Optimisation methods in CFD

Differential evolution

Ondřej Suchomel, FME at CTU
4th year of study, field of study:
Engineering Mechanics and Mechatronics

20th of April 2006

subject:	**Project II., Computer fluid mechanics**
goal:	**Use of optimisation method in aerodynamics**
description:	**Optimisation of shape and position of back wing of a model of race car**
optimisation method:	**Differential evolution**
software:	**Gambit, Fluent, DevC++**

Optimisation means finding of global extreme of function `f` by `n` variables:

$$f = f(x_k), k = 1,\dots.D$$

Mathematical condition of extreme is zero gradient function `f` :

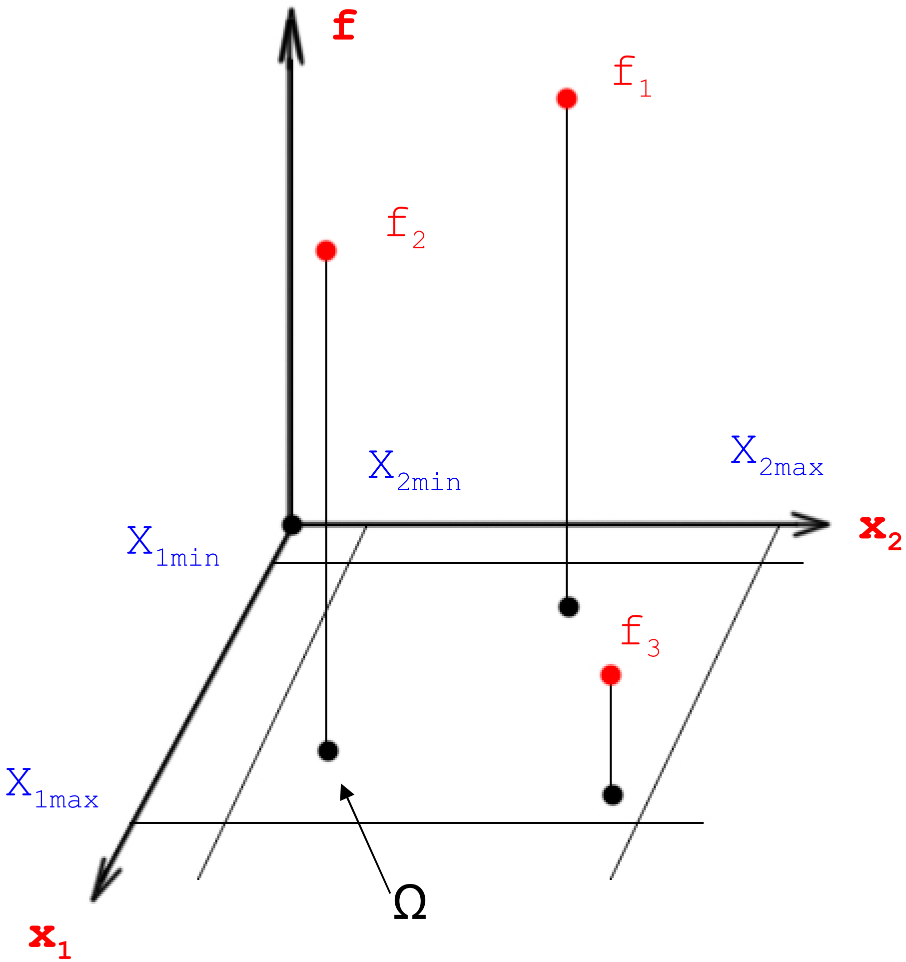
$$\frac{\partial f}{\partial x_k} = 0$$

Analytical expression of function `f` is in majority of engineering problems impossible.

→ Is necessary to utilize any instrument, which simplifies finding the extreme of coast function `f`.

Note. If analytical expresion of function `f` is possible, can be very complicated. Therefore execution of partial derivations is very difficult or so to say unviable.

Is assigned optimisation problem, where `f` is function of 2 variables:



- $D = 2$
- defined on zone Ω
- $x_1 \in <x_{1min}, x_{1max}>$
- $x_2 \in <x_{2min}, x_{2max}>$

- f_1, f_2, f_3 are values of coast function, those extreme should be found

Note. In practice engineering optimisation problem use to be function of more than 2 variables, i.e. in rank of unitsalso decades. As well may not be defined on simple (in 2D rectangular) zone. Zone can be as well function, i.e. circle, sphere etc. Then we talk about constrained extreme of function. Some optimisation parameters can take discrete values.

optimisation

Regarding definition of optimisation problem can be optimisation parameters x_i diverse physical and non-physical magnitudes:

- **geometrical** - length, angle
- **kinematic** - velocity, acceleration
- **technological** - feed, cutting velocity
- **economical** - price, operating costs
- **thermodynamic** - temperature, pressure

Coast function can be as well whatever, e.g.

- total weight, volume
- max. velocity, laptime
- machining time
- lifetime
- aerodynamic resistance, lift

Note. In the most of optimisation problems isn´t optimised only one coast function, but combination of more functions. So called pareto set of best solutions is generated. In 2D curve, surface in 3D.

Correct information about shape of coast function f in zone Ω mostly doesn´t exist. Therefore is advantageous to have a device, which helps to find the extreme.

