STRUCTURAL APPROXIMATIONS

Pierre DUYSINX
LTAS – Automotive Engineering
Academic year 2020-2021

SPECIFICATIONS OF STRUCTURAL APPROXIMATIONS

Characteristics of approximations

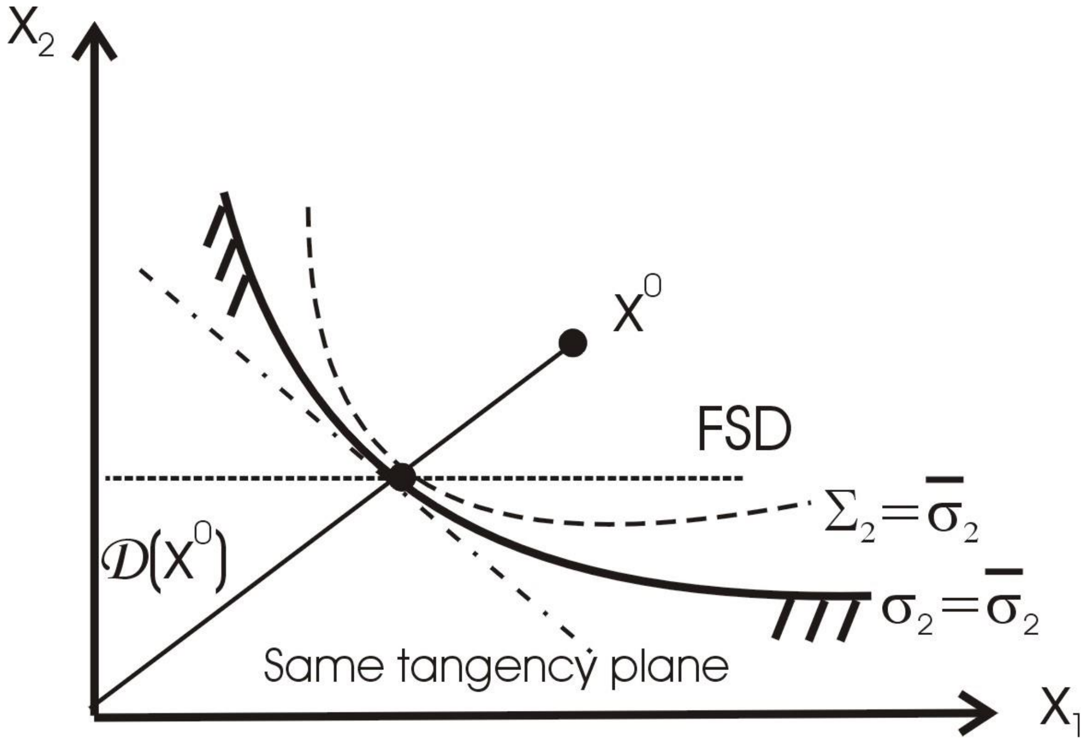
- When building approximations, one pursues several, and sometimes conflicting goals.

- In order to have an efficient solution procedure with dual methods, approximations have to be:

  – *Convex,* so that the sub-problem is convex. The solution is unique and dual solution is equivalent to primal solution.

  – *Separable*, in order to have a low-cost procedure when solving Lagrangian problem, which gives rise to the relation between primal and dual variables. This relation is often fully explicit and in closed form.

Characteristics of approximations



- Zero order and first order approximation of stresses
  - Zero order ➔ function value
  - First order: function value + all first derivatives

Characteristics of approximations

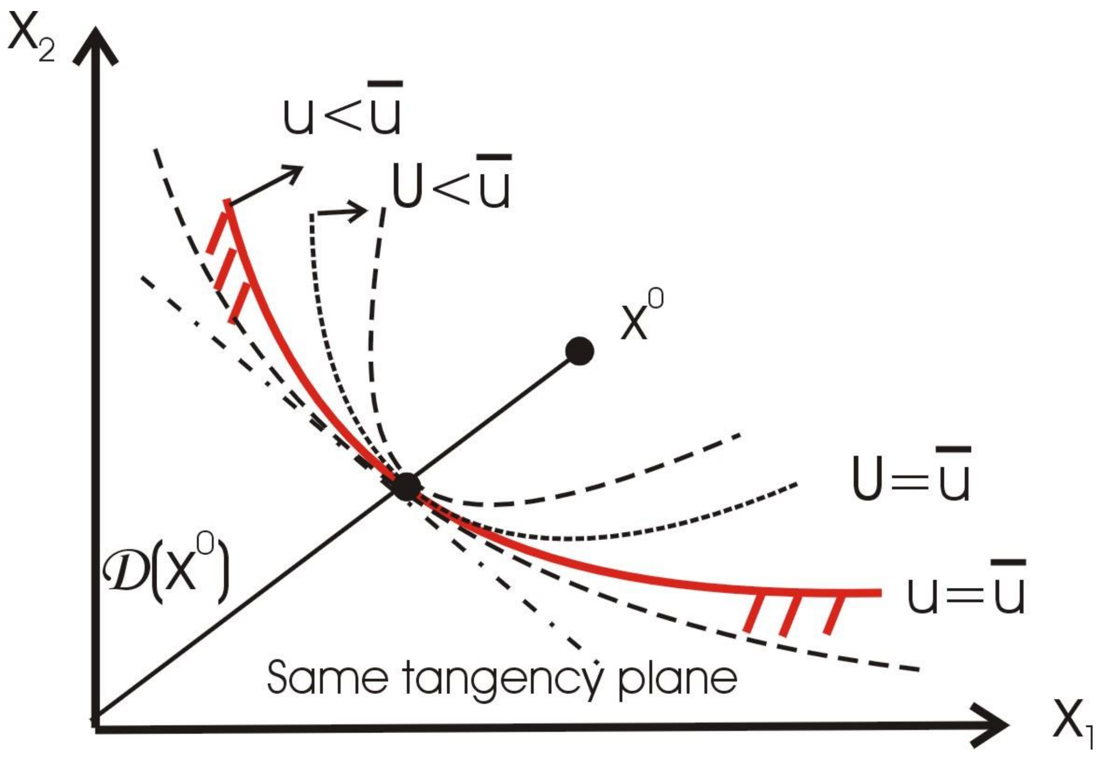
- In order to come to a stationary solution within a minimum number of stages and with a minimum of computational effort, one would like to have the following properties:

  – *Precision* to have the best fit to the exact response functions in the largest neighborhood (trust region).

  – *Conservative character* to generate a sequence of steadily improved feasible designs. This leads to increase the curvature (convexity terms) of the approximation and to reduce the size of the trust region of the approximation. However, this also slows down the convergence rate.

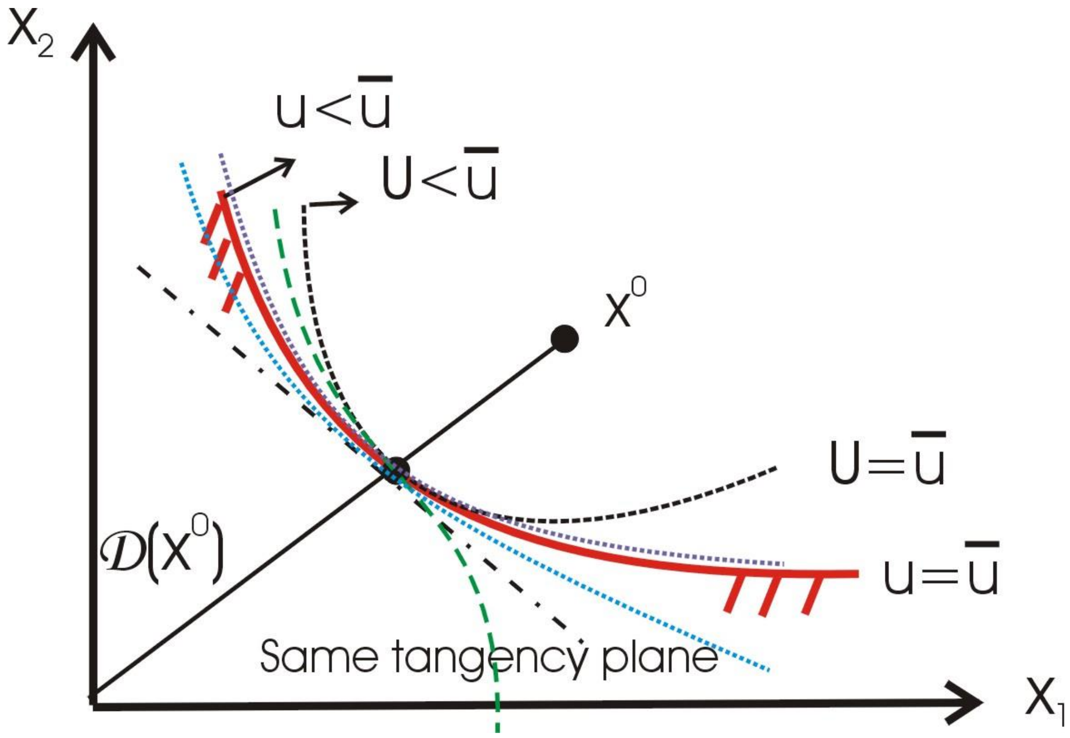
  – *Minimum computational effort* to generate the approximation and to calculate the necessary information (first and second order derivatives, etc.). With lesser information the scheme is also lesser precise.

