Optimization of multibody systems using sequential linear programming

Optimization of multibody systems using sequential linear programming

L.F.P. Etman, E.J.R.W. Thijssen,
A.J.G. Schoofs, D.H. van Campen

WFW reportnumber: 94.007

Optimization of multibody systems using sequential linear programming

L.F.P. Etman E.J.R.W. Thijssen A.J.G. Schoofs
D.H. van Campen
Stevin Centre, Eindhoven University of Technology, PO box 513, the Netherlands

January 12, 1994

Abstract

Design optimization is far less developed for multibody analysis than for structural analysis. However, multibody design problems can be solved by optimization strategies applied in structural analysis. To illustrate this, a design optimization tool has been developed for a multibody analysis software package. It is based on a linear approximation concept. An optimization process results that consists of a sequence of linear programming problems. The design optimization tool has been successfully tested for three multibody design examples.

1 Introduction

Multibody software packages are generally capable of analysing a user-supplied mechanical model that consists of a finite number of rigid or flexible bodies. Most packages do not include any routines to modify and improve the multibody system design. However, it is a great help for the designer to include design optimization tools in the software. In packages with such tools, the design problem is converted into an optimization problem. Then, for example, a flexible robot system of minimum weight can be designed.

Design optimization tools have been developed mainly in the field of finite element structural analysis. These tools do not usually directly solve the optimization problem, but use a suitable approximation concept to interface structural analysis software and mathematical programming algorithm. This avoids programming difficulties and is computationally more convenient (Haftka and Gürdal, 1992). The basic principle is to generate approximations of objective function and constraints in a certain part of the design space, and to solve the optimum point for this approximate optimization problem. The approximate problem can be easily solved using a mathematical programming algorithm, since the approximate objective function and constraints are explicitly known. Barthelemy and Haftka (1993) review both the basic and the more recently developed approximation concepts in structural optimization.

Approximate optimization strategies can also be applied to multibody system analysis software. Like structural analysis, multibody system analysis is computationally expensive. Therefore, the number of design analyses should be limited. Many approximation concepts use design sensitivity information. For the finite element structural analysis case, sensitivity analysis can be performed at reduced costs compared with finite difference sensitivities (Haug et al., 1986). However, for multibody systems cost-effective design sensitivity analysis methods are not yet generally available, but are currently being developed by several authors including Haug (1987), Bestle and Seybold (1992) and Bestle and Eberhard (1992).

In the present research, a design optimization tool has been developed for the multibody analysis code MECANO of the SAMCEF finite element software (Samtech, 1991). The optimization tool has been written separately from the multibody software, since we had no access to the source code. It is based on a linear Taylor series approximation concept. Sensitivities have been calculated by finite differences. The design optimization tool has been successfully tested for three multibody design examples.

2 Sequential linear approximate optimization

The general optimization problem is formulated as: find the set of n design variables $\mathbf{x}$, that will minimize the objective function:

$$F_{obj}(\mathbf{x}) \tag{1}$$

subject to the constraints:

$$g_h(\mathbf{x}) \leq c_h \qquad h = 1, ..., m \tag{2}$$

and side-constraints:

$$x_k^l \leq x_k \leq x_k^u \qquad k = 1, ...n \tag{3}$$

The scalar x_k is the k th element of the design vector $\mathbf{x}$. The side-constraints define the design space, i.e. the region in which is searched for an optimum. Often, the objective function and constraints are not explicitly known and have to be computed from a multibody analysis at a certain design point.

Now a linear approximation concept is introduced. A linear approximation of objective function or constraint is a first-order Taylor series approximation based on function value and derivative values with respect to the design variables in a single point $\mathbf{x}_0$ of the design space. Around this design point, the linear approximate optimization problem consists of the minimization of the approximate objective function:

$$\tilde{F}_{obj}(\mathbf{x}) = F_{obj}(\mathbf{x}_0) + \sum_{i=1}^{n}(x_i - x_{0i}) \left(\frac{\partial F_{obj}}{\partial x_i}\right)_{\mathbf{x}_0} \tag{4}$$

subject to the constraints:

$$\tilde{g}_h(\mathbf{x}) = g_h(\mathbf{x}_0) + \sum_{i=1}^{n}(x_i - x_{0i}) \left(\frac{\partial g_h}{\partial x_i}\right)_{\mathbf{x}_0} \leq c_h \qquad h = 1, ..., m \tag{5}$$

in a limited part of the design space:

$$\alpha_k^l \leq \alpha_k \leq \alpha_k^u \qquad k = 1, ...n \tag{6}$$

The size of the search subregion is restricted by movelimits α_k^l and α_k^u , since the approximations are only locally valid.

Using a local approximation concept, a sequential approximate optimization process results (see for example Vanderplaats (1993)). The process starts with the multibody analysis of the initially proposed design. Then, all constraint functions are evaluated, and only those which are critical or potentially critical are retained for further consideration. Gradients of the objective function and the retained constraints are computed by means of a sensitivity analysis. These derivatives, together with the function values, are used to generate an approximate optimization problem in the search subregion bounded by the movelimits. For the linear approximation concept, this results in a linear programming problem. A newly proposed design follows from the solution of the approximate optimization problem, at which a new cycle of approximation and optimization can be started. This process is repeated until an acceptable optimum is reached. An extra design variable may be added to avoid problems with an infeasible starting design (Haftka and Gürdal, 1992).

In the case of a linear approximation concept, a correct choice of the size of the search subregion is important for a good convergence of the optimization process. Large movelimits can cause the solution process to oscillate, while small movelimits slow down the convergence rate. In this study, the movelimit values are related to the magnitude of the design variables, the quality of the approximations and the rate of convergence.

For every optimization cycle, the quality of the approximations can be checked by comparing the approximate objective and constraint values for the newly proposed design with the corresponding multibody analysis values. Differences between the approximated and calculated values are measures of the quality of the generated approximations. Therefore, we define the following relative errors of the p th cycle:

$$e_o^{(p)} = \left| \frac{\tilde{F}_{obj}^{(p)}(\mathbf{x}_*^{(p)}) - F_{obj}(\mathbf{x}_*^{(p)})}{F_{obj}(\mathbf{x}_*^{(p)})} \right| \tag{7}$$

$$e_h^{(p)} = \left| \frac{\tilde{g}_h^{(p)}(\mathbf{x}_*^{(p)}) - g_h(\mathbf{x}_*^{(p)})}{g_h(\mathbf{x}_*^{(p)})} \right| \qquad h = 1, ..., m \tag{8}$$

where $\mathbf{x}_*^{(p)}$ is the proposed design computed from the approximate optimization problem of the p th cycle.

