# MMA and GCMMA, versions September 2007

Krister Svanberg
Optimization and Systems Theory,
KTH, Stockholm, Sweden.
krille@math.kth.se

This note describes the algorithms used in the author's latest implementations of MMA and GCMMA. The first versions of these methods were published in [1] and [2].

## 1 Considered optimization problem

Throughout this note, optimization problems of the following form are considered, where the variables are $\mathbf{x} = (x_1, \ldots, x_n)^T \in I\!R^n$, $\mathbf{y} = (y_1, \ldots, y_m)^T \in I\!R^m$, and $z \in I\!R$.

$$\begin{array}{rl} \text{minimize} & f_0(\mathbf{x}) + a_0 z + \sum_{i=1}^{m} (c_i y_i + \frac{1}{2} d_i y_i^2) \\ \text{subject to} & f_i(\mathbf{x}) - a_i z - y_i \le 0, \quad i = 1, \ldots, m \\ & \mathbf{x} \in X, \;\; \mathbf{y} \ge \mathbf{0}, \;\; z \ge 0. \end{array} \tag{1.1}$$

Here, $X = \{\mathbf{x} \in I\!R^n \mid x_j^{\min} \le x_j \le x_j^{\max}, \; j = 1, \ldots, n\}$, where $x_j^{\min}$ and $x_j^{\max}$ are given real numbers which satisfy $x_j^{\min} < x_j^{\max}$ for all $j$, $f_0, f_1, \ldots, f_m$ are given, continuously differentiable, real-valued functions on $X$, $a_0$, $a_i$, $c_i$ and $d_i$ are given real numbers which satisfy $a_0 > 0$, $a_i \ge 0$, $c_i \ge 0$, $d_i \ge 0$ and $c_i + d_i > 0$ for all $i$, and also $a_i c_i > a_0$ for all $i$ with $a_i > 0$.

In (1.1), the "natural" optimization variables are $x_1, \ldots, x_n$, while $y_1, \ldots, y_m$ and $z$ are "artificial" optimization variables which should make it easier for the user to formulate and solve certain subclasses of problems, like least squares problems and minmax problems.

As a first example, assume that the user wants to solve a problem on the following "standard" form for nonlinear programming.

$$\begin{array}{rl} \text{minimize} & f_0(\mathbf{x}) \\ \text{subject to} & f_i(\mathbf{x}) \le 0, \quad i = 1, \ldots, m \\ & \mathbf{x} \in X, \end{array} \tag{1.2}$$

where $f_0, f_1, \ldots, f_m$ are given differentiable functions and $X$ is as above. To make problem (1.1) (almost) equivalent to this problem (1.2), first let $a_0 = 1$ and $a_i = 0$ for all $i > 0$. Then $z = 0$ in any optimal solution of (1.1). Further, for each $i$, let $d_i = 1$ and $c_i =$ "a large number", so that the variables $y_i$ become "expensive". Then typically $\mathbf{y} = \mathbf{0}$ in any optimal

solution of (1.1), and the corresponding $\mathbf{x}$ is an optimal solution of (1.2). It should be noted that the problem (1.1) always has feasible solutions, and in fact also at least one optimal solution. This holds even if the user's problem (1.2) does not have any feasible solutions, in which case some $y_i > 0$ in the optimal solution of (1.1).

As a second example, assume that the user wants to solve a "min-max" problem on the form

$$\begin{array}{ll} \text{minimize} & \max\limits_{i=1,..,p} \{h_i(\mathbf{x})\} \\ \text{subject to} & g_i(\mathbf{x}) \leq 0, \qquad i = 1, \ldots, q \\ & \mathbf{x} \in X. \end{array} \tag{1.3}$$

where $h_i$ and $g_i$ are given differentiable functions and $X$ is as above. For each given $\mathbf{x} \in X$, the value of the objective function in problem (1.3) is the largest of the $p$ real numbers $h_1(\mathbf{x}), \ldots, h_p(\mathbf{x})$. This kind of problems appear e.g. when the largest stress in a structure should be minimized subject to various additional constraints. Problem (1.3) may equivalently be written on the following form with variables $\mathbf{x} \in I\!R^n$ and $z \in I\!R$:

$$\begin{array}{ll} \text{minimize} & z \\ \text{subject to} & z \geq h_i(\mathbf{x}), \quad i = 1, \ldots, p \\ & g_i(\mathbf{x}) \leq 0, \quad i = 1, \ldots, q \\ & \mathbf{x} \in X. \end{array} \tag{1.4}$$