Characteristics of approximations



- First order convex approximations of the displacement constraint
  - Increasing curvature ➔ more conservative (locally)

Characteristics of approximations



- First order approximations of the displacement constraint
  - Precise approximations → small error locally
  - Precision is different from conservative (green approximation) and convex (see blue approximation)

Characteristics of approximations

- In addition, one would like that the approximate sub-problem procedure should be:
  - *Robust* to be able handle various (all) kinds problem
  - *Flexible* to have a general procedure
  - *Globally convergent,* i.e. which means that the procedure can converge from any starting point

- Approximation schemes are
  - Local expansions around the current design point
  - Taylor expansions in terms of appropriate intermediate variables, like reciprocal variables $z=1/x$

LINEAR APPROXIMATIONS
AND
SEQUENTIAL LINEAR PROGRAMMING

Linear approximation and Sequential Linear Programming

- Linear approximation = first order Taylor expansion around x^0:

$$\tilde{g}_j(\mathbf{x}) \;=\; g_j(\mathbf{x^0}) + \sum_{i=1}^{n} \frac{\partial g(\mathbf{x^0})}{\partial x_i}\,(x_i - x_i^0)$$

- When linear approximation is applied to each function of the problem, one transforms the problem into a *sequence of linear programming problems (SLP)*:

$$\begin{aligned} \min_{x_i} \quad & g_0(\mathbf{x^0}) + \nabla g_0(\mathbf{x^0})^T(\mathbf{x} - \mathbf{x^0}) \\ \text{s.t.} \quad & g_j(\mathbf{x^0}) + \nabla g_j(\mathbf{x^0})^T(\mathbf{x} - \mathbf{x^0}) \;\leq\; 0 \\ & \underline{x}_i \;\leq\; x_i \;\leq\; \bar{x}_i \end{aligned}$$

SEQUENTIAL LINEAR PROGRAMMING METHOD

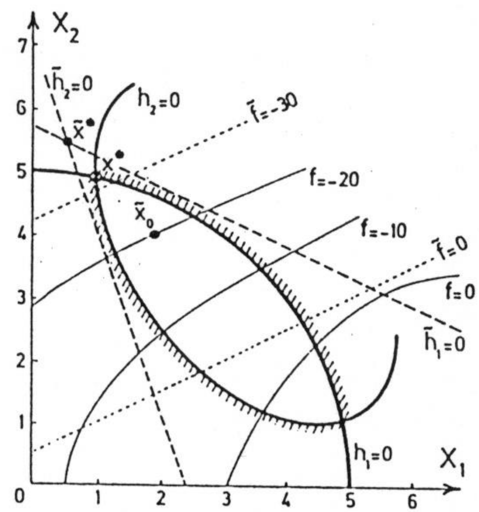
- The current design point is $x^{(k)}$. Using the first order Taylor expansion of f(x), $h_j(x)$, we can get a linear approximation of the NL problem in $x^{(k)}$:

$$\min_{\boldsymbol{x}} \quad f(\boldsymbol{x}^{(k)}) + (\boldsymbol{x} - \boldsymbol{x}^{(k)})^T \nabla f(\boldsymbol{x}^{(k)})$$

$$\text{s.t.} \quad h_j(\boldsymbol{x}^{(k)}) + (\boldsymbol{x} - \boldsymbol{x}^{(k)})^T \nabla h_j(\boldsymbol{x}^{(k)}) \leq 0 \quad j = 1 \ldots q$$

$$\underline{x_i} \leq x_i \leq \overline{x_i} \quad i = 1, \ldots, n$$

- Solving this LP problem, we get a new point in $x^{(k+1)}$ and start again.



PROBLEM FUNCTIONS
TRUE
$f(x) = 4x_1 - x_2^2 - 12$
$h_1(x) = 25 - x_1^2 - x_2^2$
$h_2(x) = 10x_1 - x_1^2 + 10x_2 - x_2^2 - 34$
LINEARIZED AT $\tilde{x}_0 = (2,4)$
$\bar{f}(x) = 4x_1 - 8x_2 + 4$
$\bar{h}_1(x) = 45 - 4x_1 - 8x_2$
$\bar{h}_2(x) = -14 + 6x_1 + 2x_2$

Linear approximation and Sequential Linear Programming

- Advantages of SLP
  - One can resort to efficient algorithms to solve the linear sub-problem like SIMPLEX or interior point methods.
  - The method is general and can treat a wide variety of problems

- Difficulties and drawbacks of SLP
  - Ignore the convexity of the non-linear problem so that
    - one can have quite systematic constraint violations (too little conservative approximation)
    - one can have some oscillating convergence processes
    - one can miss an optimum, which is not located at a vertex of constraints
  - Necessity of an appropriate move-limit strategy
    - this one can slow down convergence process
  - Require often an important number of iterations: 100 iterations are usually necessary to come to a stationary solution in topology

SEQUENTIAL LINEAR PROGRAMMING METHOD

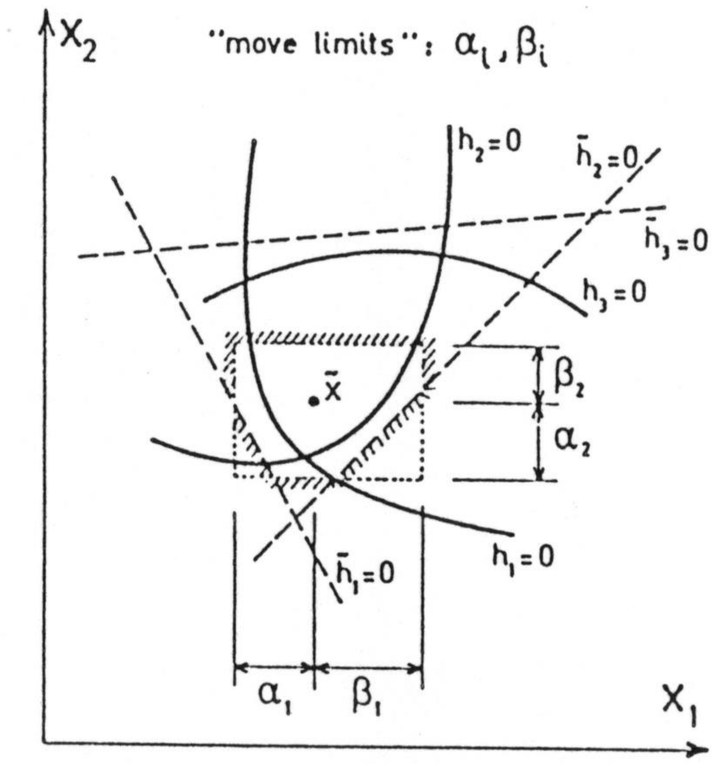
MOVE LIMIT STRATEGY

- Introduce a box constraint around the current design point to limit the variation domain of the design variables

$$\hat{x}_i - \alpha_i \leq x_i \leq \hat{x}_i + \beta_i$$

- Of course, take the most restrictive constraints with the side constraints

$$max(\underline{x}_i, x_i^0 - \alpha_i) \ \leq \ x_i \ \leq \ min(\bar{x}_i, x_i^0 + \alpha_i)$$



Linear approximation and Sequential Linear Programming

- Move-limits strategy to serve as a trust region of the approximation

$$max(\underline{x}_i, x_i^0 - \alpha_i) \;\leq\; x_i \;\leq\; min(\bar{x}_i, x_i^0 + \alpha_i)$$

- For topology, we suggest the following adaptive strategy:
  - Iterations k=1,2: initial 10% move-limits

$$\alpha_i \;=\; 0.1(\bar{x}_i - \underline{x_i})$$

  - Iterations k>2: move-limits update

$$\alpha_i^{(k+1)} = 0.7\alpha_i^{(k+1)} \quad \text{if} \quad \Delta x_i^{(k+1)} \Delta x_i^{(k)} < 0$$
$$\alpha_i^{(k+1)} = 1.2\alpha_i^{(k+1)} \quad \text{if} \quad \Delta x_i^{(k+1)} \Delta x_i^{(k)} \geq 0$$

FIRST ORDER STRUCTURAL APPROXIMATIONS

Reciprocal expansion scheme

- Key role of reciprocal variables, which have been known since 70ies to reduced the non-linear character of sizing problems in structural optimization (see Schmidt and his co-authors)

- Local approximation based on first order Taylor expansion in terms of intermediate variables $y_i = 1/x_i$:

$$\tilde{g}_j(\mathbf{x}) \; = \; g_j \; (\mathbf{x^0}) + \sum_{i=1}^{n} \; -(x_i^0)^2 \; \frac{\partial g(\mathbf{x^0})}{\partial x_i} \; (\frac{1}{x_i} - \frac{1}{x_i^0})$$

- In a lot of applications (including shape variables), this change of variable is favorable for convergence properties.