General procedure

1. Determination of a few first test points from zone Ω
2. Testing of coast function f for chosen test points
3. Determination of new test points using optimisation method

Then se loop 2. – 3. is repeating until extreme of coast function f is found

Note. Term „finding of extreme“ is not exact. Location and value of extreme function, which gives optimisation method, is only certain approximation to real extreme. Where real extreme is, we can proclaim only with analytical finding of extreme of mathematically defined function. Exactness of localization of extreme is mostly sufficient.

optimisation method

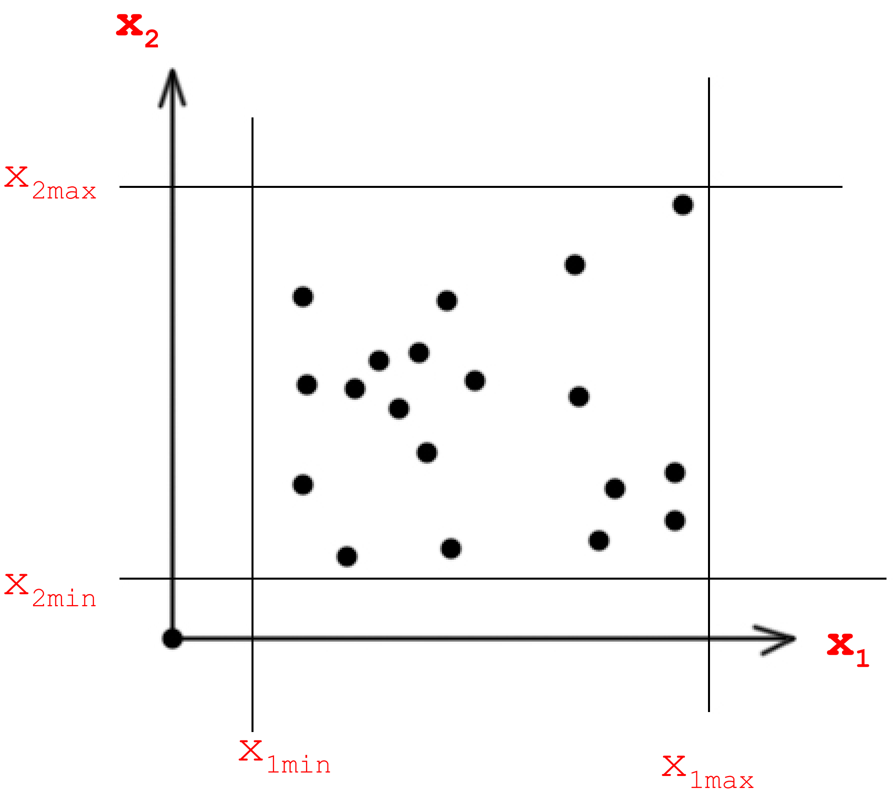
Exists large number of optimisation methods, e.g.

- stochastic
- geometrical – **Simplex method**
- evolutional – **Differential evolution**

Note. Count of type of optimisation methods and type of methods themselves is countless. Regarding modified methods the same. Every programmer can modify every method to suit his imagination or solved problem.

stochastic method

Simplest stochastic method utilize for finding of extreme random. Method generates pairs of numbers from zone Ω, in which tests their coast function f. Then maximum / minimum is proclaimed as a result.



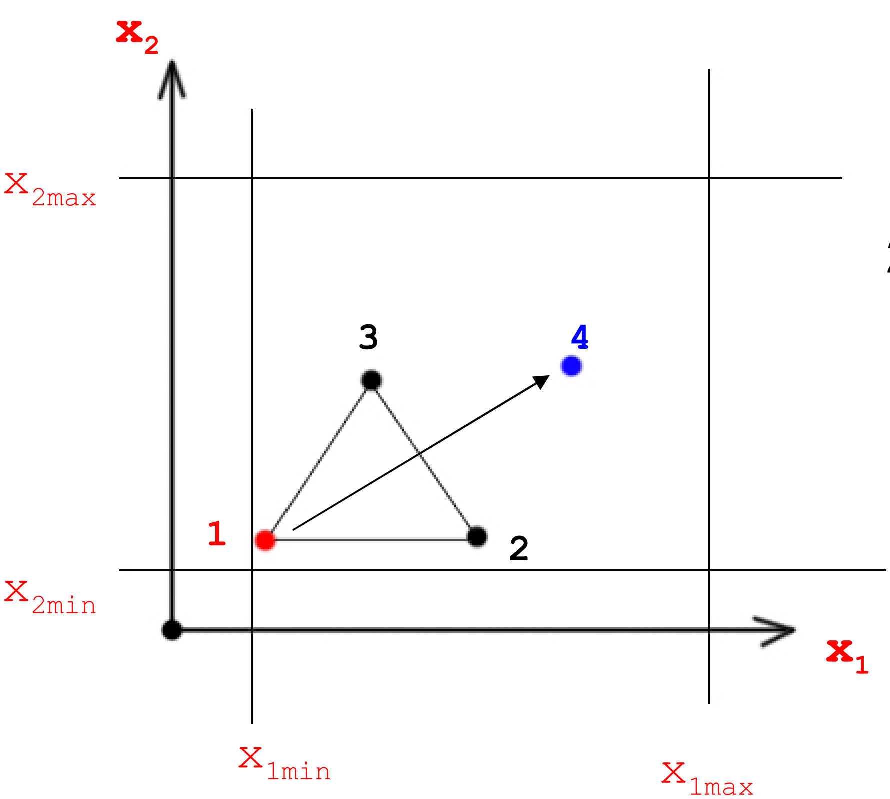
$$x_1^i = rand(x_{1\min}, x_{1\max})$$

$$x_2^i = rand(x_{2\min}, x_{2\max})$$

Note. Other stochastic method is e.g. Monte Carlo.

simplex method

Simplex method utilizes for determination of new test points simple geometrical shapes – simplexes. In 2D it is equilateral triangle (in 3D regular tetrahedron). At determination of new test points algorithm is controlled with 2 rules:



1. Point with the worst value of `f(1)`, is let out and supplanted with new vertex (**4**), so that new equilateral triangle is formed.

