Multilevel Methods for Optimal Design

Natalia M. Alexandrov

NASA Langley Research Center, Hampton VA, USA
n.alexandrov@nasa.gov

Keywords: Nonlinear optimization; Multilevel; Bilevel; Hierarchical; Multidisciplinary design optimization

1 Introduction

Multilevel, or *hierarchical*, programming problems (MLP) are constrained optimization programs in which subsets of the solution set are themselves solution sets of other, lower-level optimization programs. Several general MLP problem statements exist. They differ from one another in the specifics of optimization variable distribution among the levels, as well as the definition of the objectives and constraints at particular levels.

Given a set of objectives $\{f_i\}_{i=1,\ldots,M}$ with f_i: $\mathbf{R}^n \to \mathbf{R}$ and a vector of variables $x \in \mathbf{R}^n$, partitioned into subsets $x = (x_1, \ldots, x_M)$ for some integer M denoting the number of subsystems, a prototypical form of MLP may be stated as follows:

$$\begin{cases} \min_{x_1 \in S_1} & f_1(x) \\ \text{s.t.} & x_2 \in \operatorname{argmin}_{x_2 \in S_2}\{f_2(x)\} \\ & \vdots \\ & x_M \in \operatorname{argmin}_{x_M \in S_M}\{f_M(x)\}, \end{cases}$$

where the optimization problem at each level i controls its own subset of variables x_i, while the other subsets of variables $x_1, \ldots, x_{i-1}, x_{i+1}, x_M$ serve as parameters. The constraint set for each level is $S_i \equiv \{x\colon h_i(x) = 0,\ g_i(x) \geq 0\}$ with h_i: $\mathbf{R}^n \to \mathbf{R}^{m_{h_i}}$ and g_i: $\mathbf{R}^n \to \mathbf{R}^{m_{g_i}}$ for some integers m_{h_i}, m_{g_i}.

This form of MLP, inspired by the work of H. Stackelberg [94] can be viewed as an M-player Stackelberg game [18,85]. Its interpretation is that of M autonomous players or decision makers seeking to minimize their (possibly constrained) objective functions while manipulating subsets of decision or design variables disjoint from those of other decision makers. The higher-level problems are implicit in the variables of the lower-level problems. This formulation has been studied widely in the bilevel case. See, for example, [15] and the references therein. In general, all problem levels, but the outermost one, may contain a number of concurrent optimization problems.

A related variant of the problem, known as the *generalized bilevel programming problem*, represents the reaction of the lower-level problem to decisions made by the upper-level problem via a solution of an equilibrium problem stated as a variational inequality:

$$\begin{cases} \min_{\substack{x\in X,\\ y\in Y(x)}} & f_1(x,y) \\ \text{s.t.} & \langle f_2(x,y), y-z\rangle \leq 0 \quad \text{for all } z \in Y(x), \end{cases}$$

where the upper-level feasibility domain X is such that the lower-level feasibility domain $Y(x)$ is not empty. This formulation was introduced by P. Marcotte in [63] and studied in [45,64], and [71].

Multilevel problems may be partitioned into two classes with respect to another criterion [103]. In one of the classes, upper-level optimization problems depend on the corresponding lower-level ones through the *optimal value functions* (or the *marginal functions*) of the lower-level problems. An optimal value function represents the value of a lower-level objective function at a solution of that lower-level problem. In the other class, upper-level problems depend on the corresponding lower-level problems through the actual optimal solutions of the latter. An example illustrating such formulations in engineering design optimization will be given further in the paper.

General MLP are very difficult to solve, due to the nature of constraints represented as general nonlinear programming problems. The problem has received recent attention in the machine learning community; see, e.g., [80] and references therein.

Despite the inherent solution difficulty, multilevel problems arise in numerous applications where the structure of the application involves hierarchical *decision making* or where the sheer size and complexity of the problem necessitates partitioning of the system and processing the subsystems in a hierarchical fashion. Information about early applications of multilevel optimization in such varied areas as power systems, water resource systems, urban traffic systems, and river pollution control can be found in [36,50,51,52,62,69,70,86], and many other references. The use of multilevel algorithms in engineering control is well documented, for instance, in [46] and [57].