- Reciprocal scheme leads to positive second order derivatives (and so a convex scheme) if all first order derivatives are negative as in determinate or weakly under-determinate structures.

Reciprocal expansion scheme

- Reciprocal scheme leads to convex approximation (all positive second order derivatives if all first order derivatives are negative.

$$\tilde{g}(\mathbf{x}) = g(\mathbf{x}^0) + \sum_{i=1}^{n}(-1)(x_i^0)^2 \left.\frac{\partial g}{\partial x_i}\right|_0 \left(\frac{1}{x_i} - \frac{1}{x_i^0}\right)$$

$$\frac{\partial \tilde{g}}{\partial x_i} = (-1)(x_i^0)^2 \left.\frac{\partial g}{\partial x_i}\right|_0 \frac{-1}{x_i^2}$$

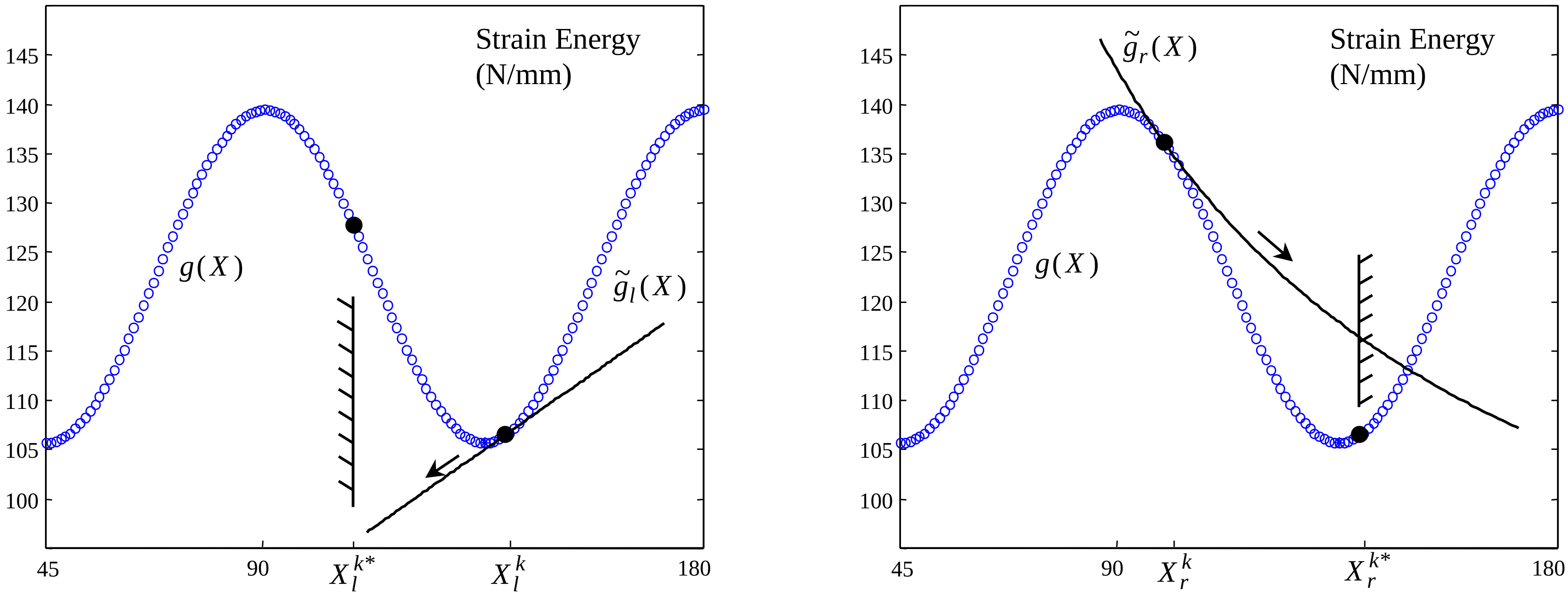
$$\left.\frac{\partial \tilde{g}}{\partial x_i}\right|_0 = (+1)(x_i^0)^2 \left.\frac{\partial g}{\partial x_i}\right|_0 \frac{-1}{(x_i^0)2} = \left.\frac{\partial g}{\partial x_i}\right|_0$$

$$\frac{\partial^2 \tilde{g}}{\partial x_i^2} = (x_i^0)^2 \left.\frac{\partial g}{\partial x_i}\right|_0 \frac{-2}{x_i^3}$$

$$\left.\frac{\partial^2 \tilde{g}}{\partial x_i^2}\right|_0 = (x_i^0)^2 \left.\frac{\partial g}{\partial x_i}\right|_0 \frac{-2}{(x_i^0)^3} = \left.\frac{\partial g}{\partial x_i}\right|_0 \frac{-2}{x_i^0}$$

$$\left.\frac{\partial^2 \tilde{g}}{\partial x_i^2}\right|_0 > 0 \iff \left.\frac{\partial g}{\partial x_i}\right|_0 < 0$$

Linear and reciprocal expansions



Approximation of the strain energy in a two plies symmetric laminate subject to shear load and torsion (Bruyneel and Fleury, 2000)

CONLIN approximation

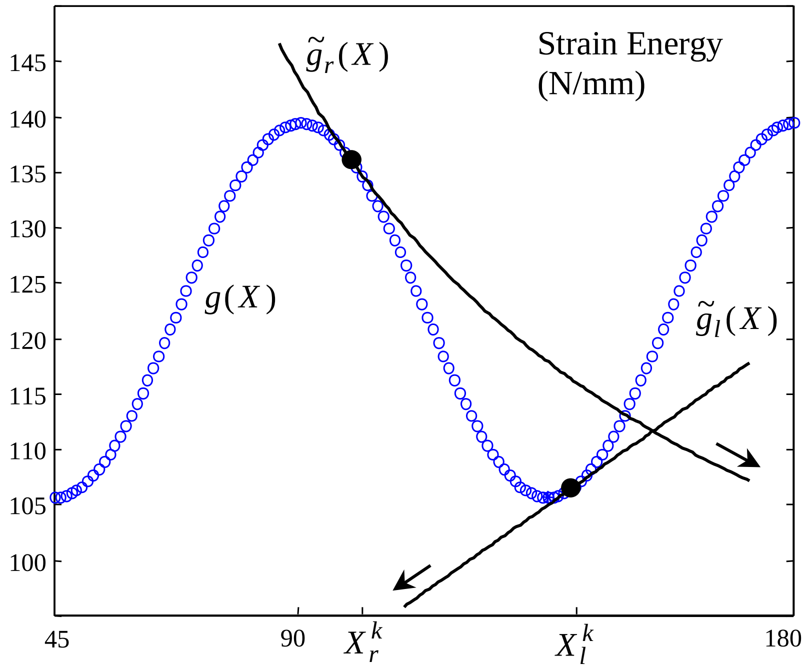
- Reciprocal expansion scheme leads to non conservative approximations when the sensitivities have mixed signs
- Idea: when sensitivities can be negative or positive, combine the reciprocal scheme with the linear approximation
- Automatic linearization procedure: choice of either direct or reciprocal variables based on the sign of the first derivative and is independent of the nature of the constraint

- CONLIN approximation (Fleury and Braibant, 1986)

$$\tilde{g}_j(\mathbf{x}) = g_j(\mathbf{x^0}) + \sum_{+} \frac{\partial g_j}{\partial x_i}(x_i - x_i^0) - \sum_{-} (x_i^0)^2 \frac{\partial g_j}{\partial x_i}(\frac{1}{x_i} - \frac{1}{x_i^0})$$

when the derivative is positive select the linear approximation while when the derivative is negative select the reciprocal expansion

CONLIN approximation



Approximation of the strain energy in a two plies symmetric laminate subject to shear load and torsion (Bruyneel and Fleury, 2000)

CONLIN approximation

PROPERTIES OF CONLIN

- Extension of the initial approximation concept based on reciprocal variables
- CONvex LINearization scheme, because the second derivatives are all positive or zero.
- CONLIN is the most conservative scheme that can be built among all combinations of mixed direct and reciprocal variables (Starnes and Haftka, 1979)
- Tendency to generate *steadily feasible designs*
- Optimum solution is generally reached within 10 to 20 FE analyses, nearly independently of the number of design variables
- *Rigorous mathematical convergence proofs* toward a local optimum under few hypotheses (Nguyen, Strodiot and Fleury, 1987)
- Successful extension to shape optimization (Braibant and Fleury, 1986)
- CONLIN problems can be efficiently solved with dual methods (Fleury, 1989)

CONLIN approximation

DRAWBACKS OF CONLIN

- CONLIN introduces fixed curvatures, which can be too small or too big (over-conservative). One can only change it through tricky change of variables. $z_i = x_i +/- a_i$

- Segregation of treatment:
  - if the derivative is negative, asymptote in x=0 and the approximation has a positive curvature,
  - if the derivative is positive, the linear approximation has a zero curvature.