The optimization is started with movelimits being 10 to 30 % of the magnitude of the design variables. The movelimits are decreased or enlarged, depending on the calculated errors. During cycles with steady decrease of objective function value, the movelimit strategy tries to keep the errors within in a range of 10 to 25 % . Near the optimum, a higher accuracy is desired. The movelimits should be decreased whenever the convergence slows down or oscillations occur. A proposed design is rejected if the errors are too large (> 40 %) or too high an infeasibility occurs (> 10 %) starting from a feasible or nearly feasible design. Then, the optimization cycle is restarted with smaller movelimits.

3 Objective function and constraints

In multibody optimum design, constraints are usually time-dependent. A time-dependent constraint:

$$g(\mathbf{x}, t) \leq c \qquad 0 \leq t \leq t_f \tag{9}$$

must be satisfied from $t = 0$ to final time t_f. To get rid of the time-dependency, the time interval is discretized in n_t time points (Haftka and Gürdal, 1992). Then, the original constraint is replaced by n_t constraints:

$$g_i(\mathbf{x}, t_i) \leq c \qquad i = 1, ..., n_t \tag{10}$$

To avoid large constraint violation between two time points, the distribution of time points must be sufficiently dense.

Discretizing time-dependent constraints causes a large increase in the number of constraints in the optimization problem. Since the computational costs of optimization are also expected to increase, it is often argued that equivalent constraints should be introduced, which remove time-dependency without increasing the number of constraints. Most of these equivalent constraints can be written in a form such as:

$$\bar{g}(\mathbf{x}) = \int_0^{t_f} p(\mathbf{x}, t)dt < c \tag{11}$$

with $p(\mathbf{x}, t)$ some function of $g(\mathbf{x}, t)$ (Haftka and Gürdal, 1992).

We believe that constraint discretization can still be successfully applied when using sequential approximate optimization. The total number of constraints may increase significantly, but the number of critical and potentially critical constraints to be approximated will increase only slightly. Therefore, the solution costs of the approximate optimization problem are expected to stay negligible compared with the computational expense of multibody system analysis and design sensitivity analysis.

The objective function is often defined as a max-value operation of some time-dependent function (e.g. acceleration):

$$F_{obj}(\mathbf{x}) = \max_{t \in [0, t_f]} f(\mathbf{x}, t) \tag{12}$$

However, the max-value operation usually introduces discontinuities in the derivatives of the objective function. In that case, a linear approximation may be less valuable. Therefore, the original objective function (12) is transformed into an alternative objective function:

$$F_{obj}(\mathbf{x}) = x_{n+1} \tag{13}$$

with an additional constraint:

$$f(\mathbf{x}, t) < x_{n+1} \qquad 0 \le t \le t_f \tag{14}$$

(see e.g. Haftka and Gürdal, 1992, and Haug and Arora, 1979). Addition of one extra design variable results in a smooth optimization problem, such that the linear approximation concept can still be effectively applied.

4 Test examples

4.1 Optimization of a vehicle suspension system

A five degree of freedom vehicle suspension system, shown in figure 1, is designed to have an optimum response to a certain road condition. Mass m_1 represents driver and seat, and is supported to the chassis (m_2) by a spring and a damper. Two spring-damper sets connect chassis with front and rear wheel axles. Wheel and axle are modeled by a mass, a linear spring and linear damper. In Haug and Arora (1979) the exact problem description can be found.

The optimum design problem is defined as to find the stiffness coefficients k_1, k_2 and k_3, and damping coefficients c_1, c_2 and c_3 such that the maximum acceleration of the driver's seat is minimized. Constraints are the maximum relative displacements between: chassis

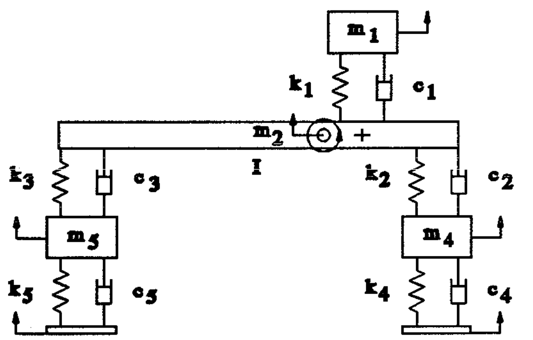


Figure 1: Five degree of freedom vehicle model.

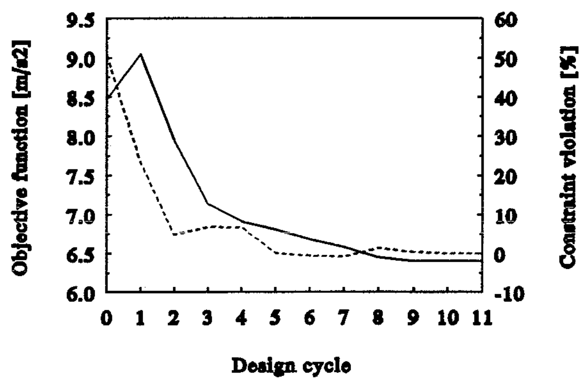


Figure 2: Optimization history of the vehicle model. The solid and dotted line represent respectively the objective function and the constraint violation.

x and $F_{obj}(\mathbf{x})$	Initial design	Optimum design	
		Haug et al	this paper
k_1 $[kNm^{-1}]$	17.5	8.76	8.76
k_2 $[kNm^{-1}]$	52.5	35.0	35.0
k_3 $[kNm^{-1}]$	52.5	42.4	35.0
c_1 $[kNsm^{-1}]$	1.75	2.26	7.27
c_2 $[kNsm^{-1}]$	4.38	13.6	13.5
c_3 $[kNsm^{-1}]$	4.38	14.0	14.0
F_{obj} $[ms^{-2}]$	8.45	6.46	6.40

Table 1: Initial and optimum design of the vehicle model

and driver's seat, chassis and front and rear wheels, and road surface and front and rear wheels. By addition of an extra design variable and constraint, the max-value operator is removed from the objective function. Every time-dependent constraint is discretized into 101 time-point constraints.