The area of *multidisciplinary design optimization*(MDO)—a term that denotes a broad set of engineering disciplinary analyses that govern the behaviour of various system aspects, combined with design optimization formulations and solution methodologies for the design of complex coupled engineering systems—is particularly amenable to the use of multilevel methods, due to the extreme computational expense and the organizational complexity of the field. For instance, the design of aircraft involves aerodynamics, structural analysis, control, weights, propulsion, and cost, to list a few disciplines. The complexity and expense of each discipline have assured that most disciplines have developed into vast, autonomous fields of study, so that practically feasible optimization methods that involve the contributing disciplines must take into account such *disciplinary autonomy* and the hierarchical organization. Maintaining disciplinary autonomy while accounting for interdisciplinary subsystem couplings and allowing for integrated system optimization with respect to system and interdisciplinary objectives is one of the tasks of MDO methodology. Overviews of multidisciplinary optimization may be found in [4], [91], and more recently in [93].

Practitioners of engineering design have been using multilevel methods in some form since optimization algorithms made their appearance in engineering problems. The seminal works [60,65], and [101] contributed to a systematic development and understanding of hierarchical optimization. Multilevel methods have been studied extensively in application to multidisciplinary design ([16,17,22,99,100]) and single-discipline design areas that give rise to large problems, such as structural optimization (e.g., [74,88,92]). Engineering multilevel optimization has always had a strong connection with *multi-objective optimization* (e.g., [53]).

Problem Formulation

Systematic formulation of an engineering design problem as a multilevel or a bilevel problem amenable to tractable computation is effort-intensive and depends on the complexity and size of the problem. The general components in formulating a multilevel optimization problem are as follows.

- The original problem is studied to determine its structure. Structure is of paramount importance in deciding to adopt a particular formulation. For instance, most formulations assume that the problem subsystems share only a relatively small number of variables, i. e., that the *bandwidth of interdisciplinary coupling* is relatively small.
- The problem is partitioned into a system (or upper-level) problem and subsystem (or lower-level) problems. Decisions are made on inclusions of particular variables and constraints into the system and subsystems. Decisions are also made on the form of the system and subsystem objectives.
- Finally, algorithms are selected for solving the system and subsystem optimization problems. One must distinguish a formulation of the problem from the algorithm used to solve that formulation. While some of the multilevel formulations can be easily shown to be mathematically equivalent to the original problem with respect to solution sets, they may not be equivalent with respect to other attributes, such as constraint qualifications and optimality conditions. Hence the numerical properties of algorithms applied to different formulations vary widely [6,7,8].

Problem *decomposition* constitutes a special area of study. In general, decomposition techniques take advantage of the problem structure and depend on the strength and bandwidth of couplings among the subsystems. *Separable* and partially separable problems are particularly amenable to decomposition. Two types of decomposition may be considered in design optimization. Coarse-grained decomposition with respect to disciplines presents no conceptual difficulty, because the design problem initially consists of autonomous areas. The difficulty at this level of problem formulation is in *integration* or *synthesis*. However, in realistic applications, even though the coarse-grained decomposition is frequently obvious, the complexity of the problem requires that a *dependence analysis* be performed in order to determine the most advantageous arrangement or sequencing of the disciplinary subsystems in the optimization procedure. Automatic techniques based on graph-theoretic foundations may be found in [78] and [79], for instance.

Finer-grained decomposition within a particular discipline may be addressed by a multitude of techniques for decomposition of mathematical programs. Extensive references on decomposition in general mathematical programming, beginning with [19] and [31], and extended in [49] and many others, can be found in [42] and [43]. Further references to decomposition techniques aimed specifically at design problems can be found in [98].

As general MLP is difficult to solve, many multilevel formulations and algorithms in engineering design rely on heuristics rather than on theoretically substantiated foundations. There are exceptions, however, such as those in [11,29,68], and [75]. While many engineering multilevel approaches have enjoyed success when applied to specific problems, insufficient analytical foundation and the difficulty of the problem usually mean that the approaches are not robust, and extensive 'fine-tuning' of heuristic parameters is required for each new problem or instance of a problem. Hence, since the 1990's, there has been renewed interest in systematic, analytically substantiated approaches to MLP that continues to this day. Many such developments have taken place in *bilevel optimization*.

Bilevel Optimization

Although bilevel optimization problems (BLP) form the simplest case of multilevel optimization, they are very difficult to solve and constitute a fertile research area. A survey of the field can be found in [13] and [28]. A large bibliography with an emphasis on theoretical developments is also provided in [97], and a more recent summary with examples is in [10].