- Convergence process can oscillate or be very slow ➔ move limits

- Despite numerous successful applications, one reports failure examples (e.g. Svanberg, 1987), especially when the sign of the sensitivity is changing around the accumulation point.

MMA approximation

- Ideas of the Method of Moving Asymptotes:
  - use intermediate linearization variables that allow modifying the degree of convexity and so the conservative character
  - keep and generalize the key role of reciprocal variables
- Intermediate variables:
  - if the derivative i is positive $$y_i = \frac{1}{U_i - x_i}$$
  - if the derivative i is negative $$z_i = \frac{1}{x_i - L_i}$$

$$\tilde{g}_j(\mathbf{x}) = g_j(\mathbf{x^0}) + \sum_{+} \frac{\partial g_j}{\partial y_i}(y_i - y_i^0) + \sum_{-} \frac{\partial g_j}{\partial z_i}(z_i - z_i^0)$$

$$= g_j(\mathbf{x^0}) + \sum_{+} \frac{\partial g_j}{\partial y_i}\left(\frac{1}{U_i - x_i} - \frac{1}{U_i - x_i^0}\right) + \sum_{-} \frac{\partial g_j}{\partial z_i}\left(\frac{1}{x_i - L_i} - \frac{1}{x_i^0 - L_i}\right)$$

MMA approximation

- Compute the change of variable

$$\frac{\partial g}{\partial y_i} = \frac{\partial g}{\partial x_i}\,\frac{\partial x_i}{\partial y_i}$$

$$y_i = \frac{1}{U_i - x_i} \qquad \Longleftrightarrow \qquad x_i = U_i - \frac{1}{z_i}$$

$$\frac{\partial x_i}{\partial y_i} = -\frac{-1}{y_i^2} = +(U_i - x_i)^2$$

- Finally, it comes $$\frac{\partial g}{\partial y_i} = \frac{\partial g}{\partial x_i}\,(U_i - x_i)^2$$

- Similarly $$\frac{\partial g}{\partial z_i} = \frac{\partial g}{\partial x_i}\frac{\partial x_i}{\partial x_i} = \frac{\partial g}{\partial x_i}\,(-1)\,(x_i - L_i)^2$$

MMA approximation

- Let's write the MMA approximation

$$\tilde{g}_j(\mathbf{x}) = g_j(\mathbf{x^0}) + \sum_{+}(U_i - x_i^0)^2 \left.\frac{\partial g_j}{\partial x_i}\right|_0 \left(\frac{1}{U_i - x_i} - \frac{1}{U_i - x_i^0}\right)$$

$$- \sum_{-}(x_i^0 - L_i)^2 \left.\frac{\partial g_j}{\partial x_i}\right|_0 \left(\frac{1}{x_i - L_i} - \frac{1}{x_i^0 - L_i}\right)$$

- Let's define

$$r_j^0 = g_j(\mathbf{x^0}) - (U_i - x_i^0)^2 \left.\frac{\partial g_j}{\partial x_i}\right|_0 \frac{1}{U_i - x_i^0} + (x_i^0 - L_i)^0 \left.\frac{\partial g_j}{\partial x_i}\right|_0 \frac{1}{x_i^0 - L_i}$$

$$= g_j(\mathbf{x^0}) - (U_i - x_i^0) \left.\frac{\partial g_j}{\partial x_i}\right|_0 + (x_i^0 - L_i) \left.\frac{\partial g_j}{\partial x_i}\right|_0$$

$$p_i = (U_i - x_i^0)^2 \left.\frac{\partial g_j}{\partial x_i}\right|_0 = \max\{0, (U_i - x_i^0)^2 \left.\frac{\partial g_j}{\partial x_i}\right|_0\}$$

$$q_i = -(x_i^0 - L_i)^2 \left.\frac{\partial g_j}{\partial x_i}\right|_0 = \max\{0, -(x_i^0 - L_i)^2 \left.\frac{\partial g_j}{\partial x_i}\right|_0\}$$

MMA approximation

- The MMA approximation proposed by Svanberg (1987) write

$$\tilde{g}_j(\mathbf{x}) = g_j(\mathbf{x^0}) + \sum_{+}(U_i - x_i^0)^2 \left.\frac{\partial g_j}{\partial x_i}\right|_0 \left(\frac{1}{U_i - x_i} - \frac{1}{U_i - x_i^0}\right)$$
$$- \sum_{-}(x_i^0 - L_i)^2 \left.\frac{\partial g_j}{\partial x_i}\right|_0 \left(\frac{1}{x_i - L_i} - \frac{1}{x_i^0 - L_i}\right)$$

- Or

$$\tilde{g}_j(\mathbf{x}) = r_j(\mathbf{x^0}) + \sum_{+} p_i \frac{1}{U_i - x_i} - \sum_{-} q_i \frac{1}{x_i - L_i}$$

$$p_i = \max\{0, (U_i - x_i^0)^2 \left.\frac{\partial g_j}{\partial x_i}\right|_0\}$$
$$q_i = \max\{0, -(x_i^0 - L_i)^2 \left.\frac{\partial g_j}{\partial x_i}\right|_0\}$$

MMA approximation

- Ideas of the Method of Moving Asymptotes:
  - use intermediate linearization variables that allow modifying the degree of convexity and so the conservative character
  - keep and generalize the key role of reciprocal variables
- Intermediate variables:
  - $1/(U_i-x_i)$ if the derivative i is positive
  - $1/(x_i-L_i)$ if the derivative i is negative
- MMA approximation (Svanberg, 1987)

$$\tilde{g}_j(\mathbf{x}) = r_j^0 + \sum_{i=1}^{n} \frac{p_{ij}}{U_i - x_i} + \sum_{i=1}^{n} \frac{q_{ij}}{x_i - L_i}$$

with

$$p_{ij} = \max\{0, (U_i - x_i^0)^2 \frac{\partial g_j}{\partial x_i}\}$$

$$q_{ij} = \max\{0, -(x_i^0 - L_i)^2 \frac{\partial g_j}{\partial x_i}\}$$

MMA approximation

- Of course, one has the condition:

$$L_i^{(k)} < x_i^{(k)} < U_i^{(k)}$$

- Generalization of CONLIN and linear schemes for particular values of U_i and L_i:
  - reciprocal expansion $L_i = 0$
  - linear approximation U_i, U_i = infinity

- The asymptotes play the role of move-limits

  By analogy with interior point methods in which side constraints can be taken into account through barrier functions

$$\ln(x_i - \underline{x}_i) \quad \text{or} \quad \frac{1}{x_i - \underline{x}_i}$$

- Parameters of the approximations: U_i and L_i

MMA approximation

- Curvature of MMA approximation can also be obtained

$$\tilde{g}_j(\mathbf{x}) = g_j(\mathbf{x^0}) + \sum_{+}(U_i - x_i^0)^2 \left.\frac{\partial g_j}{\partial x_i}\right|_0 \left(\frac{1}{U_i - x_i} - \frac{1}{U_i - x_i^0}\right)$$

$$- \sum_{-}(x_i^0 - L_i)^2 \left.\frac{\partial g_j}{\partial x_i}\right|_0 \left(\frac{1}{x_i - L_i} - \frac{1}{x_i^0 - L_i}\right)$$

$$\frac{\partial \tilde{g}_j}{\partial x_i} = \begin{cases} (U_i - x_i^0)^2 \left.\frac{\partial g_j}{\partial x_i}\right|_0 \frac{+1}{(U_i - x_i)^2} & \text{if} \quad \left.\frac{\partial g_j}{\partial x_i}\right|_0 > 0 \\ (-1)(x_i^0 - L_i)^2 \left.\frac{\partial g_j}{\partial x_i}\right|_0 \frac{-1}{(x_i - L_i)^2} & \text{if} \quad \left.\frac{\partial g_j}{\partial x_i}\right|_0 < 0 \end{cases}$$

$$\frac{\partial^2 \tilde{g}_j}{\partial x_i^2} = \begin{cases} (U_i - x_i^0)^2 \left.\frac{\partial g_j}{\partial x_i}\right|_0 \frac{+2}{(U_i - x_i)^3} & \text{if} \quad \left.\frac{\partial g_j}{\partial x_i}\right|_0 > 0 \\ (x_i^0 - L_i)^2 \left.\frac{\partial g_j}{\partial x_i}\right|_0 \frac{-2}{(x_i - L_i)^3} & \text{if} \quad \left.\frac{\partial g_j}{\partial x_i}\right|_0 < 0 \end{cases}$$

MMA approximation

SELECTION / UPDATE OF MOVING-ASYMPTOTES

- Idea:
  - If one observes an oscillating behavior of the design variables, increase the convexity, by reducing the slack between the design variable and the asymptotes U_i and L_i
  - If the convergence rate is (too) slow and monotonous, one can decrease the convexity and increase the distance between the point x_i and U_i and L_i
- Adaptation strategy needs additional information coming from convergence history
- Updating the moving asymptotes remains a difficult problem, with many (empirical or heuristic) answers adapted to particular problems