2. It is not allowed to return in actually let out point.

Disadvantage: Similar as others geometrical methods stops in local extreme, which needs not to be global extreme.

Note. Simplex method takes not into account direction in which is moving. This uses e.g. method of Hook and Jeaves, which at repeated movement in same direction increases step and aims to accelerate finding of extreme.

evolutional algorithms

Evolutional methods are very young optimisation algorithms. First evolution method is genetic annealing (1994, Price, Storn, USA).

Evolutional methods implement natural laws in optimisation. Basic rule is:
“In new generation can not be any weak specimen.“

Terminology

- **specimen** - vector of D+1 components
- **population** - set of NP specimens
- **generation** – j^{th} population
- **evolution** - sequence of G generations

$$J^{i,j} = [x_{k=1}^{i,j}, x_{k=2}^{i,j}, f^{i,j}]$$

$$i = 1,.....NP$$

$$j = 0,.....G$$

$$k = 1,.....D$$

Note. Evolutional algorihtms are relatively new methods of optimisation. This concerns with coming of computers, whose utilization is in these methods nearly essential.

differential evolution

Differential evolution is one of evolutional algorithms. Every specimen from j^{th} generation produces just 1 child, which is member of j^{th+1} generation.

Properties of differential evolution

- very simple to programme it, the same for more dimensions
- very reliable to find global extreme of coast function `F`(in contrast with geometrical optimisation methods)
- if function has more global extremes, differential evolution finds them
- can with number formats `float`, `integer`, with binary numbers and combinations
- solution of optimisation problem is one (or more) best solution

Note. Differential evolution is possible to use for analytically defined functions to find of their global extremes. If e.g. execution of partial derivations for analytical determination of extreme is difficult, then use of evolutional algorithm is suitable and very fast.

Proper algorithm

1. **Definition of parameters** - reproduction loop is driven with them
2. **Creation of population** - 1st generation
3. **Reproduction loop** - grade between specimens from actual generation creation of new specimens
4. **Testing coast function `f` for test specimen**
5. **Determination of new population**

   loop 3. – 5. is repeating until stop condition is satisfied. Best specimen is proclaimed as a result of optimisation problem.

6. **Interpretation of results of evolution**

Note. This scheme of algorithm is essence of other evolutional algorithm.

1. Definition of parameters of differential evolution

- **NP** `NP > 3` - count of specimens, size of population
- **F** `F ∈ <0,2>` - mutational constant
- **CR** `CR ∈ <0,1>` - threshold of grade
- **G** - count of generations

2. Creation of population – random determination of first generation

$$x_k^{i,0} = x_{k\,\min} + rand(x_{k\,\min}, x_{k\,\max}) \quad \texttt{for} \quad i = 1,\ldots..NP$$
$$k = 1,\ldots..D$$

Note. Definition of suitable parameters depends on experience of programmer. If chooses `F = 0` or `CR = 1`, then every next generation is the exact copy of 1st generation. More see formula for test vector on the next site. Differential evolution is possible to utilize also for definition of parameters, then is called metadifferential evolution.

3. Reproduction loop

For every specimen (parent) is generated test specimen `zk`.
At first are chosen 3 different specimens of same generation. Difference of first two specimen multiplied with constant `F` is added to third specimen. Results is so-called noise vector υ .

$$\upsilon_k^i = x_k^{r3,G} + F(x_k^{r1,G} - x_k^{r2,G}) \qquad \texttt{for} \quad i = 1,\ldots..NP$$

Then is for every dimension generated random number <0,1>. If number is smaller than threshold of grade `CR`, test specimen for actual dimension is assigned parameter from noise vector. If number is larger, then parameter of parent.

$$if(random < CR) \longrightarrow zk_k^i = \upsilon_k^i \qquad \texttt{for} \quad i = 1,\ldots..NP$$

$$if(random \geq CR) \longrightarrow zk_k^i = J_k^i \qquad k = 1,\ldots..D$$

Note. From formula for noise vector is evident signification of parameter `F`. Larger parameter `F`, the „more different“ is parameter of noise vector than parameter of specimen Nr. 3. Parameter `CR` (threshold of grade) determines probability, with which test vector gets parameter of noise vector. Reason of term “differential evolution” is necessary search likewise in first formula. Term is not derivated from differential mathematics, but from modest difference of vectors.

4. Testing coast function `f` for test specimen

For every test specimen is tested coast function `f`. Instrument for testing of coast function mostly is not part of program of evolutional algorithm. e.g. Fluent, Excel and others.

5. Determination of new population

Comparing values of coast functions of parents and test vectors is determined new generation of specimens. If coast function `f` of test vector `zk` is better than parent, progresses in new generation test vector. If not, progresses without change parent. Thereby is guarenteed, that in new generation doesn´t advance worse specimen than parent.

$$if\,(f\,(zk_k^i) > f\,(J^{i,j})) \longrightarrow J^{i,j+1} = zk^i \qquad \texttt{for } i = 1,\ldots..NP$$

$$if\,(f\,(zk_k^i) < f\,(J^{i,j})) \longrightarrow J^{i,j+1} = J^{i,j}$$

Note

During reproductional loop often happens that at least one parameter of newly generated test specimen zk is out of zone Ω. Solution:

I. Stop of specimen on boarder – probability of finding local extreme

II. Random generating of overstepped parameter, same as at at creation of the very first population. This is a standard type of differential evolution.