The prototypical general bilevel problem may be posed as follows.

$$\begin{cases} \min_{x \in X} & f_1(x,y) \\ \text{s.t.} & h_1(x,y) = 0 \\ & g_1(x,y) \geq 0, \end{cases}$$

where y solves for fixed x:

$$\begin{cases} \min_{y \in Y} & f_2(x,y) \\ \text{s.t.} & h_2(x,y) = 0 \\ & g_2(x,y) \geq 0. \end{cases}$$

The cases of linear and convex problem functions have been studied widely. A popular class of methods for the linear bilevel problem—extreme point algorithms—computes global solutions by enumerating extreme points of the lower-level feasible set (e.g., [27]). Convex bilevel problems are often solved by branch and bound methods (e.g., [15]). A survey of methods for linear and convex bilevel programming can be found in [9]. The considerably more difficult case of nonlinear and nonconvex problem functions has inspired much research activity as well. Major approaches to nonlinear bilevel optimization can be classified into several categories.

Penalty-Based Methods

In one set of algorithms (e.g., [1]), a barrier function penalizes the lower-level objective. In double-penalty methods, both the lower-level problem and the upper-level problem are approximated by sequences of unconstrained optimization problems [56,61,64]. Single or double-penalty methods are, in general, expected to converge slowly, especially for highly nonlinear problems. Thus using these methods for the usually large and nonlinear engineering design optimization problems may be difficult.

KKT-Based Methods

The algorithms of this category convert the bilevel problem into a nonconvex, single-level optimization problem by using the *Karush-Kuhn-Tucker conditions* (KKT conditions) of the lower-level problem as constraints on the upper-level problem [14,15,20,37,44]. If the lower-level problem is convex, the KKT formulation is equivalent to the original formulation [14]. However, even in this case, the KKT conditions on the lower-level problem include the *complementarity slackness* condition as a constraint. The form of the complementarity condition makes the single-level problem difficult to solve. The KKT formulation suffers from an additional difficulty. Namely, it is well known from the study of the sensitivity and stability of nonlinear programming (e.g., [40]) that even if the lower-level problem behaves exceedingly well in that it satisfies such stringent assumptions as *strong second order sufficiency* and *regularity* as a *constraint qualification*, the *feasible set* of the single-level problem will generally not be differentiable with respect to x. Hence, the performance of gradient-based solvers on the transformed problem may be adversely affected.

Descent-Based Methods

Another category of algorithms is based on solving subproblems that result in descent for the upper-level problem with gradient information of the lower-level problem used in a number of ways [34,39,59,85]. The remainder of the article will be devoted to a more detailed description of two specific approaches to nonlinear, nonconvex problems that arose from the need to solve engineering design problems. One approach is a bilevel formulation, the other is an algorithm for solving multilevel formulations.

Machine Leaning

Recently, in the area of machine learning, Franceschi et al. [38] proposed a method for solving bilevel optimization problems by replacing the lower-level problems with steepest descent update equations given a pre-selected number of iterations. Sato et al. [80] have extended the approach to multilevel optimization with an arbitrary number of levels. The extension includes a proof of asymptotic convergence to the original multilevel problem and a demonstration in the context of machine learning.

Example: Collaborative Optimization for Engineering Design

Collaborative optimization (CO) is a general approach to solving multidisciplinary design optimization problems by formulating them as nonlinear bilevel programs of special structure. CO comprises a number of methods. Its antecedents can be traced to earlier hierarchical approaches, as in [60] and [101]. The underlying idea of CO appeared in [12,81,82,83,89] and [99,100]. In the 1990's, the approach and its variants have received attention under the name of *collaborative optimization* [22,23,87,95].

Given that MDO problems are naturally partitioned into subsystems along disciplinary lines, CO suggests an intuitively attractive way to formulate the optimization problem so that the autonomy of the disciplinary subsystem computations is preserved. However, the approach presents a problem that is difficult to solve by means of conventional nonlinear programming software [7,58]. The analytical and computational aspects of CO were addressed in [7], of which the following discussion is an abstract. As a complete description of CO is lengthy, only an abbreviated version is considered here.