Within ten approximate optimization cycles, the maximum acceleration is reduced from 8.5 to 6.4 ms^{-2} when starting from the initial design given in table 1. In figure 2 objective function and constraint violation are plotted as a function of the cycle design number. The constraint violation of the optimum design is less than 0.2 %. This optimum differs from the design found by Haug and Arora (1979) for stiffness coefficient k_3 and damping coefficient c_1 (see table 1). However, the minimum objective function value is nearly the same. Probably a different path in the design space has been followed . It is remarked that all maximum accelerations are calculated with the same time discretization of 101 time-points. Therefore, the minimum objective function value reported in Haug and Arora (1979) is slightly different from the value in table 1.

4.2 Design of a landing gear mechanism

A four-bar mechanism has to be designed to move a landing gear plotted in figure 3 from a given start position to a desired end position (Hansen, 1992). The mechanism consists of four bodies: three rigid bars and the aeroplane, which are connected to each other by revolute joints. Body 2 of the mechanism is attached to the landing gear, such that a rotation of bar 1 results in a movement of the landing gear.

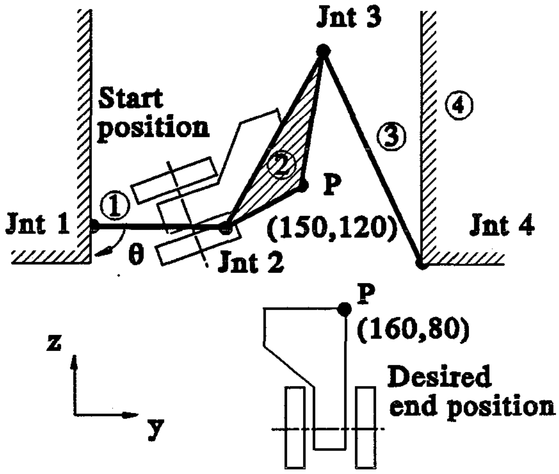


Figure 3: A four-bar mechanism has to move the landing gear from start to end position.

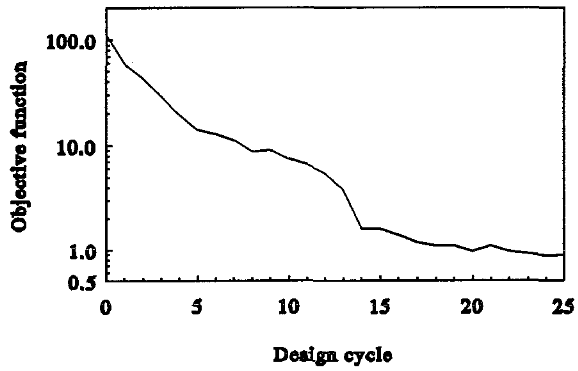


Figure 4: Optimization history of the landing gear mechanism.

To search for an appropriate mechanism an optimization problem is formulated. Design variables are the coordinates (y_k, z_k) $k = 1,..,4$ of the revolute joints and the rotation θ of bar 1. The objective is to minimize the distance Δp between the final position of point P attained by the mechanism and the desired final position, while the difference $\Delta\phi$ of attained and desired rotation of the landing gear should be zero. The starting position of point P has co-ordinates $(150, 120)$. Desired end position and rotation of P are respectively $(160, 80)$ and 1.222 *rad*.

The optimization problem is: find the design vector $\mathbf{x}^T = [\, y_1\ z_1\ y_2\ z_2\ y_3\ z_3\ y_4\ z_4\ \theta \,]$ that minimizes:

$$F_{obj}(\mathbf{x}) = \sqrt{(\Delta p)^2 + \alpha(\Delta\phi)^2} \tag{15}$$

with $x_1 \leq 100$, $y_1 \geq 100$, $x_4 \geq 180$ and $y_4 \geq 100$. Equality constraint $\Delta\phi = 0$ is incorporated in the objective function by exterior penalty. Penalty parameter α is set to 5000 and kept constant during the optimization. For this parameter value, $(\Delta p)^2$ and $(\Delta\phi)^2$ are of the same order of magnitude for the initial mechanism design, with a slight emphasis on the equality constraint. This initial design, given in table 2, is far from the desired movement.

In figure 4 the objective function is plotted as a function of the design cycle number. The minimization of Δp appears to be most difficult. The mechanism of the 25 th cycle attains an end position of point P of $(159.2, 80.3)$ and a rotation of the landing gear of 1.219 *rad*. This optimum design is given in table 2.

Remark that no constraints are included to avoid a collision of the landing gear with the aeroplane or a locking of the mechanism during rotation of bar 1. For the optimization results

Design variable	Initial design	Optimum design
y_1	100.0	99.23
z_1	110.0	100.0
y_2	130.0	127.2
z_2	110.0	139.4
y_3	160.0	173.7
z_3	155.0	184.8
y_4	180.0	185.4
z_4	100.0	104.1
θ	2.000	1.747

Table 2: Initial and optimum design of the landing gear mechanism

reported here, no locking has occurred. Still, it is important to take care of these locking constraints. Since they are explicitly known, they can be directly incorporated in the optimization problem if necessary, and do not have to be approximated. However, in our program we have to linearize all constraints because of the linear programming algorithm. During sequential linear programming often constraint violations occur, and violation of locking constraints will in our case abort the MECANO-analysis and stop the optimization. Therefore, a non-linear instead of a linear programming algorithm should be used to avoid the approximation of the locking constraints.

4.3 Stress constrained design of a four-bar mechanism

A four-bar mechanism is designed for minimum mass (Sohoni and Haug, 1982). The input crank of the mechanism shown in figure 5 rotates at a constant angular velocity of $10\pi\, rads^{-1}$. The mechanism consists of three mobile links of circular cross section, each with a modulus of elasticity E of $6.895\,10^{10}\ Pa$, a mass density ρ of $2757\ kgm^{-3}$, and a stress upperbound σ_a of $2.758\,10^{7}\ Pa$. The lengths of the bars are respectively $l_1 = 0.3048\,m$, $l_2 = 0.9144\,m$, $l_3 = 0.762\,m$ and $l_4 = 0.9144\,m$. Design variables are the cross sectional areas of the mobile links, with the constraints being the maximum bending stresses in these links.