MMA approximation

UPDATE OF MOVING-ASYMPTOTES (Svanberg, 1987)

- Iterations k=1, 2

$$L_i^{(k)} = x_i^{(k)} - s_0(\bar{x}_i - \underline{x_i})$$
$$U_i^{(k)} = x_i^{(k)} + s_0(\bar{x}_i - \underline{x_i}) \qquad s_0 = 0.5$$

- Iterations k>2
  - if process is stable $(x_i^{(k-2)} - x_i^{(k-1)}).(x_i^{(k-1)} - x_i^{(k)}) \geq 0$

$$L_i^{(k)} = x_i^{(k)} - s_1(x_i^{(k-1)} - L_i^{(k-1)})$$
$$U_i^{(k)} = x_i^{(k)} + s_1(U_i^{(k-1)} - x_i^{(k-1)})$$

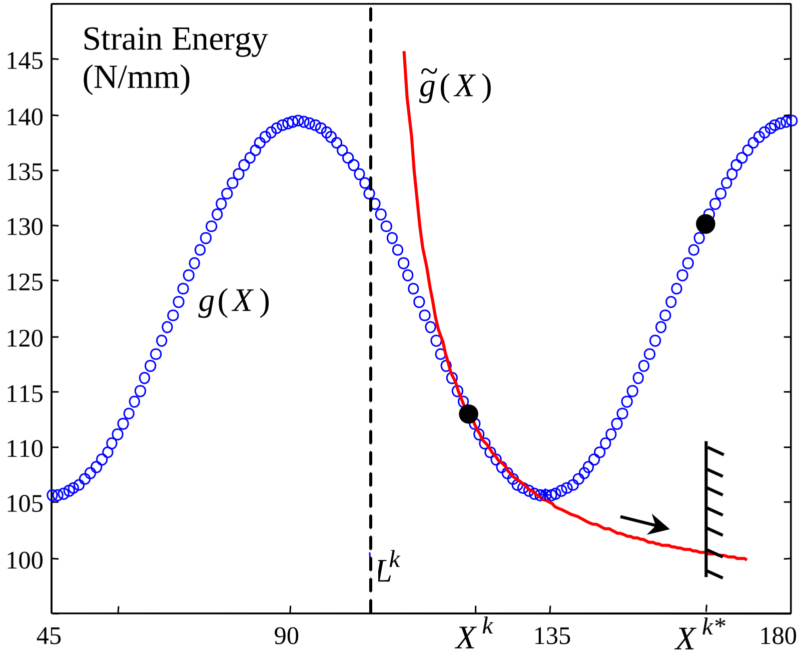
s_1= 1.2 to 1.05

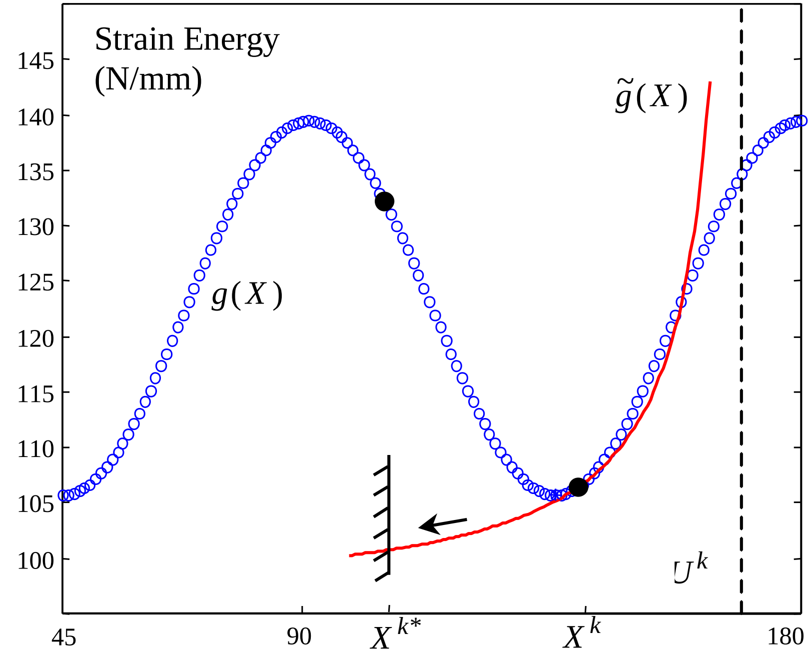
  - if process oscillates $(x_i^{(k-2)} - x_i^{(k-1)}).(x_i^{(k-1)} - x_i^{(k)}) < 0$

$$L_i^{(k)} = x_i^{(k)} - s_2(x_i^{(k-1)} - L_i^{(k-1)})$$
$$U_i^{(k)} = x_i^{(k)} + s_2(U_i^{(k-1)} - x_i^{(k-1)})$$

s_2=0.7 to 0.65

MMA approximation





Approximation of the strain energy in a two-ply symmetric laminate subject to shear load and torsion (Bruyneel and Fleury, 2000)

MMA approximation

- MMA as well as CONLIN are made of monotonous decreasing or increasing functions

- Solutions of the sub-problems can be unbounded in one direction. When the response functions are non monotonous like in composites or in shape optimization, this can generally lead to oscillations in which upper and lower asymptotes are alternatively activated. This also leads to oscillation of the optimization process.

- To prevent solutions from moving away from current design point, it is necessary to include move-limits

- Move-limits strategy suggested by Svanberg (1987)

$$max(\underline{x}_i, 0.9L_j + 0.1x_i^0) \ \leq \ x_i \ \leq \ min(\bar{x}_i, 0.1x_i^0 + 0.9U_i)$$

GCMMA approximation

- New extension of MMA (Svanberg, 1995) which activates simultaneously both asymptotes in order to create a non monotonous approximation

- Globally Convergent Method of Moving Asymptotes (GCMMA)

$$\tilde{g}_j(\mathbf{x}) = r_j^0 + \sum_{i=1}^{n} \frac{p_{ij}^{(k)}}{U_i - x_i} + \sum_{i=1}^{n} \frac{q_{ij}^{(k)}}{x_i - L_i}$$

where p_{ij} and q_{ij} are both different from zero

$$p_{ij}^{(k)} = (U_i^{(k)} - x_i^{(k)})^2 \left(\max\{0, \frac{\partial g_j(\mathbf{x}^{(\mathbf{k})})}{\partial x_i}\} + \frac{\rho_j^{(k)}}{2} (U_i^{(k)} - L_i^{(k)}) \right)$$

$$q_{ij}^{(k)} = (x_i^{(k)} - L_i^{(k)})^2 \left(\max\{0, -\frac{\partial g_j(\mathbf{x}^{(\mathbf{k})})}{\partial x_i}\} + \frac{\rho_j^{(k)}}{2} (U_i^{(k)} - L_i^{(k)}) \right)$$

GCMMA approximation

- The parameter ρ_j is a positive parameter which is precisely responsible for keeping both asymptotes in the same time.

UPDATE OF PARAMETER ρ

- The ρ_j parameter is updated according to the following rules:
  - For the first iteration (k=1)

$$\rho_j^{(1)} = \varepsilon \quad \forall\, j \in \{0, 1 \ldots, m\} \quad 0 < \varepsilon \ll 1$$

  - For other iterations (k>1),

$$\rho_j^{(k)} = 2\,\rho_j^{(k-1)} \qquad \text{if} \qquad \tilde{g}_j^{(k-1)}(\mathbf{x}^{(\mathbf{k})}) \;<\; g_j(\mathbf{x}^{(\mathbf{k})})$$

$$\rho_j^{(k)} = \rho_j^{(k-1)} \qquad \text{if} \qquad \tilde{g}_j^{(k-1)}(\mathbf{x}^{(\mathbf{k})}) \;\geq\; g_j(\mathbf{x}^{(\mathbf{k})})$$

A posteriori error of the GCMMA approximation in $x^{(k-1)}$ evaluated on the light of the new calculated real function value in $x^{(k)}$

GCMMA approximation

UPDATE OF MOVING ASYMPTOTES U_j and L_j

- Use original update scheme for U_j and L_j

- Except if

$$\tilde{g}_j^{(k-1)}(\mathbf{x}^{(\mathbf{k})}) \geq g_j(\mathbf{x}^{(\mathbf{k})}) \quad \forall j \in \{0, 1 \ldots, m\}$$