Note. Solution of problem, when specimen at least with one of his parameter leaves zone, can be arbitrary. e.g. return to first third from overstepped boarder. Can happen a situation, when becomes to downgrade of diversity of population. This can influence correct location of extreme. With solution Nr. II is this danger lower.

concrete optimisation problem

Section	**Aerodynamics, use of CFD**
Coast function	**Lift**
Method	**Differential evolution**
Description	**Finding of at least suitable position and shape of back wing of model of race car**
Preprocessor	**Gambit**
CFD solver	**Fluent**
Programme	**DevC++**

Note. Goal of this work was testing of utilization of evolutional optimisation algorithm in aerodynamics with use of CFD software. Thus answer the question, whether differential evolution is suitable with use in CFD computations.

answer : **Which configuration is aerodynamically less advantageous?**

Which generates higher lift?

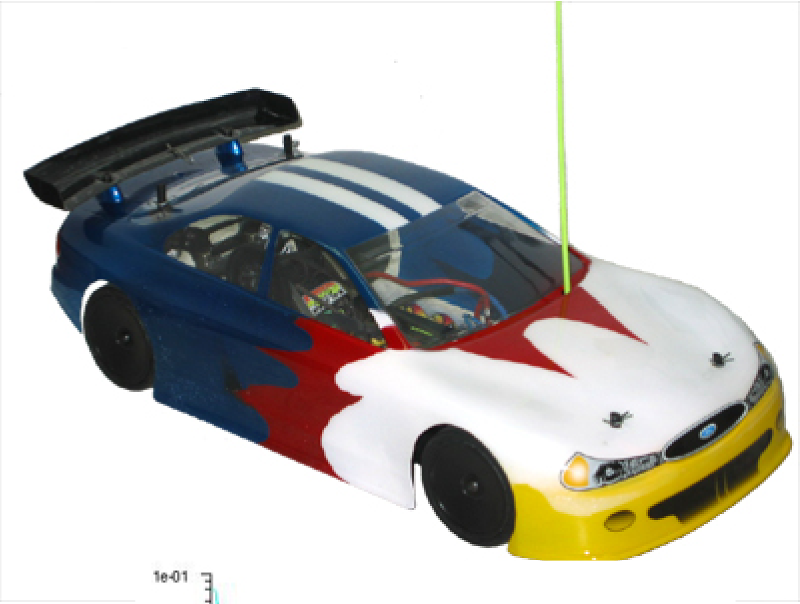
Lancer WRC

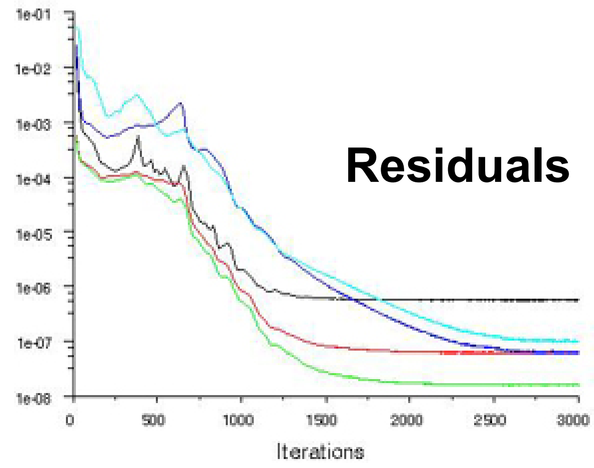
Impreza WRC

Note. Investigated object was back wing and shape and position on the car.

model case

Car:	**Ford Mondeo**
Scale:	**1/10**
Velocity	**10m/s**
Model of turbulence	**k – ε**
Number of iterations	**3000**
Scheme	**First order Upwind**

