l.s.c. proper functional Φ on V one may define the more general (nonlinear) variational inequality ([14]):

$$\begin{cases} \text{Find} & u \in V \\ \text{s.t.} & \alpha(u, v) + \Phi(v) - \Phi(u) \geq \\ & \quad \geq (l, v-u), \\ & \forall v \in V. \end{cases} \tag{3}$$

It is obvious that (2) is a special case of (3), with $\Phi = I_K$, where the *indicator function* of the set K is defined by $I_K(v) = 0$ if $v \in K$, $+\infty$ otherwise.

Let moreover j: $\mathbf{R} \to \mathbf{R}$ denotes a locally Lipschitz function and let $j^0(u, v{-}u)$ denotes the generalized gradient of the nonconvex and nonsmooth function j. By definition, one has the following connection with the generalized gradient, in the sense of Clarke:

$$j^0(u, v) = \{\max \langle w, v\rangle : \ w \in \partial_{\mathrm{CL}} j(u)\}\ . \tag{4}$$

A hemivariational inequality problem reads:

$$\begin{cases} \text{find} & u \in V \\ \text{s.t.} & a(u, v-u) + \displaystyle\int_\Omega j^0(u, v-u)\, d\Omega \\ & \quad \geq (l, v-u), \\ & \forall v \in V. \end{cases} \tag{5}$$

Implicit Variational Inequalities and Quasivariational Inequalities

If one assumes for instance that the linear form $(l, \cdot)$ or the set K in the previous relations depend on the solution u, one gets various types of implicit variational inequalities or quasivariational inequalities.

Let the set K be a variable of the solution u. Then from (2) one gets the quasivariational inequality:

$$\begin{cases} \text{find} & u \in K(u) \subset V \\ \text{s.t.} & \alpha(u, v) \geq (l, v-u), \\ & \forall v \in K(u). \end{cases}$$

Along the same lines one formulates from (3) the implicit variational inequality:

$$\begin{cases} \text{find} & u \in V \\ \text{s.t.} & \alpha(u, v) + \Phi(u; v) - \Phi(u; u) \\ & \quad \geq (l, v-u), \\ & \forall v \in V. \end{cases}$$

Here the first argument in $\Phi(\cdot, \cdot)$ is tackled as a parameter. A concrete application of this method will be demonstrated by the mechanical problem in the next section.

Finally, in analogy to the previous extensions, for a continuous mapping $h(u)$ the following quasihemivariational inequality (which may also be characterized as implicit hemivariational inequality) problem can be written (see [16, p. 128]):

$$\begin{cases} \text{find} & u \in V \\ \text{s.t.} & a(u, v-u) + h(u) j^0(u, v-u) d \\ & \quad \geq l(v-u), \\ & \forall v \in V. \end{cases} \tag{6}$$

Mechanical Example: Coupled Unilateral Contact Problem with Friction

Let $\Omega \in \mathbf{R}^3$ be an open bounded subset occupied by a deformable body in its undeformed state. On the assumption of small deformations one writes the virtual work relation (for $u \in V$)

$$\begin{aligned} \int_\Omega \sigma_{ij}\varepsilon_{ij}(v-u)\, d\Omega = &\int_\Omega f_i(v_i - u_i)\, d\Omega \\ &+ \int_\Gamma S_N(v_N - u_N)\, d\Gamma \\ &+ \int_\Gamma S_{T_i}(v_{T_i} - u_{T_i})\, d\Gamma\ , \\ \forall v \in V\,. & \end{aligned} \tag{7}$$

Here V denotes the function space of the displacements, which in general is an appropriate subset of $H^1(\Omega)$ and f_i, S_N, $S_{T_i} \in L_2(\Gamma)$. Recall here that the abstract bilinear form $\alpha(\cdot, \cdot)$ reads in this case of linear elasticity

$$\alpha(u, v) = \int_\Omega C_{ijhk}\varepsilon_{ij}(u)\varepsilon_{hk}(v)\, d\Omega\,. \tag{8}$$