Let the original system be composed of a number, say M, of interdependent but autonomous systems, each of which is described by a disciplinary analysis A_i, $i = 1, \ldots, M$, expressed in the form

$$A_i(x_i, y_i(x_i)) = 0,$$

where, given a vector of disciplinary design variables x_i, the analysis, frequently represented by a numerical differential equation solver or simulator, is performed to yield the vector of state variables or responses $y_i(x_i)$. The sets of disciplinary variables x_i are not necessarily disjoint. The disciplinary constraints are usually represented by inequalities

$$c_i(x_i, y_i(x_i)) \geq 0.$$

Once the system objective and variables and the subsystem constraints and variables are identified, the bilevel problem is formed as follows: The constraints of the system problem comprise the "consistency" (or "coupling" or "matching") conditions that are used to drive the discrepancy among the inputs and outputs shared by the subsystems to zero. The values of the constraints are computed by solving the subsystem optimization problems, and the number of *consistency constraints* is related to the number of subsystems and variables shared among the subsystems. The form of the consistency constraints determines a particular implementation of CO.

Let ξ and η represent system-level variables corresponding to inputs and outputs of the subsystems, respectively. Then, given M subsystems, the abbreviated system program is

$$\begin{cases} \min & F(\xi, \eta) \\ \text{s.t.} & G(\xi, \eta) = 0, \end{cases} \tag{1}$$

where

$$G(\xi, \eta) = \begin{pmatrix} g_1(\xi, \eta) \\ \vdots \\ g_M(\xi, \eta) \end{pmatrix}$$

is the set of system consistency constraints obtained by solving lower-level subproblems, each of which is of the form

$$\begin{cases} \min & \frac{1}{2}\left[\|\xi_i - x_i\|^2 + \|\eta_i - y(x_i)\|^2\right] \\ \text{s.t.} & c_i(x_i, y(x_i)) \geq 0, \end{cases} \tag{2}$$

where i is the number of the subsystem.

Thus, the objective of a subsystem optimization problem is always to minimize the discrepancy between the shared variables of the subsystems, in the least squares sense, subject to satisfying the disciplinary constraints, which do not depend explicitly on the system variables passed down to the subsystems as parameters. The subsystems remain feasible during optimization, while *interdisciplinary feasibility* is gradually attained at the system level via the consistency constraints. Maintaining disciplinary feasibility is extremely important from the design perspective, to ensure physical realizability of a design if the process cannot continue to an optimum.

The problem now consists of a set of decoupled subproblems that can be solved independently and in parallel. One instance of the system-level consistency conditions gives rise to the form in which CO is usually presented: namely, the consistency condition is intended to drive to zero the value function of the subproblem (2). That is,

$$g_i(\xi, \eta) = \frac{1}{2}\|\xi - x_*\|^2 + \|\eta - y(x_*)\|^2, \tag{3}$$

where x_* solves the subsystem optimization problem. Another instance of system-level consistency conditions matches the system-level variables with their subsystem counterparts computed in subproblem

$$g_i(\xi, \eta) = (\xi - x_*, \eta - y(x_*)). \tag{4}$$

The behavior of optimization algorithms applied to the all-in-one MDO formulation, where the disciplinary analyses and constraints are united into a single nonlinear program, and CO formulations differ greatly, as the formulations are not equivalent with respect to constraint qualifications or optimality conditions.

In general, value functions are not differentiable, and this may cause difficulties for optimization algorithms applied to the system-level problem. However, under a number of strong assumptions, the constraints are locally differentiable and can usually be computed. Derivatives of the system-level constraints with respect to the system-level design variables are the sensitivities of the minima or the solutions of the subsystem-level optimization problems to parameters. The area of *sensitivity in nonlinear programming* has been studied extensively. Relevant results can be found in [40] and [41]. In particular, under the assumptions of sufficient smoothness, *second order sufficiency*, regularity as constraint qualification, and *strict complementarity slackness*, the basic sensitivity theorem (BST) proves the existence of a unique, local, continuously differentiable solution-multiplier triple for the perturbed problem. Moreover, locally, the set of active constraints remains unchanged and regularity and strict complementarity hold, allowing one to compute derivatives locally. In fact, under a number of assumptions, stronger statements can be made about the differentiability of the value function [30,77].

Under the conditions of BST, local first order derivatives of the consistency constraints (3) have a particularly simple form because, in the case of CO, the constraints of the lower-level problems do not depend on parameters. On the other hand, the first order sensitivities of solutions of the lower-level problem that form the derivatives of the consistency constraints (4), while of closed form, are expensive to compute and involve second order derivatives of the subsystem Lagrangians.

