CONSTANT GRADIENT LOCUS INTEGRATION IN PENALTY METHOD OPTIMIZATION

Matti Hyötyniemi, M.Sc. (eng) 2004-05-02
Lujax tmi
Graanintie 5
FIN 50190 Mikkeli

Abstract

Penalty function method is widely known way of transforming a constrained optimization problem into a sequence of unconstrained optimization problems by adding some penalty function to the unconstrained original function (see for example [[i]]. The modified unconstrained problem has one or more (local) minima that can be found with any standard minimization methods for unconstrained problems. The minima coordinates in the design parameter space are dependent of the scalar penalty parameter value *r*, and in the penalty method the value *r* is gradually changed until the penalty function value approaches (in theory) infinity when the constraints are violated and zero otherwise. The final local minima are then the minima for the original constrained problem.

In this report, an alternative method for solving the minimization problem is proposed. General differential form for the constant gradient locus in penalty method is introduced that can be used to integrate the path to the constrained problem minimum analytically or with any of the general differential solution methods, such as Euler or Runge-Kutta methods[ii]. Calculation savings could then be possible since repeated unconstrained minimisation cycles are not required. In principle, only one such calculation is needed in order to find the start point for integration with zero gradient.

Solution convergence is studied to some length. Some special cases, such as application to the linear problems are studied and some examples shown.

Nomenclature

$x_i,\ i = 1,\ldots n_x$	i:th design parameter
r	Penalty coefficient
$\vec{X} = \{x_1 \quad x_2 \quad \cdots \quad x_{n_x}\}^T$	Vector of design parameters
$\vec{X}_{\min,r}$	Design parameter vector at the (local) minimum for a given r
$f(\vec{X})$	Unconstrained target function
$g_j(\vec{X}),\ j = 1,\ldots n_g$	j:th inequality constraint function
$G_j(g_j(\vec{X}), r)$	j:th inequality penalty function
$h_k(\vec{X}),\ k = 1,\ldots n_e$	k:th equality constraint function
$H_k(h_k(\vec{X}), r)$	k:th equality penalty function
$F(\vec{X}, r)$	Transformed target function
$[H] = \begin{bmatrix} \frac{\partial^2}{\partial x_1^2} & \frac{\partial^2}{\partial x_1 \partial x_2} & \cdots & \frac{\partial^2}{\partial x_1 \partial x_{n_x}} \\ \vdots & \vdots & \ddots & \vdots \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2}{\partial x_{n_x} \partial x_1} & \frac{\partial^2}{\partial x_{n_x} \partial x_2} & \cdots & \frac{\partial^2}{\partial x_{n_x}^2} \end{bmatrix}$	Hessian matrix operator

Differential form of penalty method

General minimization problem with inequality and equality constraints can be formulated as

minimize $f(\vec{X})$

such that $g_j(\vec{X}) \geq 0$, $j = 1,...n_g$

and $h_k(\vec{X}) = 0$, $k = 1,...n_e$

In penalty method, this constrained minimization problem is replaced with a transformed, unconstrained minimization problem, where constraints are combined with the original target function via penalty functions. In a general case, the transformed target function has the form

$$F(\vec{X},r) = f(\vec{X}) + \sum_j G_j(g_j(\vec{X}),r) + \sum_k H_k(h_k(\vec{X}),r) \quad (1)$$

All penalty functions are assumed to be using penalty coefficient *r* in a similar manner (penalty either rises or decreases continuously in all penalty functions as the *r* changes). Penalty function *G* and *H* values are a function of constraint values with increasing magnitude as the constraint violation grows larger (external penalty functions, for example $H(h,r) = rh^2$) or as the constraint is being approached from the feasible region (internal penalty functions such as $G(g,r) = \frac{1}{rg}$) or some combination of these two ($G(g,r) = \exp(-rg)$ could be used)[1]. Inequality constraint g_j is violated if its value is less than zero and equality constraint h_k when it is non-zero. It is assumed that the transformed target function is continuous and twice differentiable in the calculation subdomain. Penalty functions such as $G(g,r) = \begin{cases} rg^2, g < 0 \\ 0, g \geq 0 \end{cases}$ whose second derivative is discontinuous, could yield problems.

---

[1] All these functions assume *r* to be positive and have increasing penalty as *r* grows. Further, $\frac{\partial^2 G}{\partial g^2} \geq 0$ and $\frac{\partial^2 H}{\partial h^2} \geq 0$ inside the calculation subdomain.

In the penalty method procedure initial r and $\vec{X}$ values are first chosen. Initial r is chosen so that it yields relatively mild penalties in order to make (numerical) unconstrained minimization procedure more stable. In theory usage of large penalties from the beginning would be efficient, but solution problems would result. With these settings the local minimum $\vec{X}_{\min,r}$ for the given r is searched with any standard unconstrained minimization technique, such as Powell's conjugate directions, steepest descent, or 2nd degree conjugate gradient method, possibly with DFP, BFGS or similar method updates for (inverse) Hessian matrix of F.

Using the resulting $\vec{X}_{\min,r}$ as the start point for the next iteration, penalties are increased by changing the value of r and new iteration round is proceeded until satisfying final $\vec{X}_{\min,r}$ is achieved.

Let us now consider how the gradient of F relative to design parameters changes as the $\vec{X}$ and r are changed infinitesimally:

$$\nabla F(\vec{X}+d\vec{X},r+dr)=\nabla F(\vec{X},r)+[H]F(\vec{X},r)d\vec{X}+\frac{\partial}{\partial r}\nabla F(\vec{X},r)dr \qquad (2)$$

If gradient is kept constant after the r has been changed, the $\vec{X}$ must change as follows:

$$\frac{d\vec{X}}{dr}=-\left([H]F(\vec{X},r)\right)^{-1}\frac{\partial}{\partial r}\nabla F(\vec{X},r) \qquad (3)$$

The Hessian of F $\left([H]F(\vec{X},r)\right)$ at the calculation point must be non-singular, otherwise solution is not available.

As the special case, in the local extremum for a given *r*, $\nabla F(\vec{X}, r) = \vec{0}$, where the gradient is taken over the design parameters. Provided, that the Hessian of *F* is also positive definite, the extremum is a local minimum.