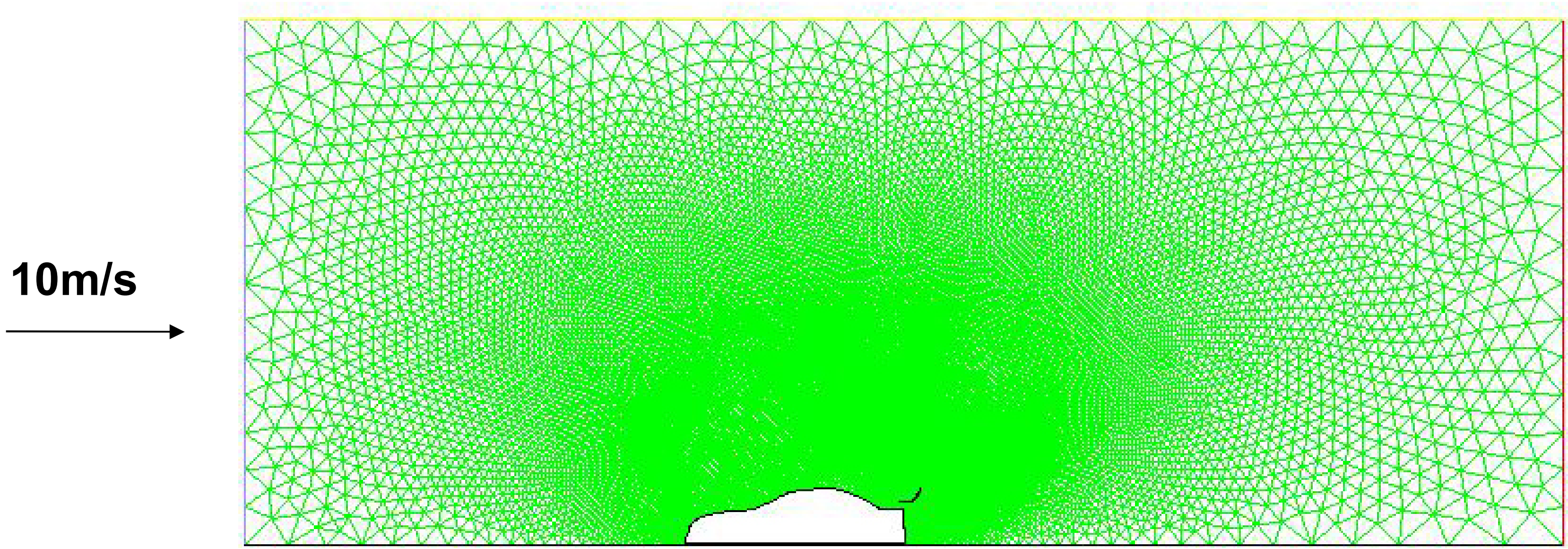




Residuals

Note. CFD simulation was executed for model of race car in scale 1/10. Velocity of flow was chosen so, that is getting near to velocity, which models in class E 1/10 TC achieve in fast corners of race tracks.

mesh



Grid

Jan 05, 2006
FLUENT 6.2 (2d, segregated, rngke)

Note. Mesh was generated automatically in program Gambit (text mode). Input data were coordinates of points of body, object of optimisation coordinates of points of back wing.

parameters

parameter k=1 **x-position of back wing,** x_k

x_k Є <365,416> [mm]

parameter k=2 **angle of displacement of vertical surface of back wing** α_k

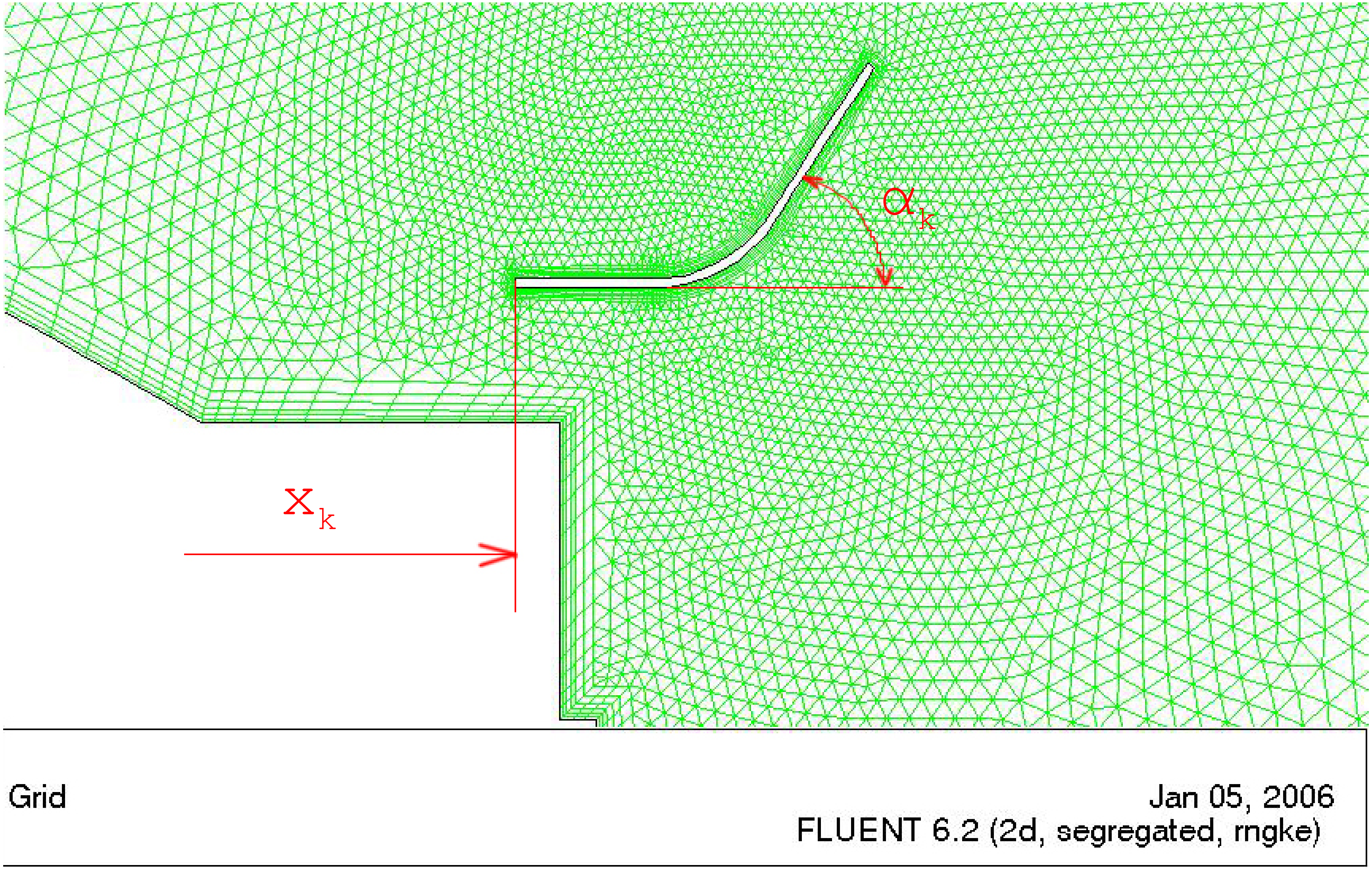
α_k Є <45,90> [°]

parameters of differential evolution:

NP = 4
F = 1,23
CR = 0,45
G = 4

Note. Parameters of differential evolution were chosen according recommendation from book [2]: Umělá inteligence (4). Time exigence of CFD computation allowed to realize 4 generations of differential evolution.

geometry



Note. Distance x_k was measured from front part of car, where was placed coordinate system.

