Fuzzy Control Rules in Convex Optimization

Bogdana Pop

Department of Computer Science, Transilvania University of Braşov,

50 Iuliu Maniu, Braşov 500091, Romania

bpop@unitbv.ro

Ioan Dziţac

Department of Business Informatics, Agora University of Oradea,

8 Piața Tineretului, Oradea 410526, Romania

idzitac@univagora.ro

Abstract: The subject of this paper is to describe how fuzzy control rules can be integrated generally in optimization and especially into convex programming and also, to generalize a solving method from the linear case to the convex one. Starting from a conventional algorithm of solving convex optimization problems fuzzy control rules at the testing point for terminating the algorithm are considered here. This choice is useful when the decision-maker could be more comfortable obtaining a solution expressed in terms of satisfaction instead of optimization.

Key words: convex programming, control rules, fuzzy rules.

Bogdana Pop graduated in Mathematics and Computer Science from "Transilvania" University of Braşov, in 1995 and obtained her Ph. D in Mathematics in 2003 from the Romanian Academy. Currently she is an assistant professor at Computer Science Department of "Transilvania" University of Braşov. Her research interests include different aspects of Fuzzy Optimization and Mathematical Fundamentals of Computers

Ioan Dziţac received M. Sc in Mathematics (1977) and Ph. D in Information Sc. (2002) from "Babes-Bolyai" University of Cluj-Napoca. Now, he is associate professor and head of Business Informatics Department at "Agora" University of Oradea, Romania. His current research interests include different aspects of Parallel and Distributed Computing and Economic Mathematics and Informatics.

1. Introduction

The process of optimization can be viewed as a closed-loop control system. The numerical optimization algorithms (which can be viewed as classical controllers) are usually crisply designed for well defined mathematical models. However, in practical optimization problems, it is dealing with decisions in a fuzzy environment. Nowadays fuzzy systems are being developed for both industrial and consumer applications.

Recent and actual list of references related to fuzzy control rules is too wide to be exhaustively mentioned. But, for example, in 2004, in [1], a fuzzy logic-based location management method to reduce paging cost in wireless communication networks is proposed. A partial candidate paging area is selected by fuzzy control rules and the performance of the proposed fuzzy logic-based location management method is evaluated by a simulation. Or, in 2006, a self-organizing fuzzy controller to control a robotic system is proposed and its control performance is evaluated ([6]). Robotic manipulators are multivariable nonlinear dynamic systems. The precise motion control of robotic manipulators is important in improving productivity and quality. In 1994, Pedricz analyses reasoning by analogy in the context of fuzzy controllers as a mechanism for providing conclusions about fuzzy control based upon a level of similarity achieved among facts available in the collection of if-then rules and a new piece of data for which an appropriate control action has to be determined [5].

The mathematical programming is one of the most frequently applied operational research techniques in solving some economical or social real-world problems. Because of their obvious convenient properties, the linear programming problems are for the first time most important to be solved. Fractional programming is the nearest generalization of the linear case and it is better to be used when the decision-maker faces the problem of optimizing various ratios. Convex programming is the next interesting generalized case in optimizations.

In real-world applications certainty, reliability and precision of data is often illusory. Furthermore the optimal solution of the problem depends on less information collected. Being able to deal with vague and imprecise data may greatly contribute to the diffusion and application of mathematical programming [9]. Fuzzy rules in optimization [4] are to satisfy a decision-maker with an early obtained good solution when an optimal solution is hard to be obtained in real time. A common strategy for decision makers is to monitor and keep "tuning" the optimization process in an interactive manner, using their knowledge of the problem and judgment on the information obtained from the previous iterations [2].

The subject of this paper is to describe how fuzzy control rules can be integrated generally in optimization and especially into convex programming and also, to generalize a solving method from the linear case to the convex one. How fuzzy control rules are applied in linear programming is presented in [2]. Part of this paper was presented in [7]. More details, some corrections, generalizations and computational results are included here.

Section 2 draft presents a deterministic optimization algorithm which solves a convex programming problem using a gradient projection method. In Section 3, the deterministic solving method is modified incorporating fuzzy rules in the optimality test. Computational results are inserted in Section 4 and short concluding remarks are made in Section 5.

2. Deterministic Optimization Algorithm

An algorithm for solving a classical optimization problem can be viewed as an interactive process that produced a sequence of points according to a prescribed set of instruction, together with a termination criterion. For a given vector x_i, applying algorithm's instructions a new point x_{i+1} is obtained. Generally it is looking for an algorithm generating a sequence $(x_1, x_2, ..., x_n)$ which converges to a global optimal solution.

In many cases, because of the difficulties of the problem, we may have to be satisfied with less favourable solutions. Then the iterative procedure may stop either if a point belonging to a prefixed set is reached or some prefixed conditions of satisfaction are verified.