To make problem (1.1) (almost) equivalent to this problem (1.4), let $m = p + q$, $f_0(\mathbf{x}) = 0$, $f_i(\mathbf{x}) = h_i(\mathbf{x})$ for $i = 1, \ldots, p$, $f_{p+i}(\mathbf{x}) = g_i(\mathbf{x})$ for $i = 1, \ldots, q$, $a_0 = a_1 = \cdots = a_p = 1$, $a_{p+1} = \cdots = a_{p+q} = 0$, $d_1 = \cdots = d_m = 1$, $c_1 = \cdots = c_m =$ "a large number".

As a third example, assume that the user wants to solve a constrained least squares problem on the form

$$\begin{array}{ll} \text{minimize} & \frac{1}{2} \sum\limits_{i=1}^{p} (h_i(\mathbf{x}))^2 \\ \text{subject to} & g_i(\mathbf{x}) \leq 0, \qquad i = 1, \ldots, q \\ & \mathbf{x} \in X. \end{array} \tag{1.5}$$

where $h_i$ and $g_i$ are given differentiable functions and $X$ is as above. Problem (1.5) may equivalently be written on the following form with variables $\mathbf{x} \in I\!R^n$ and $y_1, \ldots, y_{2p} \in I\!R$:

$$\begin{array}{lll} \text{minimize} & \frac{1}{2} \sum\limits_{i=1}^{p} (y_i^2 + y_{p+i}^2) & \\ \text{subject to} & y_i \geq h_i(\mathbf{x}), & i = 1, \ldots, p \\ & y_{p+i} \geq -h_i(\mathbf{x}), & i = 1, \ldots, p \\ & g_i(\mathbf{x}) \leq 0, & i = 1, \ldots, q \\ & y_i \geq 0 \text{ and } y_{p+i} \geq 0, & i = 1, \ldots, p \\ & \mathbf{x} \in X. & \end{array} \tag{1.6}$$

To make problem (1.1) (almost) equivalent to this problem (1.6), let $m = 2p + q$, $f_0(\mathbf{x}) = 0$, $f_i(\mathbf{x}) = h_i(\mathbf{x})$ for $i = 1, \ldots, p$, $f_{p+i}(\mathbf{x}) = -h_i(\mathbf{x})$ for $i = 1, \ldots, p$, $f_{2p+i}(\mathbf{x}) = g_i(\mathbf{x})$ for $i = 1, \ldots, q$, $a_0 = 1$, $a_1 = \cdots = a_m = 0$, $d_1 = \cdots = d_m = 1$, $c_1 = \cdots = c_{2p} = 0$, $c_{2p+1} = \cdots = c_{2p+q} =$ "a large number".

## 2 Some practical considerations

In many applications, the constraints are on the form $g_i(\mathbf{x}) \leq g_i^{\max}$, where $g_i(\mathbf{x})$ stands for e.g. a certain stress, while $g_i^{\max}$ is the largest permitted value on this stress. This means that $f_i(\mathbf{x}) = g_i(\mathbf{x}) - g_i^{\max}$ (in (1.1) as well as in (1.2)). The user should then preferably scale the constraints in such a way that $1 \leq g_i^{\max} \leq 100$ for each $i$ (and not $g_i^{\max} = 10^{10}$). The objective function $f_0(\mathbf{x})$ should preferably be scaled such that $1 \leq f_0(\mathbf{x}) \leq 100$ for reasonable values on the variables. The variables $x_j$ should preferably be scaled such that $0.1 \leq x_j^{\max} - x_j^{\min} \leq 100$, for all $j$.

Concerning the "large numbers" on the coefficients $c_i$ (mentioned above), the user should for numerical reasons try to avoid "extremely large" values on these coefficients (like $10^{10}$). It is better to start with "reasonably large" values and then, if it turns out that not all $y_i = 0$ in the optimal solution of (1.1), increase the corresponding values of $c_i$ by e.g. a factor 100 and solve the problem again, etc. If the functions and the variables have been scaled according to above, then "resonably large" values on the parameters $c_i$ could be, say, $c_i = 1000$ or 10000.