Behaviour – best specimen

I. „stop on border“

G	X_k [mm]	α_k [°]	F_y* [N]
1	376.00	45.00	4.01
2	376.00	45.00	4.01
3	365.00	48.00	4.06
3	365.00	48.00	4.06
5	365.00	45.00	4.22

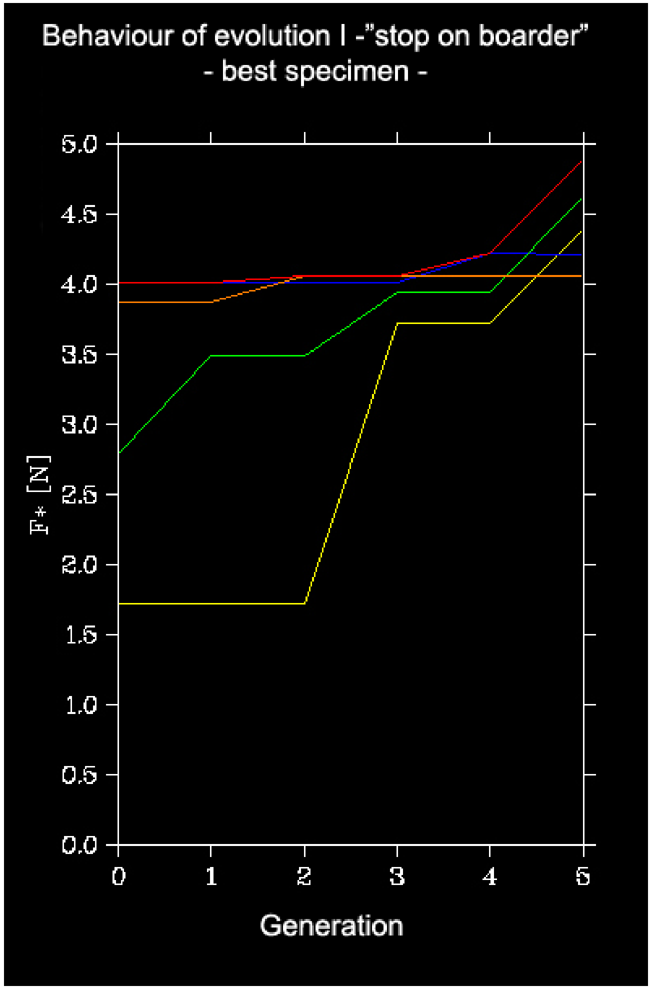
II. „standard“

G	X_k [mm]	α_k [°]	F_y* [N]
1	367.00	60.00	3.73
2	367.00	60.00	3.73
3	367.00	60.00	3.73
4	367.00	60.00	3.73
5	367.00	60.00	3.73

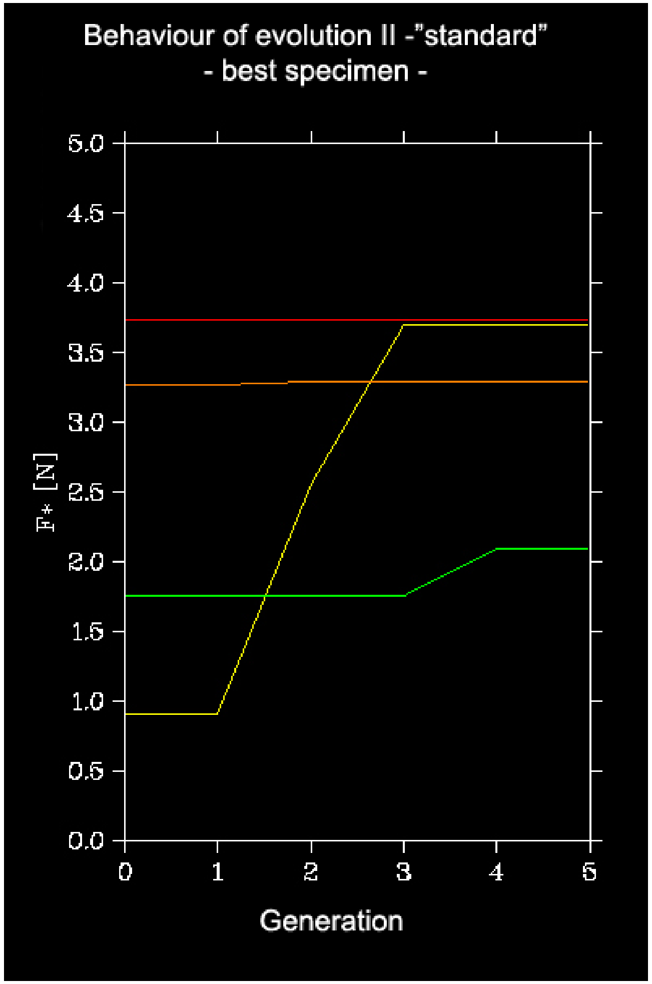
Note. Best specimen in case Nr.I. was known even in last generation, whilst in case Nr.II. already in first. This is work of random, that acts similar as in nature, notable role in differential evolution.