There is another feature of the CO formulation with compatibility constraints (3) that will cause difficulties for nonlinear programming algorithms applied to the system-level problem: *Lagrange multipliers* will almost never exist for the equality constrained system level problem. The nonexistence of Lagrange multipliers is due to the description of the feasible region that causes the Jacobian of the system-level constraints to vanish at a solution. The formulation with compatibility constraints (4) aims to address this problem. However, the computation of derivatives for this formulation is clearly expensive, as it not only involves solving a system of equations, but also requires the computation of second order information for the subsystems. The difficulties are addressed in detail in [7].

In summary, CO is an appealing approach to design optimization; however, the bilevel nature of the problem formulation will cause difficulties for conventional nonlinear programming algorithms applied to the system-level problem. Variations, special algorithms for solution, and alternatives can be found in, e.g., [33,54,55] and continue to appear in later publications [96].

Summary

Multilevel optimization has been an active research field, both in applied mathematics and in engineering design. Many open questions remain, in particular, in the area of practical computational algorithms for bilevel and multilevel problems.

Understanding the behavior of specific, nonlinear programming algorithms applied to the system-level problem of the bilevel or multilevel formulations will present an interesting and difficult area of inquiry, and would benefit from the techniques of *nonsmooth analysis and optimization* [32,36,47], unconventional notions of constraint qualifications [24,25], and optimality [102,103]. To facilitate research and testing in the area of algorithms, one may find automatic bilevel and multilevel problem generators, as well as other sources of multilevel problems, described in [26,72,73].

References

1 Aiyoshi E, Shimizu K (1981) Hierarchical decentralized system and its new solution by a barrier method. IEEE Trans Syst, Man Cybern 11:444–449
2 Alexandrov NM (1996) Multilevel and multiobjective optimization in multidisciplinary design. AIAA Paper no. 96-4122
3 Alexandrov NM, Dennis JE (1995) Multilevel algorithms for nonlinear optimization. In: Borggaard J, Burkardt J, Gunzburger M, Peterson J (eds) Optimal Design and Control. Birkhäuser, Basel, pp 1–22
4 Alexandrov NM, Hussaini MY (eds) (1997) Multidisciplinary design optimization: State of the art. SIAM, Philadelphia
5 Alexandrov NM, Kodiyalam S (1998) Initial results of an MDO method evaluation study. AIAA Paper no. 98-4884
6 Alexandrov NM, Lewis RM (2000) Algorithmic perspectives on problem formulations in MDO. AIAA Paper no. 2000-4719
7 Alexandrov NM, Lewis RM (2002) Analytical and computational aspects of collaborative optimization for multidisciplinary design. AIAA J 40:301–309
8 Alexandrov NM, Lewis RM (2000) Analytical and computational properties of distributed approaches to MDO. AIAA Paper no. 2000-4718
9 Anadalingam G, Friesz TL (1992) Hierarchical optimization: An introduction. Ann Oper Res 34:1–11
10 Antoniou M, Korosec P (2021) Mulitlevel Optimization. In Optimization under Uncertainty with Applications to Aerospace 307–331, Vasile M (ed.) Springer
11 Badhrinath K, Rao JRJ (1994) Bilevel models for optimum designs which are insensitive to perturbations in variables and parameters. Techn Report Univ Houston UH-ME-SDL-94-03
12 Balling RJ, Sobieszczanski-Sobieski J (1994) An algorithm for solving the system-level problem in multilevel optimization. In: Fifth AIAA/USAF/NASA/ISSMO Symp. Multidisciplinary Analysis and Optimization (Panama Beach, Florida, Sept. 7-9, 1994), AIAA Paper no. 94-4333 (1994)
13 Bard JE (1998) Practical Bilevel Optimization Algorithms and Applications. Kluwer Academic Publishers