Finally, concerning the simple bound constraints $x_j^{\min} \leq x_j \leq x_j^{\max}$, it may sometimes be the case that some variables $x_j$ do not have any prescribed upper and/or lower bounds. In that case, it is in practice always possible to choose "artificial" bounds $x_j^{\min}$ and $x_j^{\max}$ such that every realistic solution $\mathbf{x}$ satisfies the corresponding bound constraints. The user should then preferably avoid choosing $x_j^{\max} - x_j^{\min}$ unnecessarily large. It is better to try some reasonable bounds and then, if it turns out that some variable $x_j$ becomes equal to such an "artificial" bound in the optimal solution of (1.1), change this bound and solve the problem again (starting from the recently obtained solution), etc.

## 3 The ordinary MMA

MMA is a method for solving problems on the form (1.1), using the following approach: In each iteration, the current iteration point $(\mathbf{x}^{(k)}, \mathbf{y}^{(k)}, z^{(k)})$ is given. Then an approximating subproblem, in which the functions $f_i(\mathbf{x})$ are replaced by certain convex functions $\tilde{f}_i^{(k)}(\mathbf{x})$, is generated. The choice of these approximating functions is based mainly on gradient information at the current iteration point, but also on some parameters $u_j^{(k)}$ and $l_j^{(k)}$ ("moving asymptotes") which are updated in each iteration based on information from previous iteration points. The subproblem is solved, and the unique optimal solution becomes the next iteration point $(\mathbf{x}^{(k+1)}, \mathbf{y}^{(k+1)}, z^{(k+1)})$. Then a new subproblem is generated, etc. The MMA subproblem looks as follows:

$$\begin{array}{rl} \text{minimize} & \tilde{f}_0^{(k)}(\mathbf{x}) + a_0 z + \sum_{i=1}^{m} (c_i y_i + \frac{1}{2} d_i y_i^2) \\ \text{subject to} & \tilde{f}_i^{(k)}(\mathbf{x}) - a_i z - y_i \leq 0, \quad i = 1, \ldots, m \\ & \mathbf{x} \in X^{(k)}, \quad \mathbf{y} \geq \mathbf{0}, \quad z \geq 0, \end{array} \tag{3.1}$$

where $X^{(k)} = \{\mathbf{x} \in X \mid 0.9\,l_j^{(k)} + 0.1x_j^{(k)} \le x_j \le 0.9\,u_j^{(k)} + 0.1x_j^{(k)},\ j = 1, \ldots, n\}$, and where the approximating functions $\tilde{f}_i^{(k)}(\mathbf{x})$ are chosen as

$$\tilde{f}_i^{(k)}(\mathbf{x}) = \sum_{j=1}^{n} \left( \frac{p_{ij}^{(k)}}{u_j^{(k)} - x_j} + \frac{q_{ij}^{(k)}}{x_j - l_j^{(k)}} \right) + r_i^{(k)}, \quad i = 0, 1, \ldots, m. \tag{3.2}$$

Here,

$$p_{ij}^{(k)} = (u_j^{(k)} - x_j^{(k)})^2 \left( 1.001 \left( \frac{\partial f_i}{\partial x_j}(\mathbf{x}^{(k)}) \right)^+ + 0.001 \left( \frac{\partial f_i}{\partial x_j}(\mathbf{x}^{(k)}) \right)^- + \frac{\rho_i}{x_j^{\max} - x_j^{\min}} \right), \tag{3.3}$$

$$q_{ij}^{(k)} = (x_j^{(k)} - l_j^{(k)})^2 \left( 0.001 \left( \frac{\partial f_i}{\partial x_j}(\mathbf{x}^{(k)}) \right)^+ + 1.001 \left( \frac{\partial f_i}{\partial x_j}(\mathbf{x}^{(k)}) \right)^- + \frac{\rho_i}{x_j^{\max} - x_j^{\min}} \right), \tag{3.4}$$

$$r_i^{(k)} = f_i(\mathbf{x}^{(k)}) - \sum_{j=1}^{n} \left( \frac{p_{ij}^{(k)}}{u_j^{(k)} - x_j^{(k)}} + \frac{q_{ij}^{(k)}}{x_j^{(k)} - l_j^{(k)}} \right), \tag{3.5}$$