behaviour

I. „stop on border“

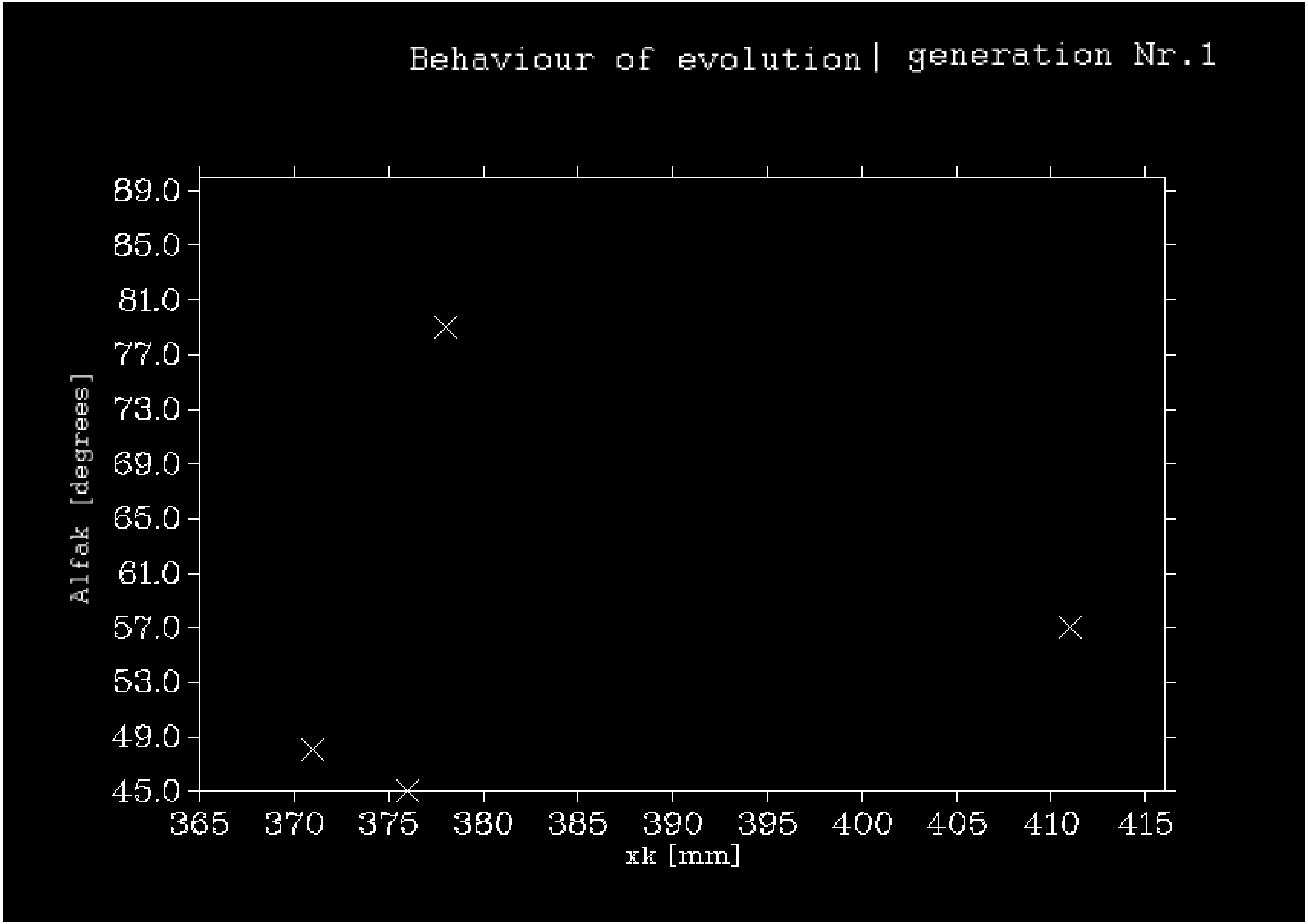


II. „standard“



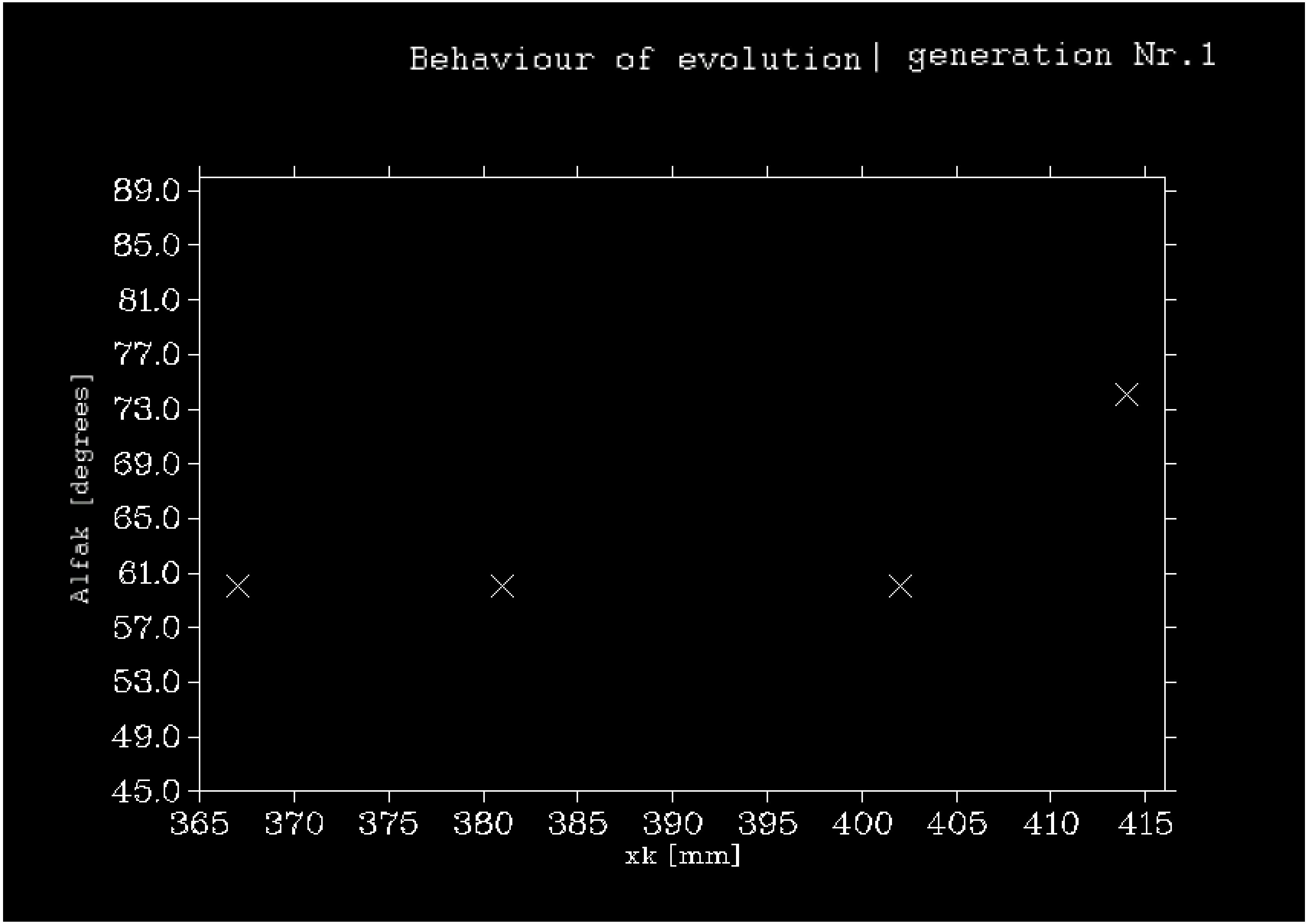
Note. Progress of every specimen is figured with lines of different color, best solution with red color. On picture is seen, that evolution Nr. I. had more interesting behaviour than evolution Nr. II., where best specimen was known in first generation.

I. „stop on border“



Note. Image presents progress evolution in individual generations. First frame is 1st generation, e.g. placement of specimens at generation of start population. Every next frame is figure of next generation. Placement of specimens in last generation responds danger of stop of specimen on a border by leaving of zone.

II. „standard“



Note. In last generation are specimens placed are in one row regarding to Alfa$_k$. This could show posibility of finding of optimised angle of displacement of vertical surface of back wing. Clear answer would provide computation of next generations.

In PDF doesn´t run animation of gif image. It is possible to watch on http://www.suchomelplasty.cz/academics/en

conclusion

Thanks limited possibilities of testing of coast function in CFD software and time-consuming computation was not able go all lengths. For simple mathematical functions (rotational paraboloid) is necessary hundreds or thousands testings of coast function.

Differential evolution is very heavy device for optimisation. However big percentage of computation of specimen is unsuccesfull, this means is not generated better specimen.

That´s why I would recommend for aerodynamic optimisation problems rather methods, which take in account as well results from previous computation.