Each mobile link is modeled by six beam elements. MECANO can compute the bending moments at every node. The maximum bending stress in every link can then be calculated from:

$$(\sigma_b)_i = \frac{4\sqrt{\pi}}{{x_i}^{3/2}}|M_i| \qquad i = 1, 2, 3 \tag{16}$$

with

i = the body number

x_i = the cross sectional area of body i

M_i = the maximum bending moment on body i

$(\sigma_b)_i$ = the maximum bending stress in body i

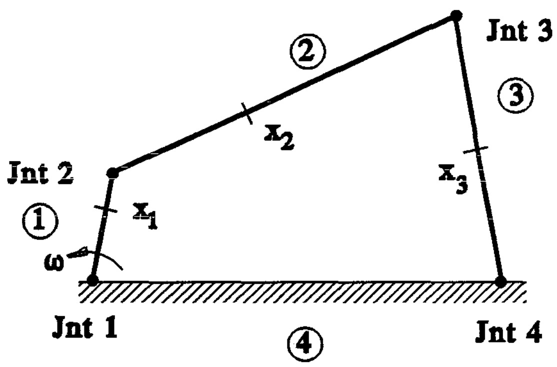


Figure 5: Stress constrained four-bar mechanism.

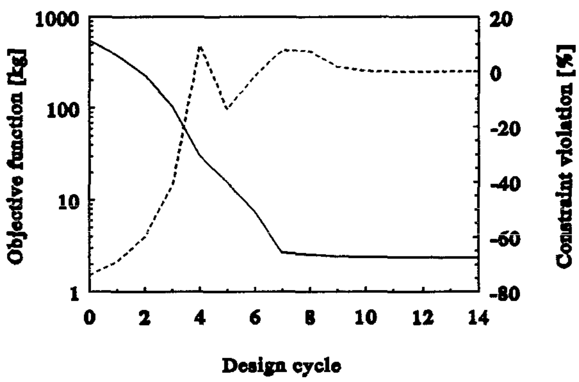


Figure 6: Optimization history of the stress constrained four-bar mechanism. The solid and dotted line represent respectively the objective function and the constraint violation.

x and $F_{obj}(\mathbf{x})$	Initial design	Optimum design	
		Sohoni et al	this paper
x_1 $[m^2]$	$1.0\,10^{-1}$	$1.03\,10^{-3}$	$9.65\,10^{-4}$
x_2 $[m^2]$	$1.0\,10^{-1}$	$6.21\,10^{-4}$	$3.92\,10^{-4}$
x_3 $[m^2]$	$1.0\,10^{-1}$	$1.17\,10^{-4}$	$2.69\,10^{-4}$
$F_{obj}(\mathbf{x})$ $[kg]$	546	2.69	2.36

Table 3: Initial and optimum design of the stress constrained four-bar mechanism

The optimum design problem is now to find the cross sectional areas x_1, x_2 and x_3 such that the mass

$$F_{obj}(\mathbf{x}) = \rho(l_1 x_1 + l_2 x_2 + l_3 x_3) \tag{17}$$

is minimized, subject to the stress constraints

$$\sigma_i(\mathbf{x}, t) \leq \sigma_a \qquad i = 1, 2, 3 \qquad 0.3 \leq t \leq 0.5s \tag{18}$$

The time interval is discretized into 101 time-points. It exactly covers one period of the steady-state solution after the transient has died away.

Mass and maximum constraint violation are plotted in figure 6 as a function of the design cycle number. The optimum design is given in table 3 completed with the design reported by Sohoni and Haug. The optimum objective function values are comparable, but the design variable values are different. These deviations are probably caused by the different kind of stress analysis used by Sohoni and Haug.

5 Conclusion and discussion

Approximation concepts can be effectively used for the optimization of multibody systems. A design optimization tool has been developed based on sequential linear approximate optimization. Solution costs of the approximate optimization problem do not significantly increase by discretizing time-dependent constraints, since only critical and potentially critical constraints have to be taken into account and approximated. In any case, our experience is that the computational expense of the linear programming algorithm is much smaller than the expense of design analysis and design sensitivity analysis, even without constraint deletion.

Linear approximation of objective function and constraints is the most simplest approximation concept. To improve the approximation, in structural optimization often suitable intermediate design variables and intermediate responses are introduced. It has to be investigated whether intermediate variables and response quantities can be found, that can be effectively used for approximate optimization of multibody systems. For example, in the case of the landing gear mechanism design of section 4.2, one can think of approximating the y and z coordinate of point P attained by the mechanism instead of the original objective function. In section 4.3 the bending moments in the nodes can be used as intermediate variables and be approximated instead of the maximum bending stresses.

If a local approximation concept is applied, design sensitivities are required. To calculate these derivatives, the finite difference method is rather cost-ineffective, because of the computational expense of a multibody analysis. Therefore, design sensitivity analysis methods have to be developed which can compute the sensitivities more efficiently without substantial loss of accuracy.

References

Afimawala, K.A.; Mayne, R.W. 1974: Optimum design of an impact absorber. J. Eng. Ind. 96, 124-130.

Barthelemy, J.-F.M.; Haftka, R.T. 1993: Approximation concepts for optimum structural design - a review. Struct. Optim. 5, 129-144.

Bestle, D.; Eberhard P. 1992: Analyzing and optimizing multibody systems. Mech. Struct. Machines 20, 67-92.

Bestle, D.; Seybold, J. 1992: Sensitivity analysis of constrained multibody systems. Arch. Appl. Mech. 62, 181-190.

Haftka, R.T.; Gürdal, Z. 1992: Elements of structural optimization. Dordrecht: Kluwer.

Hansen, M.R. 1992: Development and implementation of methods for analysis and synthesis of mechanical mechanisms, Ph-D thesis, Special report no. 15. Aalborg: University press.

Haug, E.J. 1987: Design sensitivity analysis of dynamic systems. In: Mota-Soares, C.A. (ed.) Computer aided design: Structural and Mechanical systems. Berlin: Springer.

Haug, E.J.; Arora, J.S. 1979: Applied Optimal Design. New York: John Wiley & Sons.

Haug, E.J.; Choi, K.K.; Komkov, V. 1986: Design Sensitivity Analysis of Structural Systems. Orlando: Academic Press, Inc.

Samtech 1991: Samcef user manual. Liège.

Sohoni, V.N.; Haug, E.J. 1982: A State Space Technique for Optimal Design of Mechanisms. J. Mech. Des. 104, 792-798.

Vanderplaats, G.N. 1992: Thirty years of modern structural optimization. Adv. Eng. Software 16, 81-88.