where $\left( \frac{\partial f_i}{\partial x_j}(\mathbf{x}^{(k)}) \right)^+$ denotes the largest of the two numbers $0$ and $\frac{\partial f_i}{\partial x_j}(\mathbf{x}^{(k)})$,
while $\left( \frac{\partial f_i}{\partial x_j}(\mathbf{x}^{(k)}) \right)^-$ denotes the largest of the two numbers $0$ and $-\frac{\partial f_i}{\partial x_j}(\mathbf{x}^{(k)})$.
The default rules for updating the lower asymptotes $l_j^{(k)}$ and the upper asymptotes $u_j^{(k)}$ are as follows. The first two iterations, when $k = 1$ and $k = 2$,

$$\begin{array}{rcl} l_j^{(k)} & = & x_j^{(k)} - 0.5(x_j^{\max} - x_j^{\min}), \\ u_j^{(k)} & = & x_j^{(k)} + 0.5(x_j^{\max} - x_j^{\min}). \end{array} \tag{3.6}$$

In later iterations, when $k \ge 3$,

$$\begin{array}{rcl} l_j^{(k)} & = & x_j^{(k)} - \gamma_j^{(k)}(x_j^{(k-1)} - l_j^{(k-1)}), \\ u_j^{(k)} & = & x_j^{(k)} + \gamma_j^{(k)}(u_j^{(k-1)} - x_j^{(k-1)}), \end{array} \tag{3.7}$$

where

$$\gamma_j^{(k)} = \begin{cases} 0.7 & \text{if} \quad (x_j^{(k)} - x_j^{(k-1)})(x_j^{(k-1)} - x_j^{(k-2)}) < 0, \\ 1.2 & \text{if} \quad (x_j^{(k)} - x_j^{(k-1)})(x_j^{(k-1)} - x_j^{(k-2)}) > 0, \\ 1 & \text{if} \quad (x_j^{(k)} - x_j^{(k-1)})(x_j^{(k-1)} - x_j^{(k-2)}) = 0, \end{cases} \tag{3.8}$$

provided that this leads to values that satisfy

$$\begin{array}{rcl} l_j^{(k)} & \le & x_j^{(k)} - 0.01(x_j^{\max} - x_j^{\min}), \\ l_j^{(k)} & \ge & x_j^{(k)} - 10(x_j^{\max} - x_j^{\min}), \\ u_j^{(k)} & \ge & x_j^{(k)} + 0.01(x_j^{\max} - x_j^{\min}), \\ u_j^{(k)} & \le & x_j^{(k)} + 10(x_j^{\max} - x_j^{\min}). \end{array} \tag{3.9}$$

If any of these bounds is violated, the corresponding $l_j^{(k)}$ or $u_j^{(k)}$ is put to the right hand side of the violated inequality.

The default values of the parameters $\rho_i$ in (3.3) and (3.4) are $\rho_i = 10^{-5}$ for all $i = 0, 1, \ldots, m$.

It follows from the formulas (3.2)–(3.5) that the functions $\tilde{f}_i^{(k)}$ are always first order approximations of the original functions $f_i$ at the current iteration point, i.e.

$$\tilde{f}_i^{(k)}(\mathbf{x}^{(k)}) = f_i(\mathbf{x}^{(k)}) \quad \text{and} \quad \frac{\partial \tilde{f}_i^{(k)}}{\partial x_j}(\mathbf{x}^{(k)}) = \frac{\partial f_i}{\partial x_j}(\mathbf{x}^{(k)}). \tag{3.10}$$

Moreover, since all $\rho_i > 0$, the approximating functions $\tilde{f}_i^{(k)}$ are strictly convex. This implies that there is always a unique optimal solution of the MMA subproblem.

# 4 GCMMA – the globally convergent version of MMA

The globally convergent version of MMA, from now on called GCMMA, for solving problems of the form (1.1) consists of "outer" and "inner" iterations. The index $k$ is used to denote the outer iteration number, while the index $\nu$ is used to denote the inner iteration number. Within each outer iteration, there may be zero, one, or several inner iterations. The double index $(k, \nu)$ is used to denote the $\nu$:th inner iteration within the $k$:th outer iteration.