Moreover the underlying elastostatic equilibrium equation boundary value problem has the form:

$$\sigma_{ij,j} + f_i = 0\,, \tag{9}$$

where the f_i is the volume force vector. One recalls here the strain-displacement relation (small deformation theory):

$$\varepsilon_{ij} = \frac{1}{2}(u_{i,j} + u_{j,i})\,. \tag{10}$$

Let a linearly elastic body be assumed, i. e., the constitutive material relation reads:

$$\sigma_{ij} = C_{ijhk}\varepsilon_{hk}\,,$$

where $C = \{C_{ijhk}\}$, $i, j, h, k = 1, 2, 3$, is the elasticity tensor which satisfies the well-known symmetry and ellipticity properties.

Recall here that on the assumption that classical support conditions hold on Γ (i. e., say $u_N = 0$ and $u_{T_i} = 0$, $i = 1, 2, 3$) one gets the following variational equality:

$$\begin{cases} \text{Find} \quad u \in V_0 = \left\{ v \in V\colon \begin{array}{r} v_N = 0, \\ v_{T_i} = 0 \\ \text{on } \Gamma \end{array} \right\} \\ \text{s.t.} \quad \alpha(u, v) = \displaystyle\int_\Omega f_i v_i\, d\Omega, \\ \qquad \forall v \in V_0. \end{cases} \tag{11}$$

Signorini–Coulomb Unilateral Frictional Contact

Let us assume the pointwise unilateral contact relations (known as *Signorini condition*, for the frictionless unilateral contact case):

$$\begin{aligned} &-S_N \geq 0\,, \quad u_N - g \leq 0\,, \\ &-S_N(u_N - g) = 0 \quad \text{on } \Gamma\,. \end{aligned} \tag{12}$$

Here, the inequalities on the boundary tractions correspond to the mechanical restriction that no tensile tractions are permitted. Moreover, the normal boundary displacements should not be greater that a given initial distance g, because no penetration is allowed. Finally, the complementarity relation expresses the physical fact that either contact is realized or a separation takes place.

A simplified static version of the Coulomb's friction law connects the tangential (frictional) forces S_{T_i} with the normal (contact) forces S_N by the relation

$$\gamma = \mu\,|S_N| - |S_T| \geq 0\,. \tag{13}$$

Here $|*|$ denotes the norm in $\mathbf{R}^3$ and μ is the friction coefficient. The friction mechanism is considered to work in the following way: If $|S_T| < \mu|S_N|$ (i. e. $\gamma > 0$) the slipping value γ must be equal to zero and if $|S_T| = \mu|S_N|$ (i. e. $\gamma = 0$) then we have slipping in the opposite direction of S_T. Explicitly we have:

$$\begin{cases} \text{if } \gamma > 0 \quad \text{then } y_T = 0, \\ \text{if } \gamma = 0 \quad \text{then there exists } \sigma > 0 \\ \qquad\qquad \text{s.t. } y_{Ti} = -\sigma S_{Ti}, \end{cases} \tag{14}$$

where $i = 1, 2, 3$ refers to the components of vector S_T with respect to a reference Cartesian coordinate system.

Contact law (12) can be written in the superpotential form:

$$-S_N \in \partial \mathbf{I}_{U_{\text{ad}}}(u_N) = \partial \Phi_N(u) = \mathcal{N}_{U_{\text{ad}}}(u_N)\,. \tag{15}$$

Here the set of admissible displacements is introduced:

$$U_{\text{ad}} = \{u \in V\colon\ u_N - g \leq 0\} \tag{16}$$

and the notions of the convex analysis subdifferential and of the normal cone to a set have been used. The corresponding variational inequality reads

$$-S_N(u_N)(v_N - u_N) \leq 0\,, \quad \forall v_N \in U_{\text{ad}}\,. \tag{17}$$