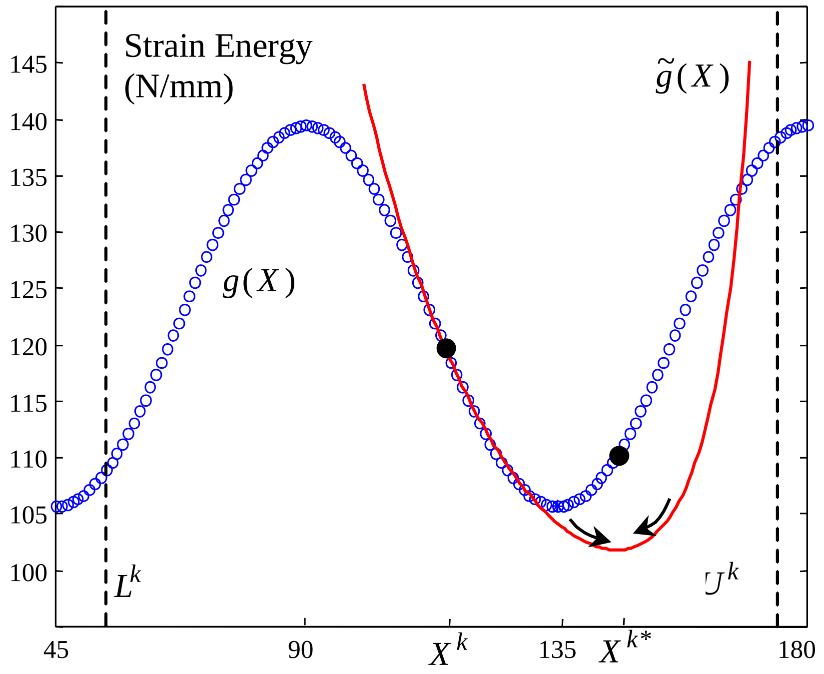
in order to prove the convergence, asymptotes have to be updated as follows

$$\begin{aligned} L_i^{(k)} &= x_i^{(k)} - (x_i^{(k-1)} - L_i^{(k-1)}) \\ U_i^{(k)} &= x_i^{(k)} + (U_i^{(k-1)} - x_i^{(k-1)}) \end{aligned}$$

CONVERGENCE PROOF

- With the following update scheme, Svanberg (1995) proved global convergence (i.e. convergence from any starting point) of the scheme
- However, when MMA converges, GCMMA is generally so conservative that pure MMA convergence rate is superior to GCMMA

GCMMA approximation



Approximation of the strain energy in a two plies symmetric laminate subject to shear load and torsion (Bruyneel and Fleury, 2000)

GCMMA second order

- As suggested in a technical note of Svanberg (1995), the non-monotonic parameter ρ can be replaced by using diagonal (non-mixed) second order derivatives. However the globally convergent character is lost.

$$
\begin{aligned}
p_{ij}^{(k)} &= \frac{(U_i^{(k)} - x_i^{(k)})^3}{2(U_i^{(k)} - L_i^{(k)})} \times \\
&\quad \left(2\frac{\partial g_j(\mathbf{x}^{(k)})}{\partial x_i} + (x_i^{(k)} - L_i^{(k)}) \frac{\partial^2 g_j(\mathbf{x}^{(k)})}{\partial x_i^2} \right) \\
q_{ij}^{(k)} &= \frac{(x_i^{(k)} - L_i^{(k)})^3}{2(U_i^{(k)} - L_i^{(k)})} \times \\
&\quad \left(-2\frac{\partial g_j(\mathbf{x}^{(k)})}{\partial x_i} + (U_i^{(k)} - x_i^{(k)}) \frac{\partial^2 g_j(\mathbf{x}^{(k)})}{\partial x_i^2} \right)
\end{aligned}
$$

Example and comparison of first order approximation schemes

- Constraint:

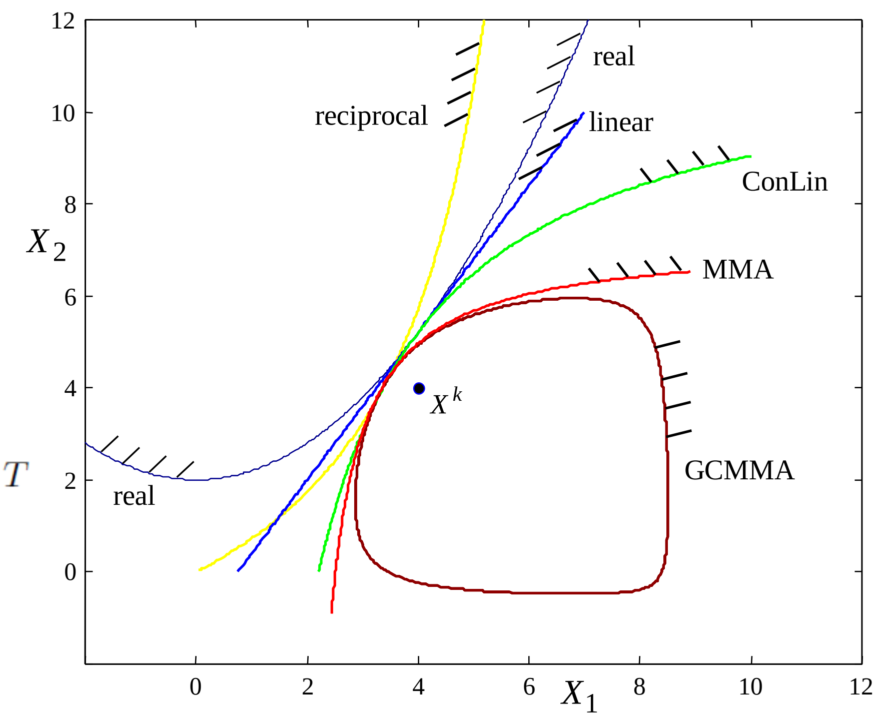
$$g(\boldsymbol{x}) = 5\,x_2 \;-\; x_1^2$$

- Derivatives

$$\frac{\partial g}{\partial x_1} \;=\; -2\,x_1 \qquad \frac{\partial g}{\partial x_2} \;=\; 5$$

- Local expansion at $(x_1^0, x_2^0) \;=\; (4,4)$

$$g(\boldsymbol{x}^{\mathbf{o}}) \;=\; 4 \qquad \nabla g(\boldsymbol{x}^{\mathbf{o}}) \;=\; [-8\;,\;5]^T$$



SECOND ORDER APPROXIMATION SCHEMES

Second order approximation schemes

- In first order approximation schemes:
  - Linear: no curvature
  - CONLIN: fixed curvature or use translation of design variable
  - MMA: curvature is adapted with moving asymptotes
  - GCMMA: curvature is ruled by asymptotes + ρ
- Conclusion: all these first order approximation techniques introduce *artificially* some curvature into explicit constraints.
- On the basis of only *first derivatives at the current point*, *estimation of the true curvature* is not easy at all.
- Curvature information is replaced by
  - The exploitation of our knowledge of the problem (but this is very problem dependent!)
  - The exploitation of another information e.g. the history of convergence, the fitting to other design points

Second order approximation schemes

- For second order schemes: the quality of the approximation is enhanced by using the *true second order* information.

- Main issues of second order approximation are:
  - being able to compute the second order sensitivity
  - being able to compute it with a reasonable numerical effort

    (computing and storing the second order derivatives can become quickly very cumbersome when the size of the problem increases)
  - being able to determine if this information is relevant

    (sometimes like in composite optimization or in perimeter constraint, true second order derivatives don't improve the quality of the approximation…)
  - when using full second order Hessian matrix, the approximation becomes not separable anymore. Dual methods are not so efficient. Is it possible to use only some kind of diagonal curvature information?

Quadratic separable approximations

- Quadratic approximation:
  - second order Taylor expansion
  - no more separable
- Quadratic separable approximation (Fleury, 1989):
  - neglect off-diagonal second order derivatives
  - introduce additional second order terms δ_{ii}

$$\tilde{g}(\mathbf{x}) = g(\mathbf{x_0}) \ + \ \sum_{i=1}^{n} \frac{\partial g(\mathbf{x_0})}{\partial x_i}(x_i - x_i^0) \ + \ \frac{1}{2}\sum_{i=1}^{n}(\frac{\partial^2 g(\mathbf{x_0})}{\partial x_i^2} + \delta_{ii})(x_i - x_i^0)^2$$

- Additional second order terms δ_{ii} reinforce the convexity and establish a trust region around the design point.

  They can be calculated to keep the unconstrained optimum within a given distance from current design point

$$|x_i^* - x_i^0| \ = \ |\frac{\partial g}{\partial x_i}/(\frac{\partial^2 g}{\partial x_i^2} + \delta_{ii})| \ < \ \alpha \ x_i^0$$

Generalized method of Moving asymptotes (GMMA)

- GMMA approximation (Smaoui et al. 1988) is given by

$$\tilde{g}(\mathbf{x^0}) = c_0 \ + \ \sum_{i=1}^{n} \frac{a_i}{x_i - b_i}$$

The 2n+1 parameters a_i, b_i and c_0 are given by matching the function value, the first derivatives and the second order diagonal derivatives.