The first iteration point is obtained by first chosing $\mathbf{x}^{(1)} \in X$ and then chosing $\mathbf{y}^{(1)}$ and $z^{(1)}$ such that $(\mathbf{x}^{(1)}, \mathbf{y}^{(1)}, z^{(1)})$ becomes a feasible solution of (1.1). This is easy. An outer iteration of the method, going from the $k$:th iteration point $(\mathbf{x}^{(k)}, \mathbf{y}^{(k)}, z^{(k)})$ to the $(k+1)$:th iteration point $(\mathbf{x}^{(k+1)}, \mathbf{y}^{(k+1)}, z^{(k+1)})$, can be described as follows:

Given $(\mathbf{x}^{(k)}, \mathbf{y}^{(k)}, z^{(k)})$, an approximating subproblem is generated and solved. In this subproblem, the functions $f_i(\mathbf{x})$ are replaced by certain convex functions $\tilde{f}_i^{(k,0)}(\mathbf{x})$. The optimal solution of this subproblem is denoted $(\hat{\mathbf{x}}^{(k,0)}, \hat{\mathbf{y}}^{(k,0)}, \hat{z}^{(k,0)})$. If $\tilde{f}_i^{(k,0)}(\hat{\mathbf{x}}^{(k,0)}) \geq f_i(\hat{\mathbf{x}}^{(k,0)})$, for all $i = 0, 1, \ldots, m$, the next iteration point becomes $(\mathbf{x}^{(k+1)}, \mathbf{y}^{(k+1)}, z^{(k+1)}) = (\hat{\mathbf{x}}^{(k,0)}, \hat{\mathbf{y}}^{(k,0)}, \hat{z}^{(k,0)})$, and the outer iteration is completed (without any inner iterations needed). Otherwise, an inner iteration is made, which means that a new subproblem is generated and solved at $\mathbf{x}^{(k)}$, with new approximating functions $\tilde{f}_i^{(k,1)}(\mathbf{x})$ which are more conservative than $\tilde{f}_i^{(k,0)}(\mathbf{x})$ for those indices $i$ for which the above inequality was violated. The optimal solution of this new subproblem is denoted $(\hat{\mathbf{x}}^{(k,1)}, \hat{\mathbf{y}}^{(k,1)}, \hat{z}^{(k,1)})$. If $\tilde{f}_i^{(k,1)}(\hat{\mathbf{x}}^{(k,1)}) \geq f_i(\hat{\mathbf{x}}^{(k,1)})$, for all $i = 0, 1, \ldots, m$, the next iteration point becomes $(\mathbf{x}^{(k+1)}, \mathbf{y}^{(k+1)}, z^{(k+1)}) = (\hat{\mathbf{x}}^{(k,1)}, \hat{\mathbf{y}}^{(k,1)}, \hat{z}^{(k,1)})$, and the outer iteration is completed (with one inner iterations needed). Otherwise, another inner iteration is made, which means that a new subproblem is generated and solved at $\mathbf{x}^{(k)}$, with new approximating functions $\tilde{f}_i^{(k,2)}(\mathbf{x})$, etc. These inner iterations are repeated until $\tilde{f}_i^{(k,\nu)}(\hat{\mathbf{x}}^{(k,\nu)}) \geq f_i(\hat{\mathbf{x}}^{(k,\nu)})$ for all $i = 0, 1, \ldots, m$, which always happens after a finite (usually small) number of inner iterations. Then the next iteration point becomes $(\mathbf{x}^{(k+1)}, \mathbf{y}^{(k+1)}, z^{(k+1)}) = (\hat{\mathbf{x}}^{(k,\nu)}, \hat{\mathbf{y}}^{(k,\nu)}, \hat{z}^{(k,\nu)})$, and the outer iteration is completed (with $\nu$ inner iterations needed).

It should be noted that in each inner iteration, there is no need to recalculate the gradients $\nabla f_i(\mathbf{x}^{(k)})$, since $\mathbf{x}^{(k)}$ has not changed. Gradients of the original functions $f_i$ are calculated only once in each outer iteration. This is an important note since the calculation of gradients is typically the most time consuming part in structural optimization.