Vehicle

==> relative move limit
==> automatic move limit strategy
==> additional design variable for infeasibility
==> additional design varaible and constraint to eliminate max-value operator
==> 1 two-sided max-value constraint --> 2 one-sided max-value constraints
==> 5 two-sided constraints --> 10 one-sided constraints
(two-sided constraint: -sa<s<sa)

Design variables:

k1=10*x(1) [lb/in]
k2=20*x(2) [lb/in]
k3=20*x(3) [lb/in]
c1=x(4) [lb-sec/in]
c2=2*x(5) [lb-sec/in]
c3=2*x(6) [lb-sec/in]
xinfeas=x(7)
xmaxval=10*x(8) [in/sec2]

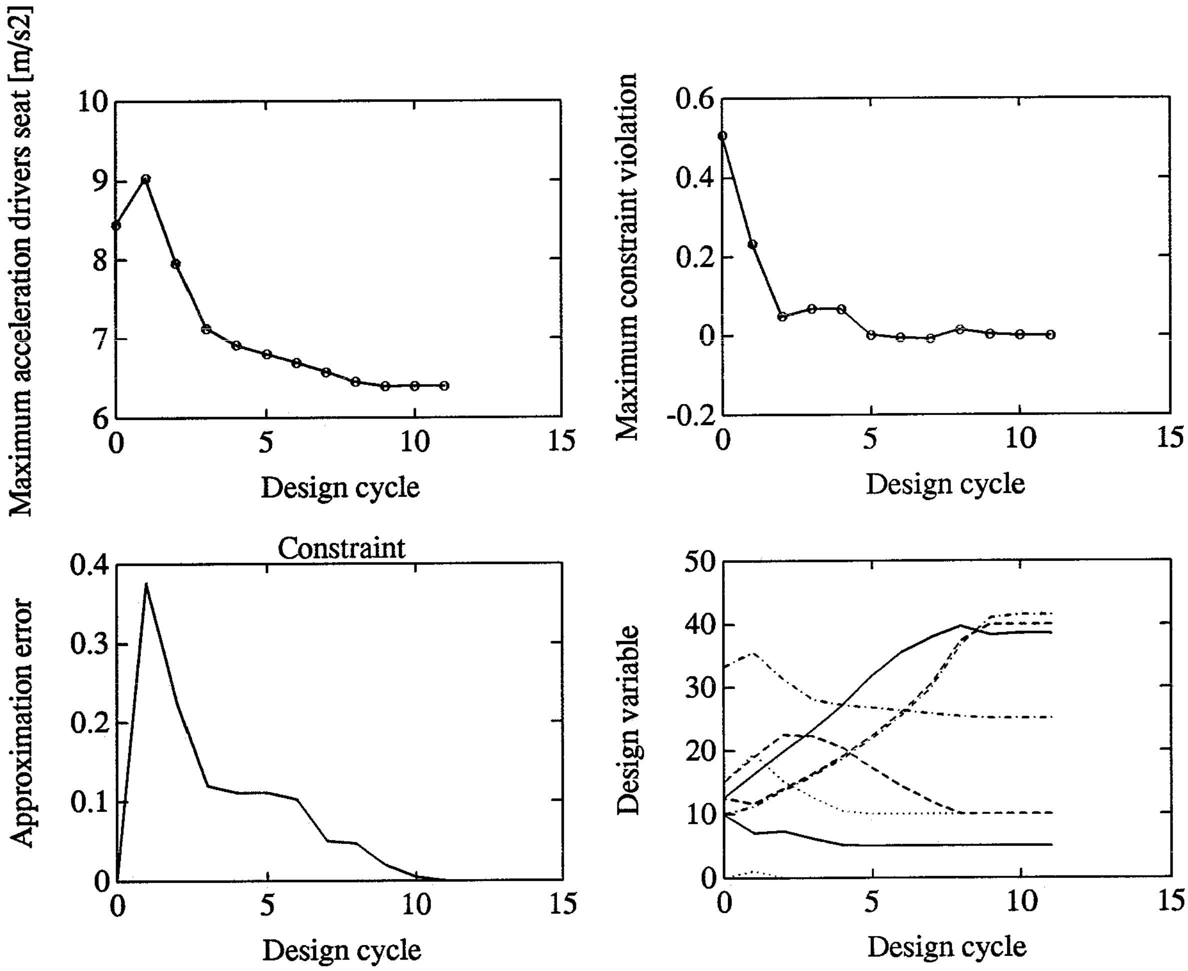
Optimum:

1. The approximated constraint 33 is active at its upper bound
2. The approximated constraint 34 is active at its upper bound
3. The approximated constraint 440 is active at its upper bound

1,2 = max-value constraint
3 = relative displacement constraint between drivers seat and chassis

L	ML [max %]	MLact	F	Fben	no	max(G) [%]	no	max(Gerr) [%]
1	.000E+00	0	.333E+02	.000E+00	440	.507E+00	0	.000E+00
2	.300E+00	1	.356E+02	.356E+02	357	.233E+00	470	.377E+00
3	.225E+00	1	.313E+02	.313E+02	439	.487E-01	235	.224E+00
4	-.169E+00	1	.281E+02	.281E+02	35	.679E-01	360	.120E+00
5	.169E+00	1	.272E+02	.272E+02	34	.673E-01	35	.110E+00
6	.169E+00	1	.268E+02	.268E+02	34	.121E-02	237	.110E+00
7	.169E+00	1	.263E+02	.263E+02	440	-.563E-02	242	.101E+00
8	.169E+00	1	.259E+02	.259E+02	440	-.800E-02	1058	.496E-01
9	.225E+00	1	.254E+02	.254E+02	34	.139E-01	33	.461E-01
10	.300E+00	1	.252E+02	.252E+02	440	.335E-02	33	.189E-01
11	.300E+00	1	.252E+02	.252E+02	440	.202E-03	847	.463E-02
12	.225E+00	1	.252E+02	.252E+02	34	.213E-04	852	.256E-03

L	X(1)	X(2)	X(3)	X(4)	X(5)	X(6)	X(7)	X(8)
1	10.00	15.00	15.00	10.00	12.50	12.50	0.000	33.26
2	7.000	18.98	19.50	11.19	16.25	11.64	1.061	35.58
3	7.272	22.50	15.11	13.71	19.91	13.98	0.01615	31.32
4	6.116	22.30	12.56	16.02	23.27	16.34	0.000	28.06
5	5.084	20.44	10.44	18.72	27.19	19.09	0.000	27.22
6	5.000	17.21	10.00	21.88	31.78	22.31	0.000	26.79
7	5.000	14.30	10.00	25.58	35.53	26.08	0.000	26.35
8	5.000	11.89	10.00	29.89	37.85	30.48	0.000	25.88
9	5.000	10.00	10.00	36.62	39.71	37.34	0.000	25.39
10	5.000	10.00	10.00	41.05	38.31	40.00	0.000	25.17
11	5.000	10.00	10.00	41.54	38.51	40.00	0.000	25.18
12	5.000	10.00	10.00	41.54	38.53	40.00	0.000	25.18