14 Bard JF (1983) An algorithm for solving the general bilevel programming program. Math Oper Res 8:260–272
15 Bard JF, Falk JE (1982) An explicit solution to the multi-level programming problem. Comput Oper Res 9:77–100
16 Barthelemy J-FM (1988) Engineering applications of heuristic multilevel optimization methods. NASA TM-101504
17 Barthelemy J-FM, Riley MF (1988) Improved multilevel optimization approach for the design of complex engineering systems. AIAA J 26:353–360
18 Basar T, Selbuz H (1979) Closed loop Stackelberg strategies with applications in optimal control of multilevel systems. IEEE Trans Autom Control AC-24:166–178
19 Benders JF (1962) Partitioning procedures for solving mixed variables programming problems. Numerische Math 4:238–252
20 Bialas WF, Karwan MH (1982) On two-level optimization. IEEE Trans Autom Control AC-27:211–214
21 Bracken J, McGill J (1973) Mathematical programs with optimization problems in the constraints. Oper Res 21:37–44
22 Braun RD (1996) Collaborative optimization: An architecture for large-scale distributed design. PhD Thesis Stanford Univ.
23 Braun RD, Moore AA, Kroo IM (1997) Collaborative approach to launch vehicle design. J Spacecraft and Rockets 34:478–486
24 Burke JV (1991) Calmness and exact penalization. SIAM J Control Optim 29:493–497
25 Burke JV (1991) An exact penalization viewpoint of constrained optimization. SIAM J Control Optim 29:968–998
26 Calamai PH, Vicente LN (1993) Generating linear and linear-quadratic bilevel programming problems. SIAM J Sci Comput 14:770–782
27 Candler W, Townsley R (1982) A linear two-level programming problem. Comput Oper Res 9:59–76
28 Chen Y (1994) Bilevel programming problems: Analysis, algorithms and applications. PhD Thesis Univ. Montreal
29 Chidambaram B, Rao JRJ (1994) A study of constraint activity in bilevel models of optimal design. Techn Report Univ Houston UH-ME-SDL-94-01
30 Clarke FH (ed) (1990) Optimization and nonsmooth analysis. SIAM, Philadelphia
31 Danzig GB, Wolfe P (1960) Decomposition principle for linear programming. Oper Res 8:101–111
32 De Luca A, Di Pillo G (1987) Exact augmented Lagrangian approach to multilevel optimization of large-scale systems. Internat J Syst Sci 18:157–176
33 De Miguel W, Murray W (2006) A Local Convergence Analysis of Bilevel Decomposition Algorithms. Optim Eng 7:99–133
34 De Silva AH, McCormick GP (1992) Implicitly defined optimization problems. Ann Oper Res 34:107–124
35 Dennis JE Jr, Schnabel RB (1983) Numerical methods for unconstrained optimization and nonlinear equations. Prentice-Hall, Englewood Cliffs, NJ
36 Dirickx YMI, Jennergren LP (1979) Systems analysis by multilevel methods. Wiley, New York
37 Edmunds TA, Bard JF (1991) Algorithm for nonlinear bilevel mathematical programs. IEEE Trans Syst, Man Cybern 21:83–89
38 Franceschi L, Frasconi P, Salzo S, Grazzi R, and Pontil M. (2018) Bilevel programming for hyperparameter optimization and meta-learning. In J. Dy and A. Krause, eds, Proceedings of the 35th International Conference on Machine Learning, v 80 of Proceedings of Machine Learning Research, pp 1568–1577

39 Falk JE, Liu J (1995) On bilevel programming, Part I: General nonlinear cases. Math Program 70:47–72
40 Fiacco AV (ed) (1983) Introduction to sensitivity and stability analysis in nonlinear programming. Acad. Press, New York
41 Fiacco AV, McCormick GP (eds) (1990) Nonlinear programming, sequential unconstrained minimization techniques. SIAM, Philadelphia
42 Flippo OE (1989) Stability, duality and decomposition in general mathematical programming. PhD Thesis, Erasmus Univ. Rotterdam
43 Flippo OE, Rinnooy Kan AHG (1993) Decomposition in general mathematical programming. Math Program 60:361–382
44 Fortuny-Amat J, McCarl B (1981) A representation of a two-level programming problem. J Oper Res Soc 32:783–792
45 Friesz T, Tobin R, Cho H, Mehta N (1990) Sensitivity analysis based heuristic algorithms for mathematical programs with variational inequality constraints. Math Program 48:265–284
46 Gahutu DWH, Looze DP (1985) Parametric coordination in hierarchical control. Large Scale Systems 8:33–45
47 Gauvin J (1978) The method of parametric decomposition in mathematical programming: the nonconvex case. In: Lemarechal C, Griffin R (eds) Nonsmooth optimization. Pergamon, Oxford, pp 131–149
48 Gauvin J, Dubeau F (1983) Some examples and counterexamples for stability analysis of nonlinear programming problems. In: Fiacco AV (ed) Math. Program. Stud., vol 21. North-Holland, Amsterdam, pp 69–78
49 Geoffrion AM (1972) Generalized Benders decomposition. J Optim Th Appl 10:237–260
50 Goulbeck B, Brdys M, Orr CH, Rance JP (1988) A hierarchical approach to optimized control of water distribution systems: Part I, decomposition. Optimal Control Appl Meth 9:51–61
51 Goulbeck B, Brdys M, Orr CH, Rance JP (1988) A hierarchical approach to optimized control of water distribution systems: part II, lower-level algorithm. Optimal Control Appl Meth 9:109–126
52 Haimes YY (1977) Hierarchical analyses of water resources systems. McGraw-Hill, New York
53 Haimes YY, Tarvainen K, Shima T, Thadathil J (1990) Hierarchical multiobjective analysis of large-scale systems. Hemisphere, Washington, DC
54 Hafka RT, Watson LT (2005) Multidisciplinary Design Optimization Problems with Quasiseparable Subsystems. Optim Eng 6:9–20
55 Hafka RT, Watson LT (2006) Decomposition Theory for Multidisciplinary Design Optimization Problems with Mixed Integer Quasiseparable Subsystems. Optim Eng 7:135–149
56 Ishizuka Y, Aiyoshi E (1992) Double penalty method for bilevel optimization problems. Ann Oper Res 34:73–88
57 Kirsch U (1989) Improved optimum structural design by passive control. Engin with Comput 5:13–22
58 Kodiyalam S (1998) Evaluation of methods for multidisciplinary design optimization (MDO), phase I. NASA Contractor Report
59 Kolstad CD, Lasdon LS (1990) Derivative evaluation and computational experience with large bilevel mathematical programs. J Optim Th Appl 65:485–499
60 Lasdon LS (1970) Optimization theory for large systems. MacMillan, New York
61 Loridan P, Morgan J (1988) Quasiconvex lower level problems and applications in two-level optimization. Techn Report Univ Montreal CRM-1578