For the friction law one writes, analogously:

$$-S_T \in \partial_{u_N}(\mu\,|S_N|\,|u_N|) = \partial \Phi_T(S_N; u_T)\,, \tag{18}$$

where the involved potential is nondifferentiable (due to the absolute value nonlinearity of $|u_N|$) and depends on the normal contact traction S_N, thus, implicitly, on the solution of the problem u, i. e. one considers the parametrized potential $\Phi_T(u;u_T) = \Phi_T(S_N;u_T)$

$= \mu|S_N||u_N|$. The corresponding variational inequality reads

$$-S_T(u_T)(v_T - u_T) \le \Phi_T(u; v_T) - \Phi_T(u; u_T)\,, \quad \forall v_T \in U_{\text{ad}}\,. \tag{19}$$

Combining relations (7), (9) and (15) one gets the implicit variational inequality: find $u \in U_{ad}$ such that

$$\int_\Omega \sigma_{ij}\varepsilon_{ij}(v-u)\,d\Omega + \Phi_T(u; v_T) - \Phi_T(u; u_T) \ge \int_\Omega f_i(v_i - u_i)\,d\Omega\,, \quad \forall v \in U_{\text{ad}}\,. \tag{20}$$

A dual problem in terms of stresses provides us a corresponding quasivariational inequality problem. For simplicity, a two-dimensional problem is considered further. Moreover, the following set of admissible boundary tractions for the Signorini–Coulomb unilateral contact problem is assumed:

$$S_{\text{ad}} = \{(S_N, S_T):\ g_j(S_N, S_T) \le 0,\ j = 1, 2\}\,,$$

where the constraint functions have the form:

$$g_1(S_N, S_T) = \mu S_N - S_T\,,$$
$$g_1(S_N, S_T) = \mu S_N + S_T\,.$$

Moreover one needs the set of admissible stresses (which include the boundary tractions): $\Sigma(\sigma) = \{\sigma: \sigma_{ij,j} + f_i = 0\} \cap S_{ad}$. It may be shown that in this case the previous problem is expressed in the form of the quasivariational inequality:

$$\begin{cases} \text{find} & \sigma \in \Sigma(\sigma) \\ \text{s.t.} & \displaystyle\int_\Omega \varepsilon_{ij}(\tau_{ij} - \sigma_{ij})\,d\Omega \ge 0, \\ & \forall \tau \in \Sigma(\sigma) \end{cases}$$

Numerical Algorithms: Applications

Theoretical results and numerical algorithms can be found in several books dealing with variational inequalities, convex analysis and their applications. For the numerical solution, usually one solves, iteratively, a number of variational inequality problems. The resulting series approximates the solution of the initial quasivariational inequality. Further information and references, mainly connected with the mechanical problems used as model applications in this paper and their generalizations, can be found in [3,5,14,18,21]. Finally iterative solution methods can be based on multilevel optimization techniques, as it is discussed in ▶ Multilevel optimization in mechanics.

See also

- ▶ Generalized Monotonicity: Applications to Variational Inequalities and Equilibrium Problems
- ▶ Hemivariational Inequalities: Applications in Mechanics
- ▶ Hemivariational Inequalities: Eigenvalue Problems
- ▶ Hemivariational Inequalities: Static Problems
- ▶ Nonconvex Energy Functions: Hemivariational Inequalities
- ▶ Nonconvex-Nonsmooth Calculus of Variations
- ▶ Quasidifferentiable Optimization
- ▶ Quasidifferentiable Optimization: Algorithms for Hypodifferentiable Functions
- ▶ Quasidifferentiable Optimization: Algorithms for QD Functions
- ▶ Quasidifferentiable Optimization: Applications
- ▶ Quasidifferentiable Optimization: Applications to Thermoelasticity
- ▶ Quasidifferentiable Optimization: Calculus of Quasidifferentials
- ▶ Quasidifferentiable Optimization: Codifferentiable Functions
- ▶ Quasidifferentiable Optimization: Dini Derivatives, Clarke Derivatives
- ▶ Quasidifferentiable Optimization: Exact Penalty Methods
- ▶ Quasidifferentiable Optimization: Optimality Conditions
- ▶ Quasidifferentiable Optimization: Stability of Dynamic Systems
- ▶ Quasidifferentiable Optimization: Variational Formulations
- ▶ Sensitivity Analysis of Variational Inequality Problems
- ▶ Solving Hemivariational Inequalities by Nonsmooth Optimization Methods
- ▶ Variational Inequalities
- ▶ Variational Inequalities: F. E. Approach
- ▶ Variational Inequalities: Geometric Interpretation, Existence and Uniqueness