The GCMMA subproblem looks as follows, for $k \in \{1, 2, 3, \ldots\}$ and $\nu \in \{0, 1, 2, \ldots\}$:

$$\begin{aligned} \text{minimize} \quad & \tilde{f}_0^{(k,\nu)}(\mathbf{x}) + a_0 z + \sum_{i=1}^{m} (c_i y_i + \tfrac{1}{2} d_i y_i^2) \\ \text{subject to} \quad & \tilde{f}_i^{(k,\nu)}(\mathbf{x}) - a_i z - y_i \le 0, \quad i = 1, \ldots, m \\ & \mathbf{x} \in X^{(k)}, \quad \mathbf{y} \ge \mathbf{0}, \quad z \ge 0, \end{aligned} \tag{4.1}$$

where $X^{(k)} = \{\mathbf{x} \in X \mid 0.9\, l_j^{(k)} + 0.1 x_j^{(k)} \le x_j \le 0.9\, u_j^{(k)} + 0.1 x_j^{(k)},\ j = 1, \ldots, n\}$, and where the approximating functions $\tilde{f}_i^{(k,\nu)}(\mathbf{x})$ are chosen as

$$\tilde{f}_i^{(k,\nu)}(\mathbf{x}) = \sum_{j=1}^{n} \left( \frac{p_{ij}^{(k,\nu)}}{u_j^{(k)} - x_j} + \frac{q_{ij}^{(k,\nu)}}{x_j - l_j^{(k)}} \right) + r_i^{(k,\nu)}, \quad i = 0, 1, \ldots, m\,. \tag{4.2}$$

Here,

$$p_{ij}^{(k,\nu)} = (u_j^{(k)} - x_j^{(k)})^2 \left( 1.001 \left( \frac{\partial f_i}{\partial x_j}(\mathbf{x}^{(k)}) \right)^+ + 0.001 \left( \frac{\partial f_i}{\partial x_j}(\mathbf{x}^{(k)}) \right)^- + \frac{\rho_i^{(k,\nu)}}{x_j^{\max} - x_j^{\min}} \right), \tag{4.3}$$

$$q_{ij}^{(k,\nu)} = (x_j^{(k)} - l_j^{(k)})^2 \left( 0.001 \left( \frac{\partial f_i}{\partial x_j}(\mathbf{x}^{(k)}) \right)^+ + 1.001 \left( \frac{\partial f_i}{\partial x_j}(\mathbf{x}^{(k)}) \right)^- + \frac{\rho_i^{(k,\nu)}}{x_j^{\max} - x_j^{\min}} \right), \tag{4.4}$$

$$r_i^{(k,\nu)} = f_i(\mathbf{x}^{(k)}) - \sum_{j=1}^{n} \left( \frac{p_{ij}^{(k,\nu)}}{u_j^{(k)} - x_j^{(k)}} + \frac{q_{ij}^{(k,\nu)}}{x_j^{(k)} - l_j^{(k)}} \right). \tag{4.5}$$

The asymptotes $l_j^{(k)}$ and $u_j^{(k)}$ are updated as in the original MMA. The formulas (3.6)–(3.9) still hold. The parameters $\rho_i^{(k,\nu)}$ in (4.3) and (4.4) are strictly positive and updated according to below. Within a given outer iteration $k$, the only differences between two inner iterations are the values of some of these parameters. In the beginning of each outer iteration, when $\nu = 0$, the following default values are used:

$$\rho_i^{(k,0)} = \frac{0.1}{n} \sum_{j=1}^{n} \left| \frac{\partial f_i}{\partial x_j}(\mathbf{x}^{(k)}) \right| (x_j^{\max} - x_j^{\min}), \quad \text{for } i = 0, 1, .., m. \tag{4.6}$$

If any of the right hand sides in (4.6) is $< 10^{-6}$ then the corresponding $\rho_i^{(k,0)}$ is set to $10^{-6}$.