62 Mahmoudi MS (1977) Multilevel systems control and applications: A survey. IEEE Trans Syst, Man Cybern SMC-7:125–143
63 Marcotte P (1985) A new algorithm for solving variational inequalities with application to the traffic assignment problem. Math Program 33:339–351
64 Marcotte P, Zhu DL (1996) Exact and inexact penalty methods for the generalized bilevel programming problem. Math Program 74:141–157
65 Mesarović MD, Macko D, Takahara Y (1970) Theory of hierarchical, multilevel, systems. Acad. Press, New York
66 Migdalas A, Pardalos PM, Värbrand P (eds) (1998) Multilevel optimization: Algorithms and applications. Kluwer, Dordrecht
67 Moré JJ (1991) Recent developments in software for trust region methods. In: Bachem A, Grötschel M, Korte B (eds) Mathematical Programming: The State of the Art. Springer, Berlin, pp 266–290
68 Muralidhar R, Rao JRJ, Badhrinath K, Kalagatla A (1995) Multilevel formulations and limit analysis and design of structures with bilateral contact constraints. Techn Report Univ Houston UH-ME-SDL-95-02
69 Nachane DM (1984) Optimization methods in multilevel systems: A methodological survey. Europ J Oper Res 21:25–38
70 Nicholls MG (1995) Aluminium production modelling – a non-linear bi-level programming approach. Oper Res 43:208–218
71 Outrata J (1994) On optimization problems with variational inequality constraints. SIAM J Optim 4:340–357
72 Padula SL, Alexandrov NM, Green LL (1996) MDO test suite at NASA Langley Research Center. In: Proc. Sixth AIAA/NASA/ISSMO Symp. Multidisciplinary Analysis and Optimization, AIAA
73 Padula SL, Young KC (1986) Simulator for multilevel optimization research. NASA Techn Memorandum TM-87751
74 Rao JRJ, Badhrinath K (1996) Solution of multilevel structural design problems using a nonsmooth algorithm. In: Proc. Sixth AIAA/NASA/ISSMO Symp. Multidisciplinary Analysis and Optimization, AIAA, AIAA-96-3986-CR.
75 Rao JRJ, Chidambaram B (1993) Parametric deformation and model optimality in concurrent design. Techn Report Univ Houston UH-ME-SDL-93-01
76 Reddy SY, Fertig KW, Smith DE (Aug. 1996) Constraint management methodology for conceptual design tradeoff studies. In: Proc. DETC'96, ASME Paper 96-DETC/DTM-1228
77 Rockafellar RT (1984) Directional differentiability of the optimal value function in a nonlinear programming problem. Math Program Stud 21:312–226
78 Rogers JL (1996) DeMAID/GA – An enhanced design manager's aid for intelligent decomposition. Proc. Sixth AIAA/NASA/ISSMO Symp. Multidisciplinary Analysis and Optimization,), AIAA Paper 96-4157.
79 Rogers JL (1996) DeMAID/GA user's guide – Design manager's aid for intelligent decomposition with a genetic algorithm. NASA Langley Res Center TM-110241
80 Sato R, Tanaka M, Takeda A (2021) A Gradient Method for Multilevel Optimization. 35th Conference on Neural Information Processing Systems (NeurIPS 2021)
81 Schmitt LA, Chang KJ (1984) A multilevel method for structural synthesis. In: A Collection of Technical Papers: AIAA/ASME/ASCE/AHS 25th Structures, Structural Dynamics and Materials Conf., AIAA
82 Schmitt LA Jr, Mehrinfar M (1982) Multilevel optimum design of structures with fiber-composite stiffened panel components. AiAA J 20(1):138–147