In the penalty method, when the *r* is changed, the gradient at the new minimum remains $\nabla F(\vec{X}_{\mathrm{min},r}, r) = \vec{0}$. Therefore, equation (3) defines the locus for a local extremum as the *r* changes provided, that the initial point of integration is on such a locus. Otherwise equation (3) defines a constant gradient locus.

Diff. form in terms of target and constraint function properties

Typically, Hessian and gradient of F itself at a given r are seldom readily available or their calculation may be expensive. Therefore, equation (3) is often beneficial to express in terms of Hessians and gradients of f, g and h.

First we may note that second term of (3), $\frac{\partial}{\partial r}\nabla F(\vec{X},r)$, does not depend of f. We get, by substitution and chain derivation

$$\frac{\partial}{\partial r}\nabla F(\vec{X},r)=\sum_j \frac{\partial}{\partial r}\left(\frac{\partial G_j}{\partial g_j}\nabla g_j\right)+\sum_k \frac{\partial}{\partial r}\left(\frac{\partial H_k}{\partial h_k}\nabla h_k\right)=\sum_j \frac{\partial^2 G_j}{\partial r\partial g_j}\nabla g_j+\sum_k \frac{\partial^2 H_k}{\partial r\partial h_k}\nabla h_k \tag{4a}$$

Similarly, the Hessian of F can be converted into the following form:

$$[H]F(\vec{X},r)=[H]f+\sum_j\left(\frac{\partial G_j}{\partial g_j}[H]g_j+\frac{\partial^2 G_j}{\partial g_j^{\,2}}[D]g_j\right)+\sum_k\left(\frac{\partial H_k}{\partial h_k}[H]h_k+\frac{\partial^2 H_k}{\partial h_k^{\,2}}[D]h_k\right) \tag{4b}$$

where the operator matrix $[D]$ is defined as

$$[D]=\begin{bmatrix} \left(\frac{\partial}{\partial x_1}\right)^2 & \frac{\partial}{\partial x_1}\frac{\partial}{\partial x_2} & \cdots & \frac{\partial}{\partial x_1}\frac{\partial}{\partial x_{n_x}} \\ \frac{\partial}{\partial x_2}\frac{\partial}{\partial x_1} & \left(\frac{\partial}{\partial x_2}\right)^2 & \cdots & \vdots \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial}{\partial x_{n_x}}\frac{\partial}{\partial x_1} & \cdots & \cdots & \left(\frac{\partial}{\partial x_{n_x}}\right)^2 \end{bmatrix}$$

If target- and constraint function values, gradients and Hessians are available, the results (4a) and (4b) can be input directly into equation (3) without the need to calculate F –function properties explicitly.

It should be noted that if the target function is linear the Hessian of the target function f is zero everywhere. Since f is independent of *r, f doesn't affect the equation (3) at all* and constraints denote the solution alone after the initial point of integration has been defined.

Further, if constraints are linear, their Hessians are zero and only the rightmost parts of equation (4b) constraint terms contribute to the Hessian of *F*.

Convergence

When using numerical methods for the equation (3) solution, it is preferable and sometimes essential that the underlying differential form of the problem is convergent and small errors won't grow in magnitude during the integration. There may be errors in the estimate for the initial coordinates of the minimum locus. Numerical estimate of gradients and Hessian matrices may also be in error. Consequently, the divergence of the vector field obtained from (3) should preferably be less than, or equal to zero everywhere at the calculation locus neighborhood:

$$\nabla \cdot \left(\frac{d\vec{X}}{dr} dr \right) \le 0 \leftrightarrow \nabla \cdot \left(\left([H] F(\vec{X}, r) \right)^{-1} \frac{\partial}{\partial r} \nabla F(\vec{X}, r) dr \right) \ge 0 \qquad (5)$$

Unfortunately, this is very cumbersome criterion in practice if it is estimated numerically. For theoretical studies it is more useful.

Proposed solution procedure (numerical)

1. Choose an initial r and penalty functions G and H. By using some standard minimization technique for unconstrained functions, find the corresponding $\vec{X}_{\min,r}$ for F –function. The initial point calculation accuracy required may be estimated according to the convergence criterion in equation (5). If the problem shows good convergence properties, the accuracy can be relaxed and vice versa. However, pointwise convergence does not inevitably give true picture of the situation (see Discussion). Since Hessian matrices are needed in the integration step in any case, second order schemes seem a rational choice for minimization. Note that it would be wise to calculate the Hessians for f and g:s instead of for F directly and compose the Hessian of F according to the equation (4b) to be used in the minimization. In that way the (approximate) Hessians required in step 2 are readily available.
2. By using equation (3), integrate the result to the final r value required with any of the standard differential solvers. If analytic Hessians for target and constraint functions are not available they may be updated by, for example, with complementary DFP (see reference [i]) formula from the calculated gradients. It is also possible to make integration up to some intermediate value and do a unconstrained minimization update loop in order to remove any cumulated errors before continuing the path integration from the corrected path point.

Special case examples:

A few examples are shown in order to make things clearer. Two linear problems are first shown and one quadratic problem. All the problems are relatively simple and analytic equations are used for gradient and Hessian matrix calculations.

Example 1, Linear 1d –case

Let us consider a linear problem:

$$\begin{aligned} &f(x) = 1 + x \\ &g(x) = x - 2 > 0 \end{aligned}$$

Clearly, the optimum is *f*(2)=3.

We use interior penalty function $G(g,r) = \frac{1}{rg}$. *r* must be positive and *x* may not violate the constraint (since interior penalty function is not defined exactly on the constraint, we use inequality > instead of ≥). Since only 1 design parameter is used, we replace gradients with derivatives and Hessians with second derivatives.

$$\begin{aligned} &F(x,r) = 1 + x + \frac{1}{r(x-2)} \\ &\frac{\partial}{\partial r}\frac{\partial F}{\partial x} = \frac{1}{r^2(x-2)^2} \\ &\frac{\partial^2}{\partial x^2}F = \frac{2}{r(x-2)^3} \end{aligned}$$

By substituting these values to the equation (3) and after some reduction we obtain

$$\frac{dx}{dr} = \frac{2-x}{2r} \Leftrightarrow \frac{dx}{2-x} = \frac{dr}{2r}$$

