MATH/AMATH 516, Spring 2009 — Homework 5 (last one ☹) — Due June 9 at 6 PM

You can discuss the problems with each other, but you must write up your answers on your own. Feel free to ask for a hint if you get stuck. [The page/equation numbers are from *Nonlinear Programming*, 2nd edition, 1999, 1st printing. They may differ for the 2nd printing.]

The answer to the * problem is to be turned in jointly with your (randomly chosen) partner.

#1. [SQP method, Exercise 4.3.1 in Bertsekas] Consider a 1-D problem with two inequality constraints: $f(x) = 0$, $g_1(x) = -x$, $g_2(x) = 1 - x^2$. Show that, for all $c > 0$, $x = \frac{1-\sqrt{5}}{2}$ and $x = 0$ are stationary points of $f + cP$, where $P(x) = \max\{0, g_1(x), g_2(x)\}$, but are infeasible for the constrained problem. Plot $P(x)$ and discuss the behavior of the SQP/linearization method for this problem. Your discussion should answer questions such as: Does $\{x^k\}$ converge or diverge? If $\{x^k\}$ converges, which point does it converge to? Your answer may depend on x^0, as well as H^k and α^k.

#2. Exercise 4.3.4. [Instead of following the book's hint, this can be argued more simply by using $0 = \nabla f(x^*) + \sum_{j \in I(x^*)} \mu_j^* \nabla g_j(x^*)$ and showing that $\sum_{j \in I(x^*)} \mu_j^* \nabla g_j(x^*)'d \le c \max_{j \in J(x^*)} \nabla g_j(x^*)'d$ for any $d \in \Re^n$.]

#3.* [Duality gap bound for integer program] A well-known NP-hard combinatorial optimization problem is the knapsack problem, whereby we are given object i with weight $w_i > 0$ and value $v_i > 0$, $i = 1, ..., n$, as well as a knapsack that can carry total weight of $W > 0$. We wish to select objects to carry by the knapsack whose total value is maximum. This can be formulated as the integer program (IP):

$$\min -\sum_{i=1}^{n} v_i x_i \quad \text{s.t.} \quad \sum_{i=1}^{n} w_i x_i \le W,\ x_i \in \{0,1\},\ i = 1, ..., n.$$

Let v_{IP} denote the optimal value of the IP.

(a) Consider the linear program (LP) obtained by relaxing the constraint "$x_i \in \{0,1\}$" to "$0 \le x_i \le 1$". Let v_{LP} denote the optimal value of this LP. Show that

$$0 \le v_{\text{IP}} - v_{\text{LP}} \le \max_{i=1,\ldots,n} v_i.$$

Hint: Round down an optimal solution of the LP to obtain a feasible solution of the IP. You might find it easier to first order the objects so that $v_1/w_1 \ge v_2/w_2 \ge ... \ge v_n/w_n$.

(b) Show that the IP and the LP have the same dual problem. Thus, by LP strong duality, $v_{\text{IPdual}} = v_{\text{LP}}$, where v_{IPdual} is the optimal value of the IP dual problem. This and (a) show that the duality gap between the IP and its dual is bounded by $\max_{i=1,\ldots,n} v_i$.

#4. [Computing problem.] On the class webpage you will find a Matlab program that implements the quadratic penalty method, with f_c minimized by the steepest descent method using the Armijo stepsize rule. The method terminates when $\max\{\|\nabla f_c(x)\|, \|h(x)\|\} \le 0.01$. The function and gradient routines for the example $n = 4$, $m = 2$, $f(x) = -x_1$, $h_1(x) = -x_1^3 + x_2 - x_3^2$, $h_2(x) = x_1^2 - x_2 - x_4^2$ are also shown. (To run this, save the program and the four routines in five files named quadpen.m, f.m, gradf.m, h.m, gradh.m and, assuming your computer system has Matlab, start Matlab and type quadpen) By modifying this program or writing from scratch in your favorite computer language, implement the *method of multipliers*, with L_c minimized by the Polak-Ribiere conjugate gradient method (or BFGS method) using the Armijo stepsize rule. Run your program on the above example with initial point $x^0 = (2,2,2,2)'$, $\lambda^0 = (0,0)'$, $c^0 = 1$. Does your program "improve" on the quadratic penalty method?