- ► Variational Inequalities: Projected Dynamical System
- ► Variational Principles

References

1. Baiocchi C, Capelo A (1984) Variational and quasivariational inequalities. Applications to free boundary problems. Wiley, New York
2. Bensoussan A, Lions J-L (1984) Impulse control and quasivariational inequalities. Gauthier-Villars
3. Bisbos CC (1995) A competitive game algorithm with five players for unilateral contact problems including the rotational and the thermal degrees of freedom. In: Raous M, Jean M, Moreau JJ (eds) Contact Mechanics. Plenum, New York, pp 251–258
4. Brézis H (1972) Problèmes unilatéraux. J Math Pures Appl 51:1–168
5. Curnier A, He Q-C, Telega JJ (1992) Formulation of unilateral contact between two elastic bodies undergoing finite deformations. CR Acad Sc Paris Ser II-1, p 314
6. Ferris MC, Pang JS (1997) Engineering and economic applications of complementarity problems. SIAM Rev 39(4):669–713
7. Friedman A and Spruck J (eds) (1993) Variational and free boundary problems. IMA vol Math Appl. Springer, Berlin
8. Friesz TL, Bernstein DH, Strough R (1996) Dynamic systems, variational inequalities and control theoretic models for predicting time-varying urban network flows. Transport Sci 30:14–31
9. Harker PT (1991) Generalized Nash games and quasivariational inequalities. Europ J Oper Res 54:81–94
10. Harker PT, Pang JS (1990) Finite dimensional variational inequality and nonlinear complementarity problems: A survey of theory, algorithms and applications. Math Program 48:161–220
11. Kalker JJ (1988) Contact mechanical algorithms. Comm Appl Numer Meth 4:25–32
12. Kalker JJ (1990) Three-dimensional elastic bodies in rolling contact. Kluwer, Dordrecht
13. Kinderlehrer D, Stampaccia G (1980) An introduction to variational inequalities and their application. Acad. Press, New York
14. Mistakidis ES, Stavroulakis GE (1998) Nonconvex Optimization in Mechanics. Algorithms, heuristics and engineering applications by the F.E.M. Kluwer, Dordrecht
15. Mosco V (1976) Implicit variational problems and quasivariational inequalities. In: Nonlinear operators and the calculus of variations. Lecture Notes Math, vol 543. Springer, Berlin, pp 83–156
16. Naniewicz Z, Panagiotopoulos PD (1995) Mathematical theory of hemivariational inequalities and applications. M. Dekker, New York
17. Oden JT, Kikuchi N (1988) Contact problems in elasticity: A study of variational inequalities and finite element methods. SIAM, Philadelphia
18. Outrata J, Kočvara M, Zowe J (1998) Nonsmooth approach to optimization problems with equilibrium constraints. Kluwer, Dordrecht
19. Panagiotopoulos PD (1975) A nonlinear programming approach to the unilateral contact and friction boundary value problem in the theory of elasticity. Ingen Archiv 44:421–432
20. Panagiotopoulos PD (1985) Inequality problems in mechanics and applications. Convex and nonconvex energy functions. Birkhäuser, Basel
21. Telega JJ (1988) Topics on unilateral contact problems of elasticity and inelasticity. In: Moreau JJ, Panagiotopoulos PD (eds) Nonsmooth Mechanics and Applications. CISM Courses and Lectures, vol 302. Springer, Berlin, pp 341–462
22. Washizu K (1968) Variational methods in elasticity and plasticity. Pergamon, Oxford