In this case $\frac{\partial}{\partial x}\left(\frac{2-x}{2r}dr\right) = \frac{-dr}{2r} < 0$ always, (*dr* and *r* are positive) so solution is convergent everywhere in the calculation region. Further, $\frac{\partial^2}{\partial x^2}F = \frac{2}{r(x-2)^3} > 0$ always (when r is positive and x does not violate the constraint) so the only extremum value is the global minimum. By integrating this differential equation we obtain

$$\begin{aligned}
&\Big|_{x_0}^{x_{end}} -\ln(2-x) = \Big|_{r_0}^{r_{end}} \frac{\ln(r)}{2} \\
&\Leftrightarrow \ln\left(\frac{2-x_0}{2-x_{end}}\right) = \frac{1}{2}\ln\left(\frac{r_{end}}{r_0}\right) \\
&\Leftrightarrow \frac{2-x_0}{2-x_{end}} = \sqrt{\frac{r_{end}}{r_0}} \\
&\Rightarrow x_{end} = \frac{x_0-2}{\sqrt{\frac{r_{end}}{r_0}}} + 2
\end{aligned}$$

If we let $r_{end} \to \infty$*,* the corresponding x_{end}=2 *with any initial x_0 –r_0 –pair* provided that x_0 and r_0 are in the feasible region.

This phenomenon can be interpreted intuitively by considering that derivatives of a linear target function *f* and linear constraint function(s) *g* is constant in respect to *x*.

Let's assume that applied penalty functions have the property $\frac{\partial^2 G}{\partial g^2} > 0$ for all *g* within the calculation region. With linear *f* and *g*:s in respect to *x*, the resulting $\frac{\partial^2 F}{\partial x^2} = \frac{\partial}{\partial x}\left(\frac{\partial F}{\partial x}\right) > 0$. Therefore the derivative of *F* grows continuously with increasing *x* and consequently the derivative is single-valued.

Let us further assume that internal inequality penalty functions also have the property $\frac{\partial^2 G}{\partial g^2}(r \to \infty) \to \begin{cases} \infty, g \to 0+ \\ 0, g > 0 \end{cases}$. Consequently, as the penalty value *r* is increased, the *F* function becomes "sharper" near the constraint and different gradients converge (see figure 1). Finally, as the $r \to \infty$, the *F* function becomes singular at the constraint and loci for different gradients converge to the constraint.

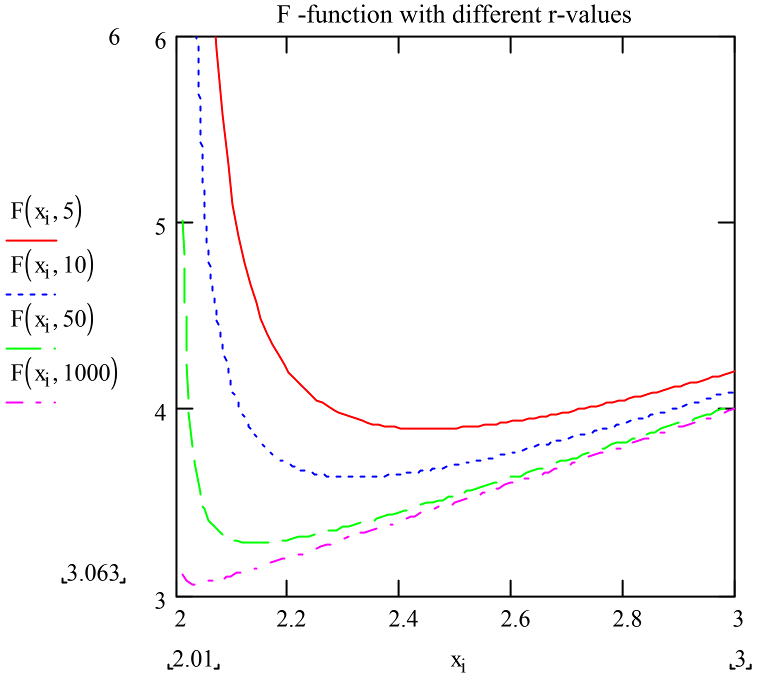


Figure 1 ***F*-function sharpening as the penalty term *r* increases**

The discussion above could propably be extended to consider other types of penalty functions and multidimensional linear problems also.

Another interesting thing in the current case is that if the initial r_0=0, the convergence happens to same constraint $x=2$ with any valid r_{end}>0. This result is rather theoretical, since G and F are singular with $r \to 0+$.

There may be several convergence points even in the linear cases as shown in the next example.

Example 2, Linear 2d-case

Let us consider a linear problem:

$$
\begin{aligned}
f(\vec{X}) &= 1 + x_1 + 2x_2 \\
g_1(\vec{X}) &= x_1 - 1 > 0 \\
g_2(\vec{X}) &= x_2 - 1 > 0 \\
g_3(\vec{X}) &= x_2 - x_1 + 1 > 0
\end{aligned}
$$

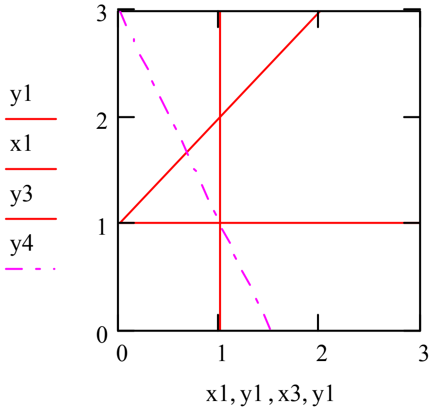


Figure 2 Constraints (solid red) and one constant target function value curve (dashdot). NOTE! x1 on vertical axis and x2 on the horizontal axis.

It can be seen that minimum value is obtained at $x_1 = x_2 = 1$ with $f = 4.0$.