They are given by:

$$\begin{aligned} a_i &= -(x_i^0 - b_i)^2 \ \frac{\partial g(\mathbf{x^0})}{\partial x_i} \\ b_i &= x_i^0 \ + \ 2 \ \frac{\partial g(\mathbf{x^0})}{\partial x_i} / \max(\epsilon, \frac{\partial^2 g(\mathbf{x^0})}{\partial x_i^2}) \qquad 0 < \epsilon \ll 1 \end{aligned}$$

- Generalization of MMA scheme since here each function has its own set of asymptotes

Approximation procedure of diagonal second order derivatives

- Theory of quasi-Newton updates preserving sparse structure (Thapa, 1981) has been adapted to diagonal pattern.
- In the framework of this theory, one can show that the best diagonal estimation is (Duysinx et al. 1995, 2000):

$$B_{ii} \simeq \frac{\dfrac{\partial g(\mathbf{x}^{(\mathbf{k})})}{\partial x_i} - \dfrac{\partial g(\mathbf{x}^{(\mathbf{k-1})})}{\partial x_i}}{x_i^{(k)} - x_i^{(k-1)}}$$

  This is the rather intuitive result of "making finite differences between the computed first derivatives in two successive iterations and ignoring the cross derivatives".
- This result needs to be adapted to structural optimization:
  - update in the reciprocal design space leads to more conservative schemes
  - initial Hessian estimate: CONLIN or MMA
  - damping of too large modifications of the curvature terms

Approximation procedure of diagonal second order derivatives

- GMMA and diagonal quasi-Newton update: DQNMMA
  - automatic selection of asymptotes based only on first order information

- Quadratic separable scheme and diagonal quasi-Newton: DQNQUA
  - automatic trust region for the approximation
  - even if the second order information is not exact, the approximation sketches the real function behavior (non monotonous)

- Convergence rates :
  - are very close to the performance of approximations using exact second order derivatives
  - are generally superior to first order approximations (CONLIN, MMA)

- Approximations are more sensitive to local optima and more fragile for difficult problems

GBMMA: A FAMILY OF MMA APPROXIMATIONS

Gradient Based MMA approximations

- Basically same general expression of the expansion as in GCMMA

$$\tilde{g}_j(\boldsymbol{x}) = g_j(\boldsymbol{x}^{(k)}) + \sum_{i=1}^{n} p_{ij}^{(k)} \left(\frac{1}{U_i^{(k)} - x_i} - \frac{1}{U_i^{(k)} - x_i^{(k)}}\right) + \sum_{i=1}^{n} q_{ij}^{(k)} \left(\frac{1}{x_i - L_i^{(k)}} - \frac{1}{x_i^{(k)} - L_i^{(k)}}\right)$$

- Asymptotes U_j and L_j updated according to Svanberg's general rule

$$L_i^{(k)} = x_i^{(k)} - (x_i^{(k-1)} - L_i^{(k-1)})$$
$$U_i^{(k)} = x_i^{(k)} + (U_i^{(k-1)} - x_i^{(k-1)})$$

- Determine parameters p_{ij} and q_{ij} :
  - don't use the non monotonic parameter ρ_j:
  - use **gradient and function value information at previous iteration point / estimation of the diagonal second order derivatives**

GBMMA1

- **Using the gradient at the previous iteration point**

- Determine p_{ij} and q_{ij} coefficients by matching the first order derivatives at current and previous iteration points:

$$\frac{\partial g_j(\boldsymbol{x}^{(k)})}{\partial x_i} = \frac{p_{ij}^{(k)}}{(U_i^{(k)} - x_i^{(k)})^2} - \frac{q_{ij}^{(k)}}{(x_i^{(k)} - L_i^{(k)})^2}$$

$$\frac{\partial g_j(\boldsymbol{x}^{(k-1)})}{\partial x_i} = \frac{p_{ij}^{(k)}}{(U_i^{(k)} - x_i^{(k-1)})^2} - \frac{q_{ij}^{(k)}}{(x_i^{(k-1)} - L_i^{(k)})^2}$$

- Solution of this linear system of equations: easy

GBMMA2

- **Using an estimation of the diagonal second order derivatives**
- As in second order GCMMA, one can determine p_{ij} and q_{ij} from diagonal second derivatives

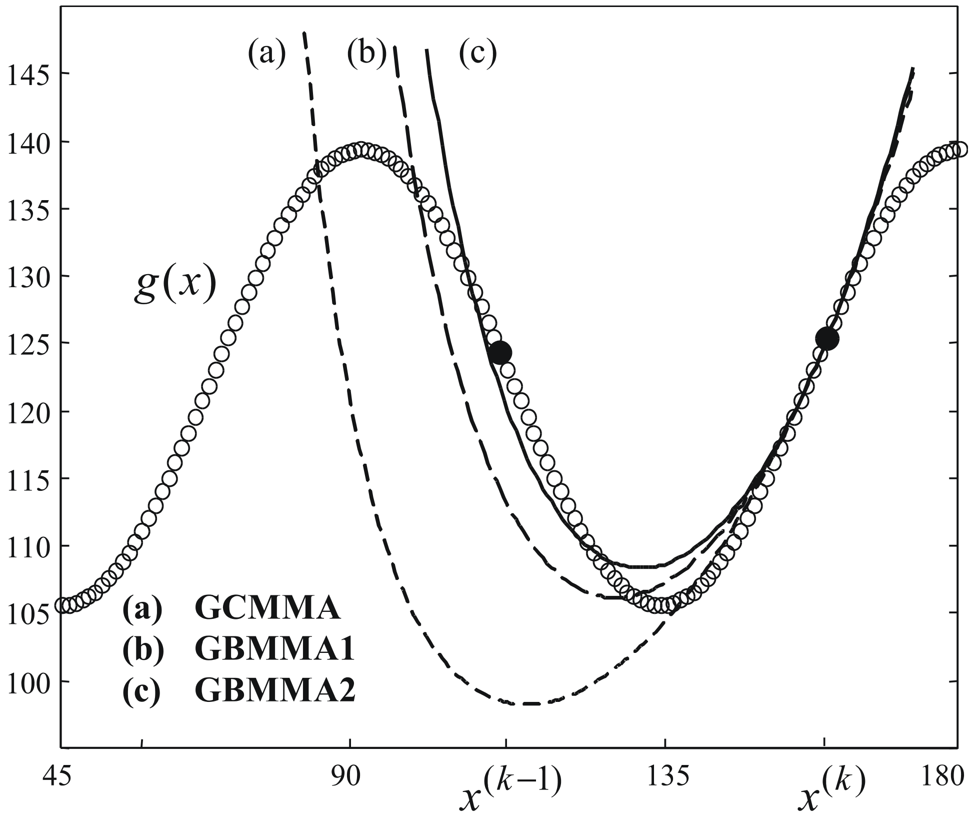
$$p_{ij}^{(k)} = \frac{(U_i^{(k)} - x_i^{(k)})^3}{2(U_i^{(k)} - L_i^{(k)})} \times \left(2\frac{\partial g_j(\mathbf{x}^{(k)})}{\partial x_i} + (x_i^{(k)} - L_i^{(k)})\frac{\partial^2 g_j(\mathbf{x}^{(k)})}{\partial x_i^2} \right)$$

$$q_{ij}^{(k)} = \frac{(x_i^{(k)} - L_i^{(k)})^3}{2(U_i^{(k)} - L_i^{(k)})} \times \left(-2\frac{\partial g_j(\mathbf{x}^{(k)})}{\partial x_i} + (U_i^{(k)} - x_i^{(k)})\frac{\partial^2 g_j(\mathbf{x}^{(k)})}{\partial x_i^2} \right)$$