My vision is e.g. Interlay of regress surface through tested points, computation of gradient and continuation in testing in direction of biggest gradient.

Note. Data obtained from bad specimens can be used for study of proper problem, that is with optimisation method solved.
But this doesn´t concern with proper optimisation.

answer

Although wasn´t possible lead optimisation till end, from obtained data results, that worse configuration of back wing has Mitsubishi Lancer WRC.

Note. Result of optimisation says, that the worst place to mount back wing is front part of baggage hold. This should Mitsubishi handicap. Practise, i.e. World Rally Championships, shows, that Mitsubishi Lancer is very competitive car. Answer is possible to find in vertical placement of surface creating thrust. That surface is on japanese car placed nearly in plane with roof. To the surface flows “quality“ air generating thrust. Vertical position of back wing wasn´t object of optimisation.

Author
Ondřej Suchomel
Faculty of Mechanical Engineering, Czech Technical University
ondra@suchomelplasty.cz

Head of project
Ing. Tomáš Hyhlík, Faculty of Mechanical Engineering at Czech Technical University, Department of Fluid Mechanics and Thermomechanics

Actual version of presentation
http://www.suchomelplasty.cz/academics

Literature

[1] Mařík a kolektiv: Umělá inteligence (4), Academia, Prague, 2003
[2] S. Kračmar, J. Vogel: Programovací jazyk C, Faculty of Mechanical Engineering of CTU, CTU, Prague, 2002

Other sources

http://wrc.auto.cz
http://www.mitsubishi-motors.com
http://www.swrt.com