We use same interior penalty functions $G(g,r) = \dfrac{1}{rg}$ as in the previous example for all constraints. Modified target function will be

$$F(\vec{X},r) = 1 + x_1 + 2x_2 + \frac{1}{r(x_1 - 1)} + \frac{1}{r(x_2 - 1)} + \frac{1}{r(x_2 - x_1 + 1)}$$

From (4a) and (4b)

$$\frac{\partial}{\partial r}\nabla F=\frac{1}{r^2}\begin{Bmatrix}\dfrac{1}{(x_1-1)^2}-\dfrac{1}{(x_2-x_1+1)^2}\\ \dfrac{1}{(x_2-1)^2}+\dfrac{1}{(x_2-x_1+1)^2}\end{Bmatrix}$$

$$[H]F=\frac{2}{r}\begin{bmatrix}\dfrac{1}{(x_1-1)^3}+\dfrac{1}{(x_2-x_1+1)^3} & \dfrac{-1}{(x_2-x_1+1)^3}\\ \dfrac{-1}{(x_2-x_1+1)^3} & \dfrac{1}{(x_2-1)^3}+\dfrac{1}{(x_2-x_1+1)^3}\end{bmatrix}$$

We substitute these values to (3) and make matrix division:

$$\frac{d\vec{X}}{dr}=\frac{-1}{2r\left(2x_2{}^3+3x_2x_1{}^2-3x_1x_2{}^2-6x_2x_1+6x_2-1\right)}*$$

$$\begin{Bmatrix}(x_1-1)\left(2x_2{}^3+3x_2x_1{}^2-3x_1x_2{}^2-x_1{}^2-6x_2x_1+6x_2+2x_1-2\right)\\ x_2(x_2-1)\left(3x_1{}^2+2x_2{}^2-3x_2x_1+x_2-6x_1+4\right)\end{Bmatrix}$$

For convergence criterion in eq. (5) we get

$$\frac{dr}{2r\left(3\cdot x_2\cdot x_1{}^2-6\cdot x_2\cdot x_1-3\cdot x_2{}^2\cdot x_1-1+6\cdot x_2+2\cdot x_2{}^3\right)^2}\cdot\begin{pmatrix}-3\cdot x_1{}^4\cdot x_2+18\cdot x_1{}^4\cdot x_2{}^2-66\cdot x_2{}^2\cdot x_1{}^3-36\cdot x_2{}^3\cdot x_1{}^3+12\cdot x_2\cdot x_1{}^3\ \ldots\\ +42\cdot x_2{}^4\cdot x_1{}^2+126\cdot x_2{}^2\cdot x_1{}^2+72\cdot x_2{}^3\cdot x_1{}^2-39\cdot x_2\cdot x_1{}^2\ \ldots\\ +6\cdot x_1{}^2-105\cdot x_2{}^2\cdot x_1-24\cdot x_2{}^5\cdot x_1\ \ldots\\ +48\cdot x_2\cdot x_1-12\cdot x_1-51\cdot x_2{}^4\cdot x_1-72\cdot x_2{}^3\cdot x_1+60\cdot x_2{}^2-14\cdot x_2{}^3\ \ldots\\ +8\cdot x_2{}^6+54\cdot x_2{}^4-30\cdot x_2+8\end{pmatrix}\ge 0$$

The term *dr/r* is always positive in this problem. The convergence isocontours relative to (*dr/r*) are shown in figures 2 and 3.

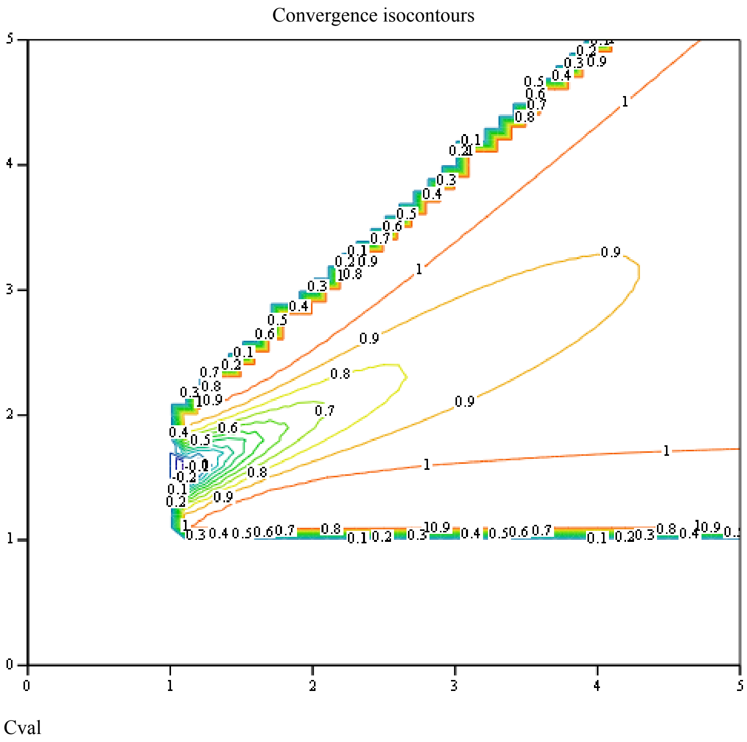


Figure 2 Convergence isocontours divided by (dr/r) (Note: contour values outside the constraint limits are forced to zero in the figures for clarity)