Maximum acceleration drivers seat [m/s2]
10
9
8
7
6
0
5
10
15
Design cycle
Maximum constraint violation
0.6
0.4
0.2
0
-0.2
0
5
10
15
Design cycle
Constraint
Approximation error
0.4
0.3
0.2
0.1
0
0
5
10
15
Design cycle
Design variable
50
40
30
20
10
0
0
5
10
15
Design cycle

Landing gear mechanism

==> relative move limit
==> automatic move limit strategy

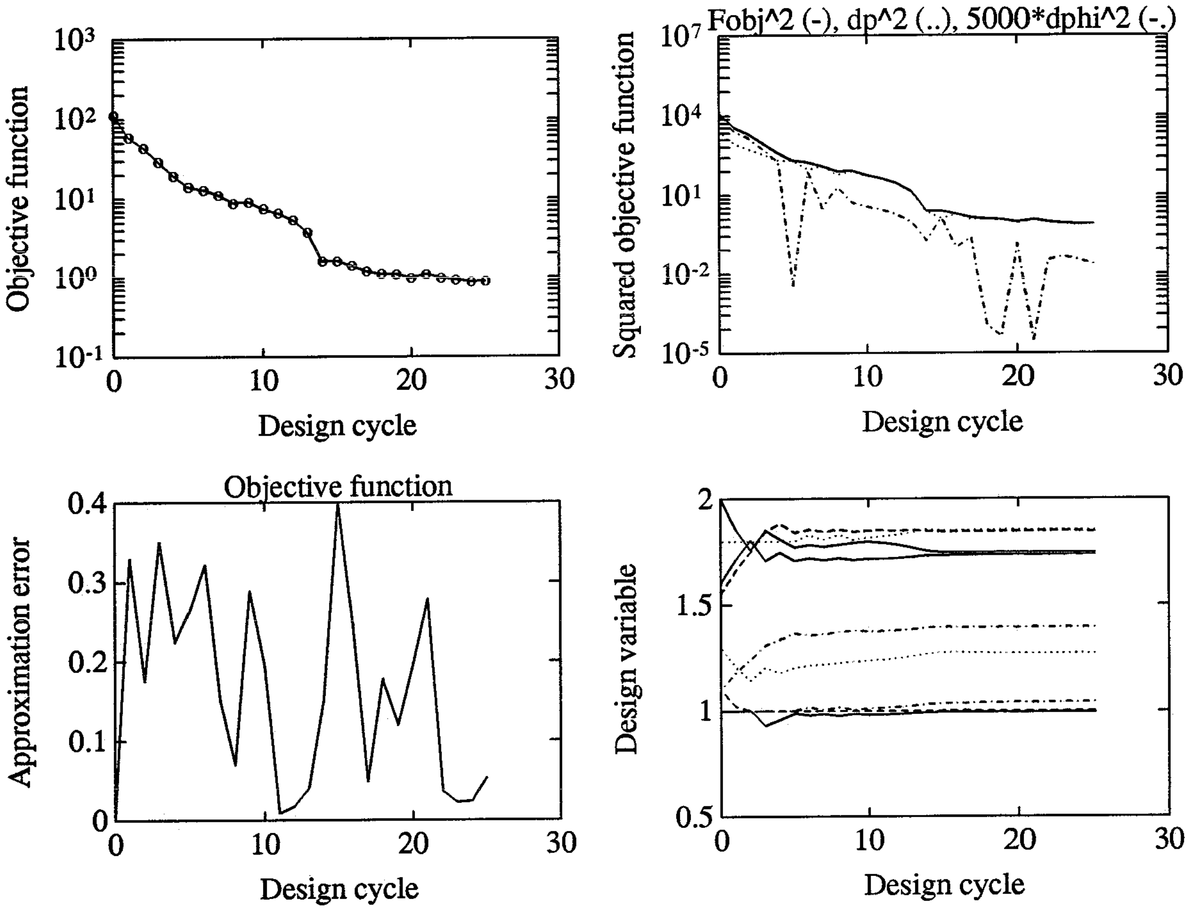
Design variables:
```
    y1=x(1)*100
    z1=x(2)*100
    y2=x(3)*100
    z2=x(4)*100
    y3=x(5)*100
    z3=x(6)*100
    y4=x(7)*100
    z4=x(8)*100
    theta=x(9)
```

L	ML [max %]	MLact	F	Fben
1	.000E+00	0	.112E+03	.000E+00
2	-.712E-01	1	.589E+02	.396E+02
3	.712E-01	1	.437E+02	.360E+02
4	.712E-01	1	.293E+02	.190E+02
5	-.400E-01	1	.195E+02	.151E+02
6	.400E-01	1	.141E+02	.103E+02
7	-.169E-01	1	.129E+02	.876E+01
8	.127E-01	1	.112E+02	.949E+01
9	.127E-01	1	.884E+01	.823E+01
10	-.127E-01	1	.909E+01	.646E+01
11	.713E-02	1	.755E+01	.608E+01
12	.713E-02	1	.663E+01	.669E+01
13	.713E-02	1	.541E+01	.551E+01
14	.713E-02	1	.377E+01	.392E+01
15	.950E-02	1	.160E+01	.184E+01
16	-.401E-02	1	.161E+01	.967E+00
17	-.127E-02	1	.140E+01	.106E+01
18	.127E-02	1	.118E+01	.113E+01
19	.169E-02	1	.110E+01	.908E+00
20	.169E-02	1	.109E+01	.960E+00
21	.169E-02	1	.976E+00	.785E+00
22	.169E-02	1	.109E+01	.787E+00
23	.846E-03	1	.974E+00	.938E+00
24	.714E-03	1	.925E+00	.945E+00
25	.714E-03	1	.880E+00	.901E+00
26	.752E-03	1	.890E+00	.843E+00