In each new inner iteration, the updating of $\rho_i^{(k,\nu)}$ is based on the solution of the most recent subproblem. Note that $\tilde{f}_i^{(k,\nu)}(\mathbf{x})$ may be written on the form:

$$\tilde{f}_i^{(k,\nu)}(\mathbf{x}) = h_i^{(k)}(\mathbf{x}) + \rho_i^{(k,\nu)} d^{(k)}(\mathbf{x}),$$

where $h_i^{(k)}(\mathbf{x})$ and $d^{(k)}(\mathbf{x})$ do not depend on $\rho_i^{(k,\nu)}$. Some calculations give that

$$d^{(k)}(\mathbf{x}) = \sum_{j=1}^{n} \frac{(u_j^{(k)} - l_j^{(k)})(x_j - x_j^{(k)})^2}{(u_j^{(k)} - x_j)(x_j - l_j^{(k)})(x_j^{\max} - x_j^{\min})}\,. \tag{4.7}$$

Now, let

$$\delta_i^{(k,\nu)} = \frac{f_i(\hat{\mathbf{x}}^{(k,\nu)}) - \tilde{f}_i^{(k,\nu)}(\hat{\mathbf{x}}^{(k,\nu)})}{d^{(k)}(\hat{\mathbf{x}}^{(k,\nu)})} \ . \tag{4.8}$$

Then $h_i^{(k)}(\hat{\mathbf{x}}^{(k,\nu)}) + (\rho_i^{(k,\nu)} + \delta_i^{(k,\nu)}) d^{(k)}(\hat{\mathbf{x}}^{(k,\nu)}) = f_i(\hat{\mathbf{x}}^{(k,\nu)})$, which shows that $\rho_i^{(k,\nu)} + \delta_i^{(k,\nu)}$ might be a natural value of $\rho_i^{(k,\nu+1)}$. In order to get a globally convergent method, this natural value is modified as follows.

$$\begin{aligned} \rho_i^{(k,\nu+1)} &= \min\{\, 1.1\,(\rho_i^{(k,\nu)} + \delta_i^{(k,\nu)}) \,,\, 10\rho_i^{(k,\nu)} \} && \text{if } \delta_i^{(k,\nu)} > 0, \\ \rho_i^{(k,\nu+1)} &= \rho_i^{(k,\nu)} && \text{if } \delta_i^{(k,\nu)} \le 0. \end{aligned} \tag{4.9}$$

It follows from the formulas (4.2)–(4.5) that the functions $\tilde{f}_i^{(k,\nu)}$ are always first order approximations of the original functions $f_i$ at the current iteration point, i.e.

$$\tilde{f}_i^{(k,\nu)}(\mathbf{x}^{(k)}) = f_i(\mathbf{x}^{(k)}) \ \text{ and } \ \frac{\partial \tilde{f}_i^{(k,\nu)}}{\partial x_j}(\mathbf{x}^{(k)}) = \frac{\partial f_i}{\partial x_j}(\mathbf{x}^{(k)}). \tag{4.10}$$

Since the parameters $\rho_i^{(k,\nu)}$ are always strictly positive, the functions $\tilde{f}_i^{(k,\nu)}$ are strictly convex. This implies that there is always a unique optimal solution of the GCMMA subproblem.

There are at least two approaches for solving the subproblems in MMA and in GCMMA, the "dual approach" and the "primal-dual interior-point approach".

The dual approach is based on Lagrangean relaxation duality. The dual problem corresponding to the MMA subproblem is a maximization problem with concave objective function and no other constraints than non-negativity requirements on the (dual) variables. This dual problem can be solved by a modified Newton method, and then the optimal dual solution can be translated to a corresponding optimal solution of the primal subproblem. We have implemented this approach in Fortran.

In the primal-dual interior-point approach, a sequence of relaxed KKT conditions are solved by Newton's method. We have implemented this approach in Matlab, since all the required calculations are most naturally carried out on a matrix and vector level.

## 5 A challenging test problem - the snake

Let $\ell$ be a given positive integer and let $\delta$ be a given "small" positive real number.
For $i = 1, \ldots, \ell$, let $\alpha_i = \dfrac{(3i-2\ell)\pi}{6\ell}$, $g_i(\mathbf{x}) = \dfrac{x_i^2 + x_{\ell+i}^2 - 1}{\delta}$ and $h_i(\mathbf{x}) = \dfrac{x_{2\ell+i} - 2x_i x_{\ell+i}}{\delta}$.
Then consider the following problem in the variables $\mathbf{x} = (x_1, \ldots, x_{3\ell})^T$:

$$\begin{aligned} \text{minimize} \quad & \sum_{i=1}^{\ell} (x_i \cos\alpha_i + x_{\ell+i} \sin\alpha_i - 0.1 x_{2\ell+i}) \\ \text{subject to} \quad & \sum_{i=1}^{\ell} (x_i^2 + x_{\ell+i}^2) \le \ell \\ & -2 \le g_i(\mathbf{x}) + g_i(\mathbf{x})^7 \le 2, \qquad i = 1, \ldots, \ell \\ & -2 \le h_i(\mathbf{x}) + h_i(\mathbf{x})^7 \le 2, \qquad i = 1, \ldots, \ell \\ & -2 \le x_j \le 2, \qquad j = 1, \ldots, 3\ell \end{aligned} \tag{5.1}$$

If $\delta \ll 1$, the constraints involving $g_i(\mathbf{x})$ imply that $x_i^2 + x_{\ell+i}^2 \approx 1$ while the constraints involving $h_i(\mathbf{x})$ imply that $x_{2\ell+i} \approx 2x_i x_{\ell+i}$. This means that $(x_i, x_{\ell+i}, x_{2\ell+i}) \approx (\cos\varphi_i, \sin\varphi_i, \sin 2\varphi_i)$ for some $\varphi_i$. The feasible set may therefore be interpreted as the Cartesian product of $\ell$ "snakes", each living in three dimensions and with thickness of the order $\delta$. The $\ell$ different three-dimensional problems are all connected through the first constraint (involving the sum over $i$) which is active in the optimal solution. Finally, the power 7 in the terms $g_i(\mathbf{x})^7$ and $h_i(\mathbf{x})^7$ implies that the constraint functions increase rapidly outside the feasible set (i.e. outside the snakes). Together, all these things make the problem rather difficult to solve if the following feasible, but far from optimal, starting point $\mathbf{x}^{(0)}$ is chosen:

$$x_i^{(0)} = \cos(\alpha_i + \frac{\pi}{12}), \quad x_{\ell+i}^{(0)} = \sin(\alpha_i + \frac{\pi}{12}), \quad x_{2\ell+i}^{(0)} = \sin(2\alpha_i + \frac{\pi}{6}), \quad \text{for } i = 1, \ldots, \ell.$$

With $\delta = 0.1$ and $\ell = 10$, the problem has 30 variables $x_j$ and 41 nonlinear inequality constraints. The optimal objective value turns out to be $f_0(\hat{\mathbf{x}}) = -10.02298$, while the objective value for the starting point is $f_0(\mathbf{x}^{(0)}) = +9.55926$. The problem is considered to be solved when a point $\mathbf{x}^{(k)}$ which satisfies the following requirements has been found: $-2 \le x_j^{(k)} \le 2$ for all the 30 variables, $f_i(\mathbf{x}^{(k)}) \le 10^{-5}$ for all the 41 nonlinear constraints (written as $f_i(\mathbf{x}) \le 0$), and $f_0(\mathbf{x}^{(k)}) \le f_0(\hat{\mathbf{x}}) + 10^{-5} = -10.02297$. In the optimal solution, all the 30 variables are strictly between their upper and lower bounds, and 19 of the 41 nonlinear constraints are satisfied with equality.

For GCMMA, with the default setting of the parameters described above, it took 39 outer iterations to solve the problem.

The ordinary MMA, with the default setting of the parameters described above, in some sense failed. Some iteration points became very much infeasible, and then it took a *very* long time to solve the corresponding MMA subproblems. In spite of this, the complete problem was in fact solved after 48 iteration (but where some iterations took a *very* long time). Later, after playing with the parameters, it turned out that if the parameter 0.5 in (3.6) was changed to 0.1, the parameter 1.2 in (3.8) was changed to 1.0, and the parameter 0.7 in (3.8) was changed to 0.95, then MMA solved the problem in 101 iterations *without* the numerical difficulties mentioned above. But such a problem-dependent trial and error is of course not satisfactory! Therefore, this example illustrates why GCMMA should be used instead of MMA.

# 6 References


**[1]** K. Svanberg, The method of moving asymptotes – a new method for structural optimization, *International Journal for Numerical Methods in Engineering*, 1987, 24, 359-373.

**[2]** K. Svanberg, A class of globally convergent optimization methods based on conservative convex separable approximations, *SIAM Journal of Optimization*, 2002, 12, 555-573.