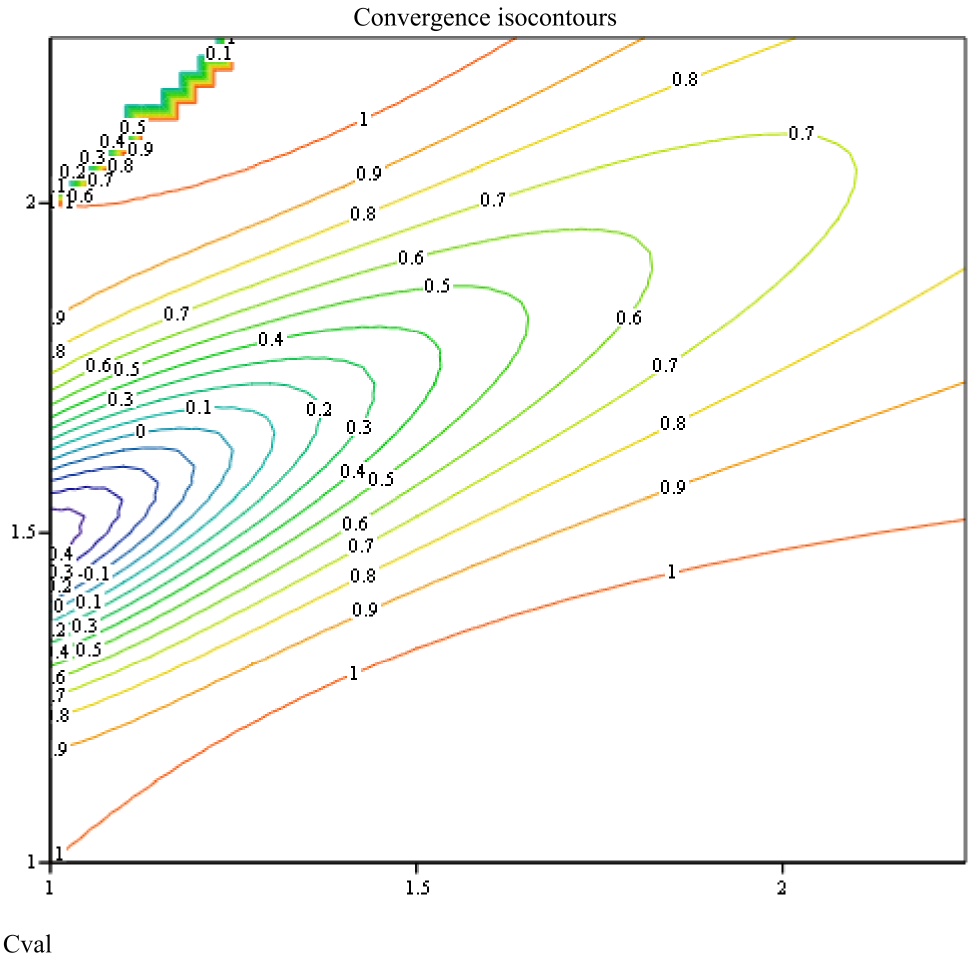


Figure 3 Convergence isocontours, detail

It can be seen that now the solution is convergent (value is >0) everywhere except near the constraint x_2-1>0 midspan.

Since obtained differential equation for the gradient loci is too difficult to be solved analytically, we integrate it with Mathcad®[iii]'s adaptive Runge-Kutta scheme. We start with r=1 and stop with r=10^6.

We start by finding the zero gradient with r=1. Gradient for the current function is

$$\nabla F\left(\vec{X},r\right)=\begin{Bmatrix} 2-\dfrac{1}{r\left(x_1-1\right)^2}+\dfrac{1}{r\left(x_2-x_1+1\right)^2} \\ 1-\dfrac{1}{r\left(x_1-1\right)^2}-\dfrac{1}{r\left(x_2-x_1+1\right)^2} \end{Bmatrix}$$

When solving for $\nabla F\left(\vec{X},r=1\right)=\begin{Bmatrix}0\\0\end{Bmatrix}$ simultaneously for both design parameters we get numerically with 6 decimal places for $\vec{X}_{\min,r=1}=\begin{Bmatrix}1.648187\\2.270131\end{Bmatrix}$.

The algorithm intermediate r evaluation values are shown at the table 1 leftmost column (0) and corresponding x_1 and x_2 values at the corresponding columns 1 and 2. The result converge towards the correct x-values at the optimum $\{1,1\}^{\mathrm{T}}$.

$\mathrm{rkadapt}(x, 1, 1000000, 0.000001, D, 100, 1) =$

	0	1	2
0	1	1.648187	2.270131
1	4.801277	1.307579	1.48348
2	7.188493	1.253811	1.386106
3	9.507766	1.221884	1.331647
4	12.667325	1.193097	1.284494
5	16.946196	1.167557	1.244055
6	22.785566	1.14492	1.20919
7	30.808731	1.124915	1.179052
8	41.909748	1.107292	1.15296
9	57.37997	1.09182	1.130363
10	79.099814	1.078286	1.110799
11	109.830768	1.066489	1.093882
12	153.664347	1.056244	1.079278
13	216.719105	1.047381	1.066699
14	308.233913	1.039742	1.055892
15	442.301227	1.033184	1.046638
16	640.645912	1.027577	1.038739
17	937.13271	1.022804	1.032022
18	$1.385168\cdot10^{3}$	1.018758	1.026335
19	$2.070008\cdot10^{3}$	1.015346	1.02154
20	$3.129512\cdot10^{3}$	1.012481	1.017517
21	$4.789612\cdot10^{3}$	1.010089	1.014159
22	$7.425852\cdot10^{3}$	1.008103	1.011371
23	$1.167182\cdot10^{4}$	1.006463	1.009069
24	$1.861335\cdot10^{4}$	1.005118	1.007182
25	$3.014229\cdot10^{4}$	1.004022	1.005643
26	$4.961281\cdot10^{4}$	1.003135	1.004399
27	$8.308219\cdot10^{4}$	1.002422	1.003399
28	$1.417045\cdot10^{5}$	1.001855	1.002603
29	$2.464468\cdot10^{5}$	1.001406	1.001974
30	$4.375943\cdot10^{5}$	1.001055	1.001481
31	$7.943652\cdot10^{5}$	1.000783	1.001099
32	$1\cdot10^{6}$	1.000698	1.00098

Table 1 Numerical integration results

Let us consider next the situations where the integration start point is not located at the minimum point corresponding to the r -value. We use initial r=1 and following initial test $\vec{X}$ values: $\begin{Bmatrix}1.5\\4\end{Bmatrix}$ $\begin{Bmatrix}2.1\\2\end{Bmatrix}$ $\begin{Bmatrix}3\\3\end{Bmatrix}\begin{Bmatrix}2.5\\3.001\end{Bmatrix}$ $\begin{Bmatrix}1.9\\2\end{Bmatrix}$.