83 Schmitt LA Jr, Ramanathan RK (1978) Multilevel approach to minimum weight design inclusing buckling constraints. AiAA J 16(2):97–104
84 Shimizu K, Aiyoshi E (1981) A new computational method for Stackelberg and min-max problems by use of a penalty method. IEEE Trans Autom Control AC-26:460–466
85 Simaan M, Cruz JB Jr (1973) On the Stackelberg strategy in nonzero-sum games. J Optim Th Appl 11(5):535–555
86 Singh MG, Mahmoud MS, Titli A (1981) A survey of recent developments in hierarchical optimization and control. Proc. IFAC Control Sci. and Techn. 8th Triennial World Congress, Kyoto, Japan, IFAC
87 Sobieski I, Kroo I (1996) Aircraft design using collaborative optimization. In: AIAA paper 96-0715 Presented at the 34th AIAA Aerospace Sci. Meeting, Reno, Nevada, Jan. 15-18, 1996, AIAA
88 Sobieszczanski-Sobieski J (1993) Optimization by decomposition. In: Kamat MP (ed) Structural Optimization: Status and Promise. Progress in Astronautics and Aeronautics, vol 150. AIAA, pp 487–515
89 Sobieszczanski-Sobieski J (1993) Two alternative ways for solving the coordination problem in multilevel optimization. Structural Optim 6:205–215
90 Sobieszczanski-Sobieski J (1996) Multidisciplinary aerospace design optimization: Survey of recent developments. In: Proc. 34-th Aerospace Sci. Meeting and Exhibit, Reno, Nevada, AIAA, AIAA paper 96-0711.
91 Sobieszczanski-Sobieski J, Haftka RT (1997) Multidisciplinary aerospace design optimization: survey of recent developments. Structural Optim 14:1–23
92 Sobieszczanski-Sobieski J, James BB, Dovi AR (1985) Structural optimization by generalized multilevel optimization. AIAA J 23:1775–1782
93 Sobieszczanski-Sobieski J, Morris A, van Tooren M (2015) Multidisciplinary Design Optimization. Wiley
94 Stackelberg H (ed) (1952) The theory of the market economy. Oxford Univ. Press, Oxford
95 Tappeta RV, Renaud JE (1997) Multiobjective collaborative optimization. J Mechanical Design 119:403–411
96 Tosserams S (2008) Distributed Optimization for Systems Design. Eindhoven University of Technology Library Press
97 Vicente LN, Calamai PH (1994) Bilevel and multilevel programming: A bibliographic review. J Global Optim 5:291–306
98 Wagner TC (1993) A general decomposition methodology for optimal system design. PhD Thesis, Univ. Michigan
99 Walsh JL, LaMarsh WJ, Adelman HM (1992) Fully integrated aerodynamic/dynamic/structural optimization of helicopter rotor blades. NASA TM-104226
100 Walsh JL, Young KC, Pritchard JI, Adelman HM, Mantay WR (1995) Integrated aerodynamic/dynamic/structural optimization of helicopter rotor blades using multilevel decomposition. NASA TP-3465
101 Wismer DA (ed) (1971) Optimization methods for large-scale systems. McGraw-Hill, New York
102 Ye JJ, Zhu DL, Zhu QJ (1997) Exact penalization and necessary optimality conditions for generalized bilevel programming problems. SIAM J Optim 7:481–507
103 Zhang R (1994) Problems of hierarchical optimization in finite dimensions. SIAM J Optim 4:521–536