- Estimate second derivatives from formula

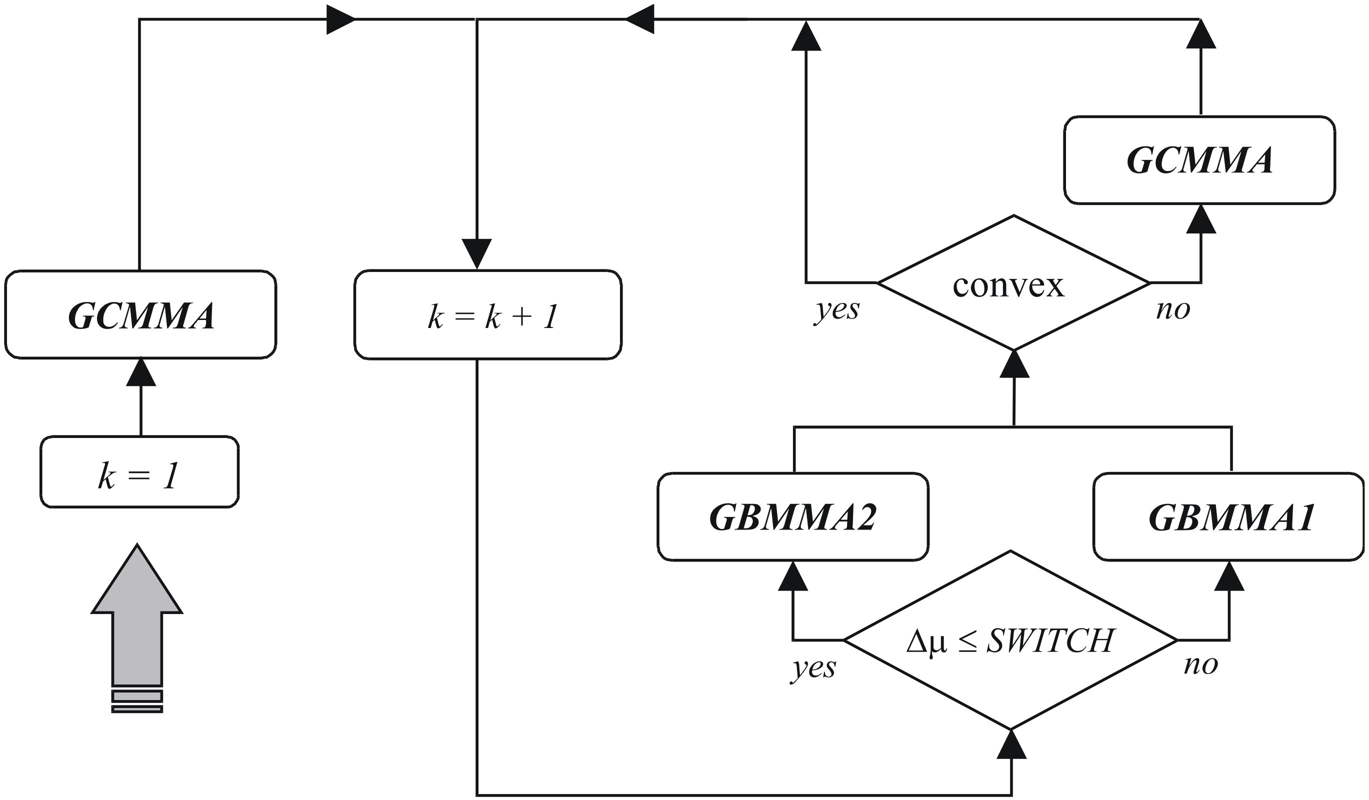
$$\frac{\partial^2 g_j(\boldsymbol{x}^{(k)})}{\partial x_i^2} \simeq \frac{\dfrac{\partial g_j(\boldsymbol{x}^{(k)})}{\partial x_i} - \dfrac{\partial g_j(\boldsymbol{x}^{(k-1)})}{\partial x_i}}{x_i^{(k)} - x_i^{(k-1)}}$$

GBMMA1 & GBMMA2



Approximation of the strain energy in a two plies symmetric laminate subject to shear load and torsion (Bruyneel and Fleury, 2000)

GBMMA automatic selection strategy



- Using the second order derivatives is only efficient when close to the optimum: introduce SWITCH parameter to move from GBMMA1 to GBMMA2

$$\frac{|x_i^{(k)} - x_i^{(k-1)}|}{\overline{x}_i - \underline{x}_i} \leq SWITCH$$

CONCLUSIONS

Advantages of Sequential Convex Programming in structural optimization

- Efficiency of dual solvers even with a large number of design variables
- Robust solution approach even for problems with a large number of constraints
- Improvement of convergence rates when using high quality / flexible approximation schemes (GBMMA)
- Flexibility and generality to solve various kinds of problems in topology: compliance, eigenfrequency, stress constraints, design of material, design of compliant mechanisms...
- Mathematical foundations to attack special problems: relaxing unfeasible constraints, perturbation of non-regular problems like singularity phenomenon of local constraints...

ANNEX 1: GCMMA CODE IN MATLAB BY SVANBERG

GCMMA code in Matlab by Svanberg

- For academic purpose, Svanberg wrote a Matlab version of its GCMMA solver: i.e. Globally convergent approximations + solver (IP method)

- GCMMA : mmasub function in Matlab

```
function [xmma,ymma,zmma,lam,xsi,eta,mu,zet,s,low,upp] = ...
mmasub(m,n,iter,xval,xmin,xmax,xold1,xold2, ...
f0val,df0dx,df0dx2,fval,dfdx,dfdx2,low,upp,a0,a,c,d);
```

- Solver : subsolv function (called from mmasub)

```
function [xmma,ymma,zmma,lamma,xsimma,etamma,mumma,zetmma,
smma] = ...
subsolv(m,n,epsimin,low,upp,alfa,beta,p0,q0,P,Q,a0,a,b,c,d);
```

GCMMA code in Matlab by Svanberg

- Solve canonical problems of the form :

$$
\begin{array}{lll}
\min & f_0(\mathbf{x}) + z + \sum_{j=1}^{m} (c_j y_j + \frac{1}{2} d_j y_j^2) & \\
\text{s.t.:} & f_j(\mathbf{x}) - a_j z - y_j \leq 0 & j = 1 \dots m \\
 & \underline{x}_i \ \leq \ x_i \ \leq \overline{x}_i & i = 1 \dots n \\
 & y_j \geq 0 & j = 1 \dots m \\
 & z \geq 0 &
\end{array}
$$

- With
  - f_0, f_1… f_m real functions, continuous and differentiable
  - a_i, c_i, d_i real non negative numbers, with $a_i+d_i>0$

GCMMA code in Matlab by Svanberg

- Classic problems of non-linear programming

$$
\begin{array}{lll}
\min & f_0(\mathbf{x}) & \\
\text{s.t.:} & f_j(\mathbf{x}) \leq 0 & j = 1 \ldots m \\
 & \underline{x}_i \ \leq \ x_i \ \leq \overline{x}_i & i = 1 \ldots n
\end{array}
$$

- Parameter choice:

$$
\begin{array}{ll}
a_j = 0 & j = 1 \ldots m \\
d_j = 0 & j = 1 \ldots m \\
c_j = \text{big} & j = 1 \ldots m
\end{array}
$$

GCMMA code in Matlab by Svanberg

- Least square problem

$$\begin{array}{lll} \min & \sum_{k=1}^{p} (h_k(\mathbf{x}))^2 & \\ \text{s.t.:} & g_j(\mathbf{x}) \le 0 & j = 1 \ldots q \\ & \underline{x}_i \;\le\; x_i \;\le \overline{x}_i & i = 1 \ldots n \end{array}$$

- Play with the quadratic term in y_j and the coefficient d_j
- Parameter choice:

$$\begin{array}{rlrl} m = 2p + q & & a_j = 0 & j = 1 \ldots m \\ f_0(\mathbf{x}) = 0 & & d_k = 2 & k = 1 \ldots 2p \\ f_k(\mathbf{x}) = h_k(\mathbf{x}) & k = 1 \ldots p & d_{2p+j} = 0 & j = 1 \ldots q \\ f_{p+k}(\mathbf{x}) = -h_k(\mathbf{x}) & k = 1 \ldots p & c_k = 0 & k = 1 \ldots 2p \\ f_{2p+j}(\mathbf{x}) = g_j(\mathbf{x}) & j = 1 \ldots q & c_{2p+j} = \text{big} & j = 1 \ldots q \end{array}$$

GCMMA code in Matlab by Svanberg

- L_1 minimization problem

$$
\begin{array}{lll}
\min & \sum_{k=1}^{p} |h_k(\mathbf{x})| & \\
\text{s.t.:} & g_j(\mathbf{x}) \leq 0 & j = 1 \ldots q \\
 & \underline{x}_i \;\leq\; x_i \;\leq \overline{x}_i & i = 1 \ldots n
\end{array}
$$

- Play with the linear terms in y_j and the coefficient c_j
- Parameter choice:

$$
\begin{array}{rllll}
m = 2p + q & & & & \\
f_0(\mathbf{x}) = 0 & & a_j = 0 & j = 1 \ldots m \\
f_k(\mathbf{x}) = h_k(\mathbf{x}) & k = 1 \ldots p & d_j = 0 & j = 1 \ldots m \\
f_{p+k}(\mathbf{x}) = -h_k(\mathbf{x}) & k = 1 \ldots p & c_k = 1 & k = 1 \ldots 2p \\
f_{2p+j}(\mathbf{x}) = g_j(\mathbf{x}) & j = 1 \ldots q & c_{2p+j} = \text{big} & j = 1 \ldots q
\end{array}
$$

GCMMA code in Matlab by Svanberg

- L_∞ minimization problems or min max problems

$$
\begin{array}{llr}
\min & \max_{k=1\ldots p} |h_k(\mathbf{x})| & \\
\text{s.t.:} & g_j(\mathbf{x}) \leq 0 & j = 1 \ldots q \\
 & \underline{x}_i \;\leq\; x_i \;\leq \overline{x}_i & i = 1 \ldots n
\end{array}
$$

- Play with the linear terms in z and the coefficient a_j
- Parameter choice:

$$
\begin{array}{rlll}
m = 2p + q & & & \\
f_0(\mathbf{x}) = 0 & & a_k = 1 & k = 1 \ldots 2p \\
f_k(\mathbf{x}) = h_k(\mathbf{x}) & k = 1 \ldots p & a_{2p+j} = 0 & j = 1 \ldots q \\
f_{p+k}(\mathbf{x}) = -h_k(\mathbf{x}) & k = 1 \ldots p & d_j = 0 & j = 1 \ldots m \\
f_{2p+j}(\mathbf{x}) = g_j(\mathbf{x}) & j = 1 \ldots q & c_j = \text{big} & j = 1 \ldots m
\end{array}
$$