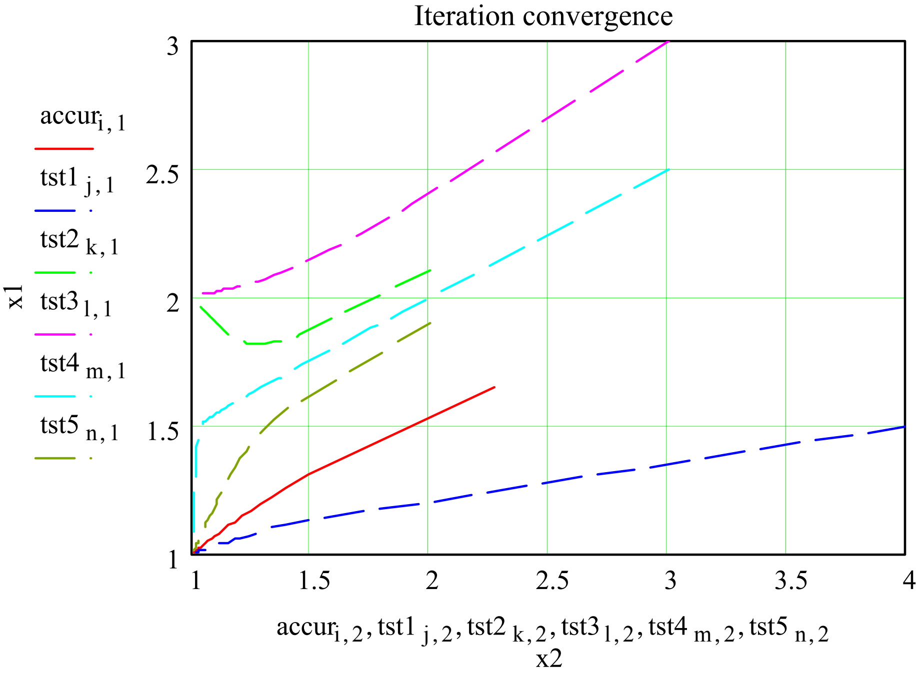


Figure 4 Loci from various start points. Zero gradient curve shown with solid line.

It seems that as far as the initial point is below the line $2x_1$-x_2=2, the result converges to $\{1,1\}^T$, but when start point is above that curve, the result converges to $\{2,1\}^T$. Test curve 4 shows a curve that is initially located near the line. At first, the curve closely follows the line. When the curve reaches the divergent domain (see figure 2 or 3), the curve abruptly bends towards one of the convergence points at $\{1,1\}^T$.

If the initial point is *exactly* on the curve $2x_1$-x_2=2, we obtain by substituting x_2 with $2x_1$-2

$$\frac{d\vec{X}}{dr}=\frac{(1-x_1)(10x_1^{\ 3}-43x_1^{\ 2}+62x_1-30)}{2r(10x_1^{\ 3}-42x_1^{\ 2}+60x_1-29)}\begin{Bmatrix}1\\2\end{Bmatrix}$$

so integration continues along the line $2x_1$-x_2=2 for all r. In this special case, integration converges to $\{1.5,1\}^T$ as r approaches infinity and x_1 is in the feasible domain. However, small disturbances may inhibit this to happen in numerical solution (differential equation becomes divergent as this point is approached) and convergence happens to $\{1,1\}^T$or $\{2,1\}^T$ instead.

Similarly as in the first example, it is intuitively obvious that the solutions converge to constraint intersection points in multidimensional linear problems also since F's curvature at these points approaches infinity while r grows in value. There may be several such points and where the solution converges depends of the start point. From a limited set of start points with non-zero gradient, the convergence may even happen to other points on the constraint surface as shown in the current example.

Example 3: Quadratic problem

We define a simple quadratic potential and make up a single constraint. In this case we use equality constraint

$$f(\vec{X}) = x_1^{\,2} + x_2^{\,2}$$

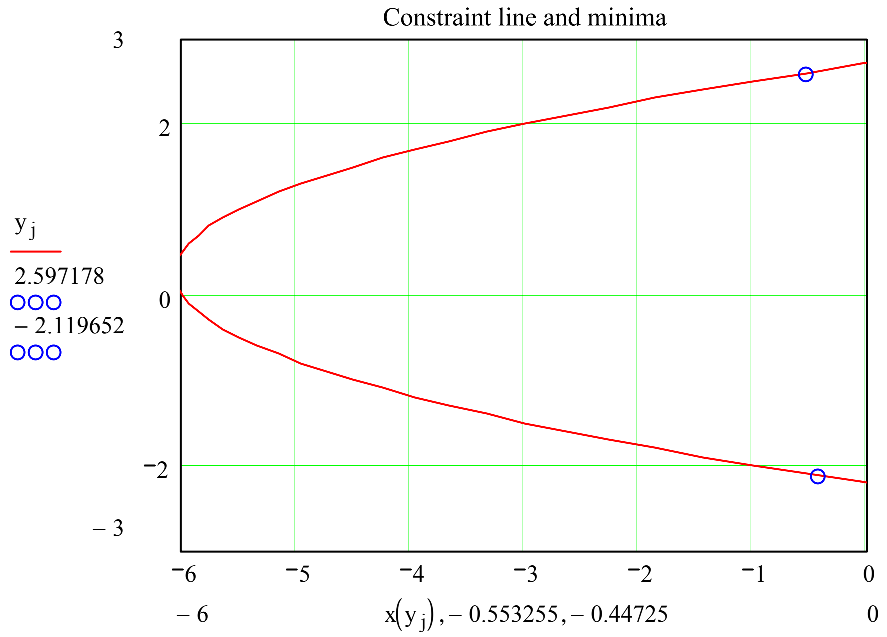
$$h(\vec{X}) = x_1^{\,2} - \frac{x_1}{2} - x_2 - 6 = 0$$

This problem has 2 minima; Numerically obtained minimas (Mathcad®) are, with 6 decimal places

$$\vec{X}_{\min A} = \begin{Bmatrix} 2.597178 \\ -0.553255 \end{Bmatrix} \qquad \vec{X}_{\min B} = \begin{Bmatrix} -2.119652 \\ -0.44725 \end{Bmatrix}$$

and target function values

$$f_{\min A} = 7.051425 \qquad f_{\min B} = 4.692957$$



In this case we use quadratic penalty function $H(r,h) = rh^2$. Modified unconstrained target function is now

$$F(\vec{X}, r) = x_1^{\,2} + x_2^{\,2} + r\left(x_1^{\,2} - \frac{x_1}{2} - x_2 - 6\right)^2$$

From (4a) and (4b)

$$\frac{\partial}{\partial r}\nabla F = 2\left\{\begin{matrix}\left(2x_1-\frac{1}{2}\right)\left(x_1{}^2-\frac{1}{2}x_1-x_2-6\right)\\ -x_1{}^2+\frac{1}{2}x_1+x_2+6\end{matrix}\right\}$$

$$[H]F = 2\begin{bmatrix}1+r\left(2x_1-\frac{1}{2}\right)^2+2r\left(x_1{}^2-\frac{1}{2}x_1-x_2-6\right) & -r\left(2x_1-\frac{1}{2}\right)\\ -r\left(2x_1-\frac{1}{2}\right) & 1+r\end{bmatrix}$$