L	X(1)	X(2)	X(3)	X(4)	X(5)	X(6)	X(7)	X(8)	X(9)
1	1.000	1.100	1.300	1.100	1.600	1.550	1.800	1.000	-2.000
2	1.000	1.022	1.207	1.178	1.714	1.660	1.800	1.000	-1.858
3	1.000	1.000	1.143	1.241	1.805	1.749	1.800	1.000	-1.758
4	0.9288	1.000	1.204	1.307	1.709	1.842	1.800	1.000	-1.852
5	0.9567	1.000	1.177	1.337	1.748	1.884	1.800	1.000	-1.811
6	0.9854	1.000	1.203	1.367	1.708	1.841	1.800	1.000	-1.770
7	0.9761	1.000	1.212	1.357	1.720	1.855	1.830	1.017	-1.782
8	0.9830	1.000	1.218	1.365	1.711	1.845	1.807	1.004	-1.773
9	0.9760	1.000	1.225	1.372	1.720	1.855	1.830	1.017	-1.782
10	0.9830	1.000	1.232	1.379	1.711	1.845	1.807	1.004	-1.792
11	0.9795	1.000	1.236	1.374	1.716	1.850	1.818	1.010	-1.799
12	0.9812	1.000	1.241	1.376	1.718	1.852	1.824	1.013	-1.792
13	0.9836	1.000	1.248	1.380	1.721	1.849	1.832	1.018	-1.782
14	0.9867	1.000	1.257	1.385	1.725	1.853	1.842	1.023	-1.770
15	0.9908	1.000	1.269	1.392	1.731	1.847	1.856	1.031	-1.753

16	0.9926	1.000	1.274	1.394	1.733	1.845	1.850	1.034	-1.746
17	0.9920	1.001	1.276	1.394	1.734	1.846	1.851	1.035	-1.748
18	0.9923	1.000	1.274	1.394	1.734	1.846	1.852	1.036	-1.747
19	0.9919	1.000	1.272	1.393	1.735	1.846	1.853	1.038	-1.748
20	0.9923	1.000	1.270	1.394	1.736	1.847	1.854	1.039	-1.747
21	0.9921	1.000	1.272	1.394	1.737	1.847	1.854	1.041	-1.748
22	0.9923	1.000	1.270	1.394	1.736	1.848	1.854	1.039	-1.747
23	0.9922	1.000	1.271	1.394	1.737	1.848	1.854	1.040	-1.747
24	0.9923	1.000	1.271	1.394	1.737	1.849	1.854	1.040	-1.747
25	0.9923	1.000	1.272	1.394	1.737	1.848	1.854	1.041	-1.747
26	0.9924	1.000	1.271	1.394	1.738	1.847	1.854	1.041	-1.746

N.B. Scaling of the design variables could have been better.



Objective function
Design cycle
Fobj^2 (-), dp^2 (..), 5000*dphi^2 (-.)
Squared objective function
Design cycle
Objective function
Approximation error
Design cycle
Design variable
Design cycle

Stress constrained fourbar mechanism

==> relative movelimit
==> automatic move limit strategy
==> additional design variable for infeasibility
==> 3 two-sided constraints (-sa<s<sa) --> 6 one-sided constraints

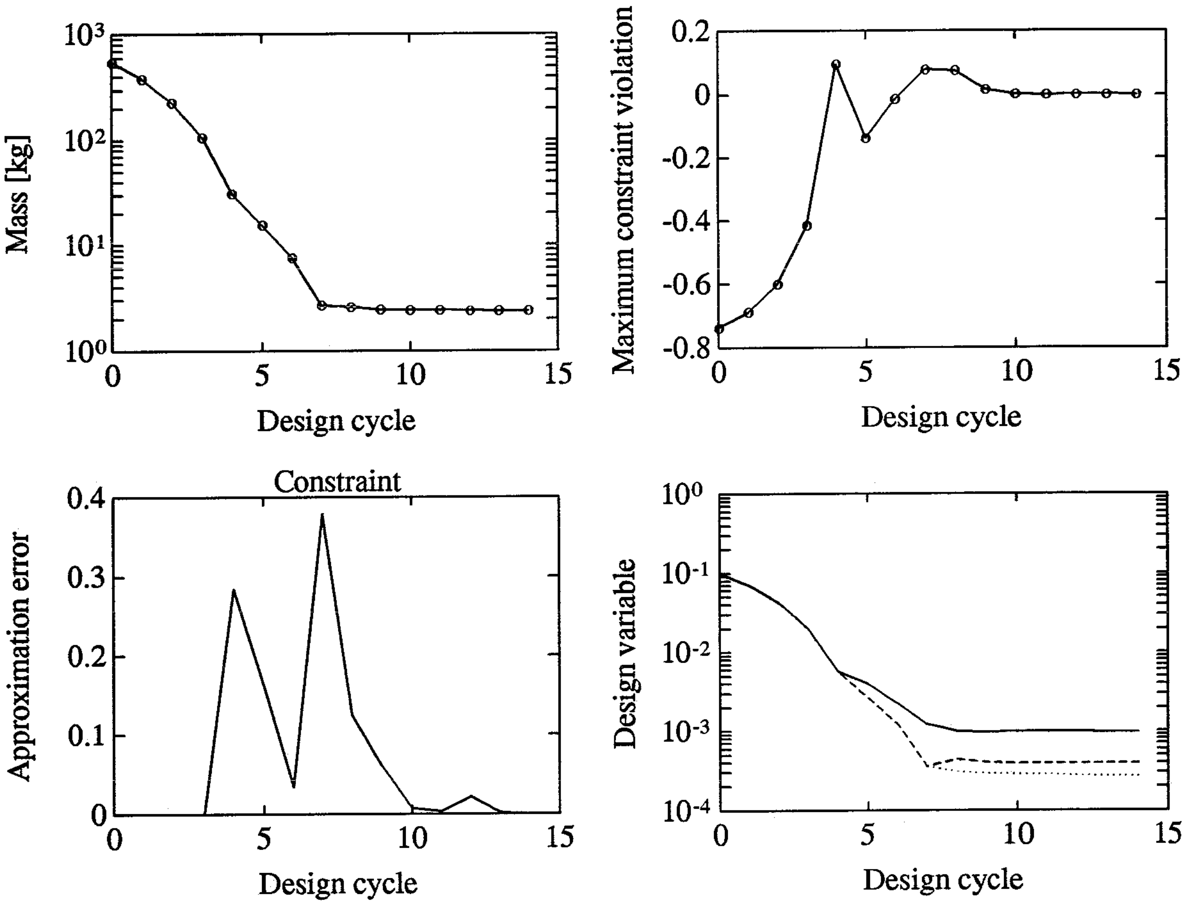
Design variables: no scaling

Optimum:
1. The approximated constraint 154 is active at its upper bound
2. The approximated constraint 1774 is active at its upper bound
3. The approximated constraint 2785 is active at its upper bound