Considering the convex programming problem $\min\{f(x) \;|\; x \in X\}$ (CPP) where $X = \{x \in R^n \;|\; Ax \le b\}$ is a convex and bounded set; A is an $m \times n$ constraint matrix; $b \in R^m$; x is an n-dimensional vector of decision variable and $f(x)$ is the convex objective function and using $\bar{x}$, A_r $P_r = I_n - A_r^T\left(A_r A_r^T\right)^{-1} A_r$, φ, $d^T = -P_r\left(\nabla f\left(\bar{x}\right)\right)^T$ (according to [8]) the deterministic optimization algorithm (based on a gradient optimization method) consists in the following steps.

- Step 1. Find $x^0 \in X$ as a feasible solution for Problem (CPP), $k = 0$.
- Step 2. Evaluate $\left(A_r A_r^T\right)^{-1} A_r\left(\nabla f\left(x^k\right)\right)^T \le 0$ and $P_r\left(\nabla f\left(x^k\right)\right)^T = 0$. If both are true then STOP, with x^k as a solution for the initial problem. Otherwise, choose $-P_r\left(\nabla f\left(x^k\right)\right)^T$ (if only first condition is not verified) or $-P_{r-1}\left(\nabla f\left(x^k\right)\right)^T$ (if the second condition is not verified) for the direction $\left(d^k\right)^T$ and go to Step 3.
- Step 3. Compute x^{k+1} [8] and go to Step 2 with $k = k + 1$.

3. Fuzzy Control Rules Based Modified Algorithm

The deterministic algorithm by incorporating fuzzy control rules in the optimality test is reformulated here (generally all tests of the algorithm could be fuzzy interpreted in the same way). We intend to use fuzzy control rules in the termination criterion. For that, we translate the classic inequality $v \le 0$ (and the classic equality $v = 0$) into a fuzzy one $v \bar{\le} 0$ (and $v \bar{=} 0$ respectively) characterised by a membership function $\mu_{\le} : [0,1] \to R$ (and $\mu_{=} : [0,1] \to R$ respectively).

The decision-maker could obtain an optimality degree for each feasible solution using membership function defined above because he could permit low positive values (nonzero values closed to zero respectively) in the right hand side of each optimality conditions.

The membership function $\mu_{\le}$ (and $\mu_{=}$ respectively) is defined to have values closed to 1 for v positive closed to 0 (v nonzero closed to 0 respectively) and to have the value 1 when v is negative ($v = 0$ respectively). Because of that, initial conditions could be transformed in $\mu_{\le}(v) \ge \alpha$ (and $\mu_{=}(v) \ge \alpha$ respectively). Consequently, at any moment, value v fits the optimal condition with degree α. If each involved value v shall verify the property to be low positive (or equal to 0 respectively) in a certain degree α_v then the current solution will be fairly optimal with $\alpha = \min\{\alpha_v \,|\, \forall\, v\}$ degree.

The solving algorithm is described below.

- Step 1. Find $x^0 \in X$ as a feasible solution of Problem (CPP), k=0.
- Step 2. Evaluate $v = \left(A_r A_r^T\right)^{-1} A_r \left(\nabla f\left(x^k\right)\right)^T$, $w = P_r\left(\nabla f\left(x^k\right)\right)^T$ (considering $\alpha_{v_i} = \mu_{\leq}(v_i)$ and $\alpha_{w_j} = \mu_{=}(w_j)$ respectively). If $\alpha = \min\left\{\alpha_{v_i}, \alpha_{w_j} \mid \forall\, i, j\right\} \geq p$ (where p is a threshold predefined by the decision maker) then STOP, with x^k as a solution for the initial problem. Otherwise, choose $-P_r\left(\nabla f\left(x^k\right)\right)^T$ (if and only if $\min\left\{\alpha_{w_j} \mid \forall\, j\right\} \geq p$) or $-P_{r-1}\left(\nabla f\left(x^k\right)\right)^T$ (if and only if $\exists j, \alpha_{w_j} \leq p$) for the direction $\left(d^k\right)^T$ according to the deterministic solving algorithm and go to Step 3.
- Step 3. Compute x^{k+1} [8] and go to Step 2 with $k = k + 1$.

4. Numerical Examples

Let us consider the following convex programming problem.

$$\min\left\{x_1^2 + x_2^2 - 2x_1 - x_2 \mid 4x_1 + 6x_2 \leq 7, x_2 \in [0,1], x_1 \in [0,+\infty)\right\}. \tag{1}$$

Starting with $x^1 = (0,0)$ as initial feasible point, three iterations are needed in order to reach the exact solution. A possible membership function (actually, it should be defined by the decision-maker) to characterise the fuzzy control rules for equalities (inequalities) could be

$$\mu_{=}(x) = \begin{cases} \dfrac{3-x}{3}, & x \in (0,3] \\ 0, & x \in (-\infty,-3) \cup (3,+\infty) \\ \dfrac{3+x}{3}, & x \in [-3,0] \end{cases} \qquad \left(\mu_{\leq}(x) = \begin{cases} 0, & x \in (-\infty,0) \\ \dfrac{6-x}{6}, & x \in [0,6] \\ 0, & x \in (6,+\infty) \end{cases} \right).$$

The following values α^k for corresponding points x^k respectively are obtained.

$$\alpha^1 = \frac{2}{3}\left(x^1 = (0,0)\right), \quad \alpha^2 = \frac{5}{6}\left(x^2 = (1,0)\right), \quad \alpha^3 = 1\left(x^3 = \left(1, \frac{1}{2}\right)\right).$$

The crisp optimal solution is obtained for α = 1, so that the minimal value of the objective function is $f(1,0.5) = -1.25$.