Equation (3) becomes

$$\frac{d\vec{X}}{dr}=\frac{2x_1{}^2-x_1-2x_2-12}{4-43r+24rx_1{}^2+8r^2x_1{}^2-12rx_1-4r^2x_1-48r^2-8rx_2-8r^2x_2}\left\{\begin{matrix}1-4x_1\\ 2\left(2rx_1{}^2-rx_1-12r+1-2rx_2\right)\end{matrix}\right\}$$

For convergence criterion we get from eq. (5)

$$\frac{\left(\begin{array}{l}1801\cdot r-2896r^2\cdot x_1{}^2+4128r^2+1456r^2\cdot x_2+1496r^2\cdot x_1\,...\\ +296r\cdot x_1{}^2-124r\cdot x_1+680r\cdot x_2+96\cdot x_1{}^2+64\cdot r\cdot x_2{}^2\,...\\ +128r^2\cdot x_2{}^2-752\cdot r^3\cdot x_1{}^2+768r^3\cdot x_2-48\cdot x_1\,...\\ +64\cdot r^3\cdot x_2{}^2-32\cdot x_2-172-128r^3\cdot x_2\cdot x_1{}^2\,...\\ +64\cdot r^3\cdot x_2\cdot x_1+384r^3\cdot x_1+2304r^3-384r^2\cdot x_1{}^3\,...\\ +384r^2\cdot x_1{}^4-512\cdot r^2\cdot x_2\cdot x_1{}^2+256r^2\cdot x_2\cdot x_1\,...\\ +192\cdot x_1{}^4\cdot r-192\cdot r\cdot x_1{}^3+64\cdot r^3\cdot x_1{}^4-64\cdot r^3\cdot x_1{}^3\end{array}\right)}{\left(4-43\cdot r+24\cdot r\cdot x_1{}^2+8\cdot r^2\cdot x_1{}^2-12\cdot r\cdot x_1-4\cdot r^2\cdot x_1-48\cdot r^2-8\cdot r\cdot x_2-8\cdot r^2\cdot x_2\right)^2}\geq 0$$

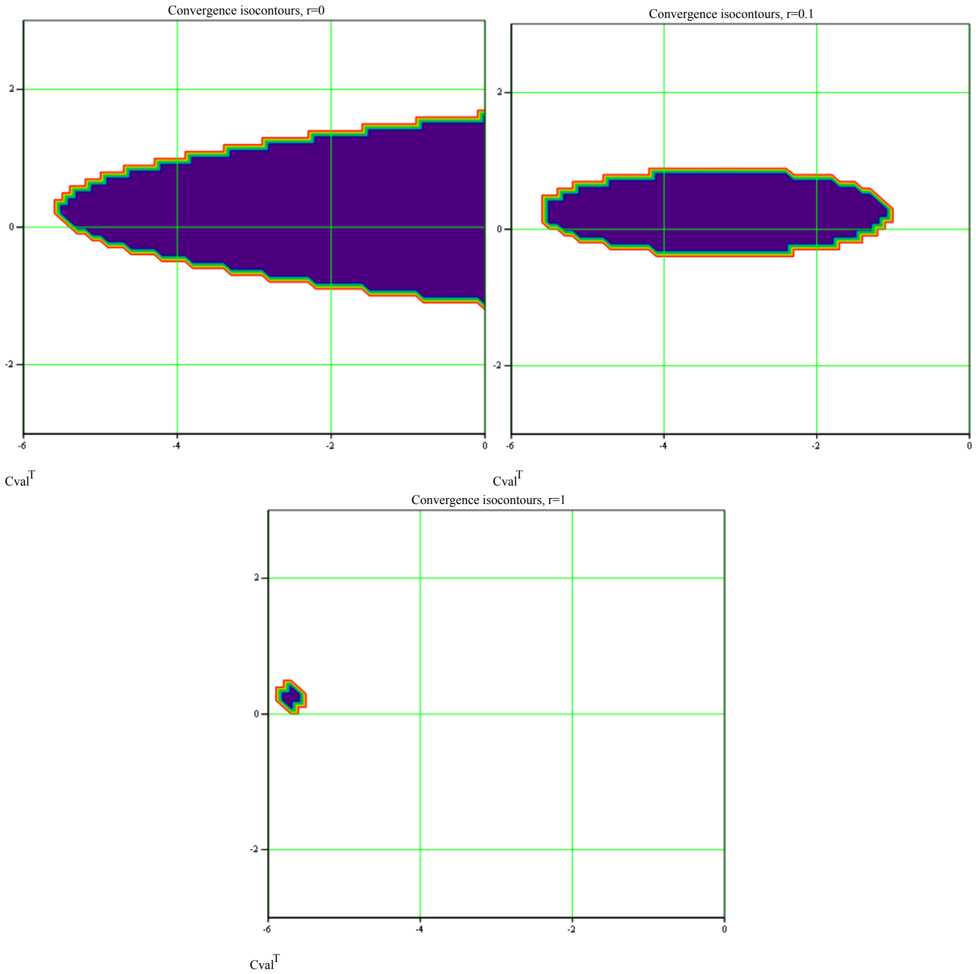


Figure 5 Convergence limits at different *r* values. Divergent domain (criterion value<0) shown with dark colour.

In this case the convergence criterion value depends heavily of *r* value. Divergent domain diminishes quickly as *r* grows.