1 = stress constraint acting at joint 1
2 = stress constraint at the node lying halfway between joint 2 and joint 3 of bar 2
3 = stress constraint at the node lying halfway between joint 3 and joint 4 of bar 3

L	ML [max %]	MLact	F	Fben	no	max(G) [%]	no	max(Gerr) [%]
1	.000E+00	0	.546E+03	.000E+00	154	-.739E+00	0	.000E+00
2	.300E+00	1	.382E+03	.382E+03	154	-.689E+00	0	.000E+00
3	.400E+00	1	.229E+03	.229E+03	154	-.600E+00	0	.000E+00
4	.533E+00	1	.107E+03	.107E+03	154	-.415E+00	0	.000E+00
5	.711E+00	1	.309E+02	.309E+02	154	.960E-01	148	.284E+00
6	.533E+00	1	.156E+02	.156E+02	154	-.137E+00	152	.163E+00
7	.533E+00	1	.756E+01	.756E+01	154	-.145E-01	150	.336E-01
8	.711E+00	1	.267E+01	.267E+01	1776	.788E-01	154	.379E+00
9	.533E+00	0	.258E+01	.258E+01	154	.758E-01	1580	.124E+00
10	.400E+00	0	.243E+01	.243E+01	152	.167E-01	1788	.604E-01
11	.400E+00	0	.241E+01	.241E+01	1774	.162E-02	2801	.735E-02
12	.400E+00	0	.240E+01	.240E+01	1774	-.194E-03	2801	.268E-02
13	.400E+00	0	.238E+01	.238E+01	1774	.123E-02	2803	.217E-01
14	.400E+00	0	.236E+01	.236E+01	154	.115E-02	156	.181E-02
15	.400E+00	0	.236E+01	.236E+01	154	.324E-05	2987	.875E-05

L	X(1)	X(2)	X(3)	X(4)
1	.1000E+00	.1000E+00	.1000E+00	.0000E+00
2	.7000E-01	.7000E-01	.7000E-01	.0000E+00
3	.4200E-01	.4200E-01	.4200E-01	.0000E+00
4	.1960E-01	.1960E-01	.1960E-01	.0000E+00
5	.5662E-02	.5662E-02	.5662E-02	.0000E+00
6	.3999E-02	.2642E-02	.2642E-02	.0000E+00
7	.2216E-02	.1233E-02	.1233E-02	.0000E+00
8	.1218E-02	.3562E-03	.3562E-03	.0000E+00
9	.9889E-03	.4395E-03	.3060E-03	.0000E+00
10	.9731E-03	.3980E-03	.2899E-03	.0000E+00
11	.9768E-03	.3921E-03	.2854E-03	.0000E+00
12	.9748E-03	.3924E-03	.2822E-03	.0000E+00
13	.9814E-03	.3915E-03	.2722E-03	.0000E+00
14	.9644E-03	.3921E-03	.2692E-03	.0000E+00
15	.9650E-03	.3921E-03	.2691E-03	.0000E+00



Mass [kg]
10³
10²
10¹
10⁰
0
5
10
15
Design cycle
Maximum constraint violation
0.2
0
-0.2
-0.4
-0.6
-0.8
0
5
10
15
Design cycle
Constraint
Approximation error
0.4
0.3
0.2
0.1
0
0
5
10
15
Design cycle
Design variable
10⁰
10⁻¹
10⁻²
10⁻³
10⁻⁴
0
5
10
15
Design cycle

Movelimit strategy

```
compute x* of LPP
compute maxfout, maxg, feas, mvlimact, boundact
if {(maxfout>0.4) or (maxg>0.1 and feas=yes)}
   f(k):=f(k)*3/4 with k=1,...,n
   do not accept optimum, restart cycle
else
   accept optimum as starting design next cycle
   for i:=1:1:n
      if (dxinew*dxiold<0 and
         [abs{(Fnew-Fold)/Fold}<0.02 or dFnew*dFold<0] and
         feas=yes) then
         f(i):=f(i)/2
      elseif {F*>F0 and feas=yes} then
         f(i):=f(i)*3/4
      else
         if { mvlimact(i)=yes and boundact(i)=no} then
            if {maxfout<0.10 and (maxg<0.05 or feas=no)} then
               f(i):=f(i)*4/3
            end
            if {maxfout>0.25 or (maxg>0.075 and feas=yes)} then
               f(i):=f(i)*3/4
            end
         else
            if [{maxfout>0.25 or (maxg>0.075 and feas=yes)} and
               boundact(i)=no] then
               f(i):=f(i)*3/4
            end
         end
      end
   end
   if [abs{(Fnew-Fold)/Fold}<0.005 and maxg<0.005] then
      convergence
   end
end
```

abbreviations:

n : number of design variables excluding the additional max-value and feasibility design variable
x0 : starting design of the cycle in concern
x* : solution of the linear programming problem within the search subregion of the cycle in concern
x0i : the i th element of the vector x0
x*i : the i th element of the vector x*
LPP : linear programming problem
maxfout : max(e0, e1, e2 , ..., em) , see equation (7) and (8)
maxg : maximum constraint value. A number larger than zero means a constraint violation
feas=yes : there is a feasible solution of the LPP, so the additional feasibility design variable is zero
feas=no : there is no feasible solution of the LPP. The additional feasibility design variable is larger than zero to create a feasible LPP.
aiu : upper movelimit of the i th design variable
ail : lower movelimit of the i th design variable
xiu : upper bound of the i th design variable
xil : lower bound of the i th design variable
mvlimact(i)=yes : xi lies on aiu or ail
mvlimact(i)=no : xi lies within aiu and ail
boundact(i)=yes : xi lies on xiu or xil for i=1,...,n

```
boundact(i)=no : xi lies within xiu and xil
f(i) : relative move limit factor
       xi - xi * f(i) <= xi <= xi + xi * f(i)
dxnewi : dxnewi = (x*i - x0i) for the present design cycle
dxoldi : dxoldi = (x*i - x0i) for the previous design cycle
dxnewi * dxoldi < 0 change of sign in xi
F* : objective function value at x*
F0 : objective function value at x0
```