Let us consider another convex programming problem:

$$\min\left\{x_1^2 + x_2^2 - x_1 x_2 - 3x_1 \mid x_1 + x_2 \leq 5, x_1, x_2 \in [0,+\infty)\right\} \tag{2}$$

Thirty six iterations will be needed to reach the crisp optimal solution. The essential iterations (those which improve the α -level of satisfaction of decision-maker) are mentioned in Table 4.1. To implement the deterministic algorithm which was used to solve Problem (2) the 10^{-10} approximation error level was chosen. It means that all inequalities $v \leq 0$ were rewritten $v \leq 10^{-10}$ and all equalities $w = 0$ were rewritten $w \in \left[-10^{-10}, 10^{-10}\right]$ in order to make the algorithm properly to reach its end. Generally, numerical algorithms need such adjustments considering that internal numerical information representation is finite for each standard format and rounds are made in order to keep fixed number of digits in representation.

Analysing numerical results contained in Table 4.1 we can conclude that if the approximation error level was chosen to be 10^{-9} thirty two iterations would have been necessary, if the approximation error level was chosen to be 10^{-8} twenty eight iterations would have been necessary, if the approximation error level was chosen to be 10^{-5} eighteen iterations would have been necessary and so on.

Table 4.1 Numerical results of Problem (2)

Iteration No.	x[1]	x[2]	f(x)
1	0	0	0
2	1.5	0	-2.25
3	1.5	0.75	-2.8125
4	1.875	0.75	-2.953125
5	1.875	0.9375	-2.98828125
6	1.96875	0.9375	-2.9970703125
8	1.9921875	0.984375	-2.99981689453125
15	1.9998779296875	0.99993896484375	-2.99999998882413
16	1.99996948242188	0.99993896484375	-2.99999999720603
18	1.99999237060547	0.999984741210938	-2.99999999982538
22	1.99999952316284	0.999999046325684	-2.99999999999932
26	1.99999997019768	0.999999940395355	-3
28	1.99999999254942	0.999999985098839	-3
32	1.99999999953434	0.999999999068677	-3
36	1.9999999999709	0.999999999941792	-3

5. Concluding Remarks

After a draft presentation of a deterministic optimization algorithm which solves a convex programming problem using a gradient projection method, fuzzy control rules in the optimality test of it were inserted in order to reduce the computation time in obtaining a convenient solution. Computational results were used to describe the modified solving method and also to highlight the importance of it.

Consequently, fuzzy control rules are to be meant as a help solving conventional convex programming problems, presenting its advantage of permitting to obtain solutions fairly optimal instead of solution fully optimal, although optimal solution in a crisp sense can be obtained (the case of $\alpha=1$). As it has been shown, a fairly optimal solution can be obtained quickly. It only depends on the wishes of the decision-maker about the degree for which to obtain that solution.

REFERENCES

1. BAE, I.-H., OH, S.-J., and OLARIU, S., **A Fuzzy Logic-Based Location Management Method for Mobile Networks**, Software Engineering Research and Applications, Springer Berlin / Heidelberg, 2004, pp. 304-319.
2. DELGADO, M., VERDEGAY, J.L., and VILA, M.A., **Fuzzy linear programming: from classical methods to new applications**, Fuzzy Optimization: Recent Advances, DELGADO, M. (eds.), Omnitech, Warsaw and Physica Verlag, Heidelberg, 1994.
3. HERRERA, F., and VERDEGAY, J.L., **Fuzzy Control Rules in Optimization, International Journal of Science & Technology**, Scientia Iranica, 3(1, 2, 3), 1996, pp. 89-96.
4. HERRERA, F., and VERDEGAY, J.L., **Fuzzy Sets and Operations Research. Perspectives**, Fuzzy Sets and Systems 90, 1997, pp. 207-218.
5. PEDRYCZ, W., **Reasoning by Analogy in Fuzzy Controllers**, Fuzzy Control Systems, KANDEL, A. (eds.), CRC Press, New York, 1994.
6. LIAN, R.-J., and LIN, B.-F., **New modifying self-organizing fuzzy controller for robotic motion control**, International Journal Advance Manufacturing and Technology, 29(9-10), 2006, pp. 1074-1087.
7. POP, B., **Algorithm to solving convex programming problems using fuzzy control rules**, Bulletin of the Transilvania University of Braşov, 11(46) - New Series, Series B, 2004, pp. 51-54.
8. POP, B., **Algorithm to solving multiple criteria linear fractional optimization**, An. Univ. Bucuresti, Inform., 1, 2004, pp. 105-114.
9. ZIMMERMANN, H.-J., **Fuzzy Set Theory**, Kluwer Academic Publishers, Boston, 1996.