We can immediately see, that if we start solution from r=0, whose $\vec{X}_{\min,r=0} = \begin{Bmatrix}0\\0\end{Bmatrix}$, we are initially in the divergent region. However, when using adaptive Runge-Kutta integration from the accurate start point up to *r* value of 10^6 we obtain following result:

$$\mathrm{rkadapt}\left[\begin{pmatrix}0\\0\end{pmatrix},0,10^{6},0.000001,D,100,1\right]=\begin{pmatrix}0 & 0 & 0\\ 1.207694 & -2.037687 & -0.445761\\ 2.743332 & -2.083887 & -0.446497\\ 5.072525 & -2.100385 & -0.446752\\ 7.06142 & -2.105837 & -0.446836\\ 10.003239 & -2.10992 & -0.446899\\ 14.443631 & -2.112928 & -0.446945\\ 21.29726 & -2.115108 & -0.446978\\ 32.140581 & -2.116656 & -0.447001\\ 49.77674 & -2.117733 & -0.447018\\ 79.370278 & -2.118464 & -0.447029\\ 130.838616 & -2.118948 & -0.447036\\ 224.206898 & -2.119259 & -0.447041\\ 402.652397 & -2.119452 & -0.447044\\ 769.16094 & -2.119568 & -0.447046\\ 1637.556233 & -2.119635 & -0.447047\\ 3500.055856 & -2.119667 & -0.447047\\ 8050.954751 & -2.119682 & -0.447047\\ 17683.978923 & -2.119689 & -0.447047\\ 46219.860829 & -2.119692 & -0.447047\\ 104554.751411 & -2.119694 & -0.447047\\ 323622.66869 & -2.119694 & -0.447047\\ 759627.516891 & -2.119694 & -0.447048\\ 1000000 & -2.119695 & -0.447048\end{pmatrix}$$

Table 2 Adaptive Runge-Kutta integration intermediate results between r=0 to 10^6.

As before, in the first column is shown the r- value and next columns correspond to the design parameters x_1 and x_2 respectively. The result converges closely towards the other (absolute) minimum of the constrained problem in this case. We now introduce small errors to the start point:

In the matrices tst2 and tst3 below the the first row shows the initial (disturbed) values for the integration.

$$\mathrm{tst2} = \begin{pmatrix} 0 & 0.01 & 0 \\ 1.076543 & -2.029518 & -0.448519 \\ 2.428067 & -2.07964 & -0.4496 \\ 4.463697 & -2.097645 & -0.449977 \\ 6.193527 & -2.10362 & -0.450101 \\ 8.742364 & -2.108107 & -0.450194 \\ 12.573421 & -2.111425 & -0.450262 \\ 18.45913 & -2.113837 & -0.450311 \\ 27.723081 & -2.115558 & -0.450347 \\ 42.703393 & -2.11676 & -0.450371 \\ 67.6755 & -2.117581 & -0.450388 \\ 110.776732 & -2.118127 & -0.450399 \\ 188.25754 & -2.118479 & -0.450406 \\ 334.632682 & -2.1187 & -0.450411 \\ 630.154841 & -2.118833 & -0.450414 \\ 1300.518193 & -2.118911 & -0.450415 \\ 3047.492289 & -2.118952 & -0.450416 \\ 5943.857428 & -2.118968 & -0.450416 \\ 16987.905171 & -2.118978 & -0.450416 \\ 31908.07799 & -2.118981 & -0.450416 \\ 105965.724827 & -2.118983 & -0.450416 \\ 207840.743991 & -2.118983 & -0.450416 \\ 717215.839809 & -2.118984 & -0.450416 \\ 1000000 & -2.118984 & -0.450416 \end{pmatrix} \quad \mathrm{tst3} = \begin{pmatrix} 0 & 0 & 0.01 \\ 1.199221 & -2.039164 & -0.435741 \\ 2.712334 & -2.08557 & -0.436467 \\ 4.99772 & -2.1022 & -0.43672 \\ 6.943527 & -2.107709 & -0.436803 \\ 9.814969 & -2.111842 & -0.436865 \\ 14.138147 & -2.114894 & -0.436911 \\ 20.792185 & -2.117109 & -0.436944 \\ 31.28699 & -2.118687 & -0.436967 \\ 48.296777 & -2.119788 & -0.436984 \\ 76.726235 & -2.120538 & -0.436995 \\ 125.943405 & -2.121035 & -0.437002 \\ 214.738555 & -2.121356 & -0.437007 \\ 383.258577 & -2.121556 & -0.43701 \\ 725.77981 & -2.121676 & -0.437012 \\ 1515.280077 & -2.121746 & -0.437013 \\ 3398.459397 & -2.121782 & -0.437013 \\ 7063.109651 & -2.121797 & -0.437014 \\ 19395.816101 & -2.121806 & -0.437014 \\ 37946.4656 & -2.121808 & -0.437014 \\ 130699.713095 & -2.12181 & -0.437014 \\ 251046.915049 & -2.12181 & -0.437014 \\ 852782.924817 & -2.121811 & -0.437014 \\ 1000000 & -2.121811 & -0.437014 \end{pmatrix}$$

With r=10^6, the numerically obtained $\vec{X}_{\min,r=10^6}$ coordinates with a normal unconstrained minimization method are the same as $\vec{X}_{\min,B}$ coordinates of constrained minimum within 6 decimal places. 2-norms of final error (distance to the correct minimum) in different test cases are tabulated below.

Test case	Initial error norm	Final error norm
Accurate initial point	0.0	0.000207
tst2	0.01	0.003236
tst3	0.01	0.010461

Initial errors result into errors in the integration end result, but small initial errors don’t destroy the end result despite of the initially divergent differential equation in the current case.

Discussion

Numerical sensitivity of the solution method was not investigated. These properties depend heavily of the chosen calculation algorithms, penalty functions etc.

In examples 2 and 3 integration goes *temporarily* via the divergent region without fatal consequences if the design parameters have been disturbed. Discarding the results due to the divergent differential equation somewhere along the integration path would therefore be too conservative. Perhaps the convergence criterion should somehow take into account the ”cumulative” divergence properties *along* the integration path, not only at separate points.

On the other hand, current convergence estimate is difficult and expensive to calculate. Calculationally more economical and still robust estimate for stability and convergence would be preferred.

The inverse Hessian of F must currently be calculated at every evaluation point. For larger problems, this may become expensive. Possibility for some kind of update formula for $\left([H]F(\vec{X},r)\right)^{-1}$, similar to DFP and BFGS algorithms for usual inverse Hessians, defined in terms of target and constraint function properties, could be worth studying.

Conclusions

Presented method seems to be an usable way for finding constrained target function minimum or as a theoretical tool for penalty method development research. Linear problems exhibit potentially very useful property of convergence into singular points.

References

---

[i] Haftka, R.T., Gürdal, Z., Kapat, M.P.: Elements of Structural Optimization. 2nd editon, Kluwer Academic Publishers

[ii] See, for example, Mäkelä, Nevanlinna, Virkkunen: Numeerinen matematiikka. 3rd edition, Gaudeamus 1986 (in finnish)

[iii] Mathcad® 2000 Professional –computer program. Mathsoft Co.