

# Optimalizační metody v CFD

diferenciální evoluce

20.dubna 2006

Ondřej Suchomel, FS ČVUT  
4.ročník, obor IMM

předmět: **Projekt II., Počítačová mechanika tekutin**

cíl: **Použití optimalizační metody v aerodynamice**

popis: **Optimalizace tvaru a umístění přitlačného křídla modelu závodního vozu**

optimalizační metoda: **Diferenciální evoluce**

programy: **Gambit, Fluent, DevC++**

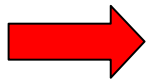
Optimalizace je nalezení globálního extrému funkce  $f$  o  $n$  proměnných:

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

Matematickou podmínkou extrému je nulový gradient funkce  $f$  :

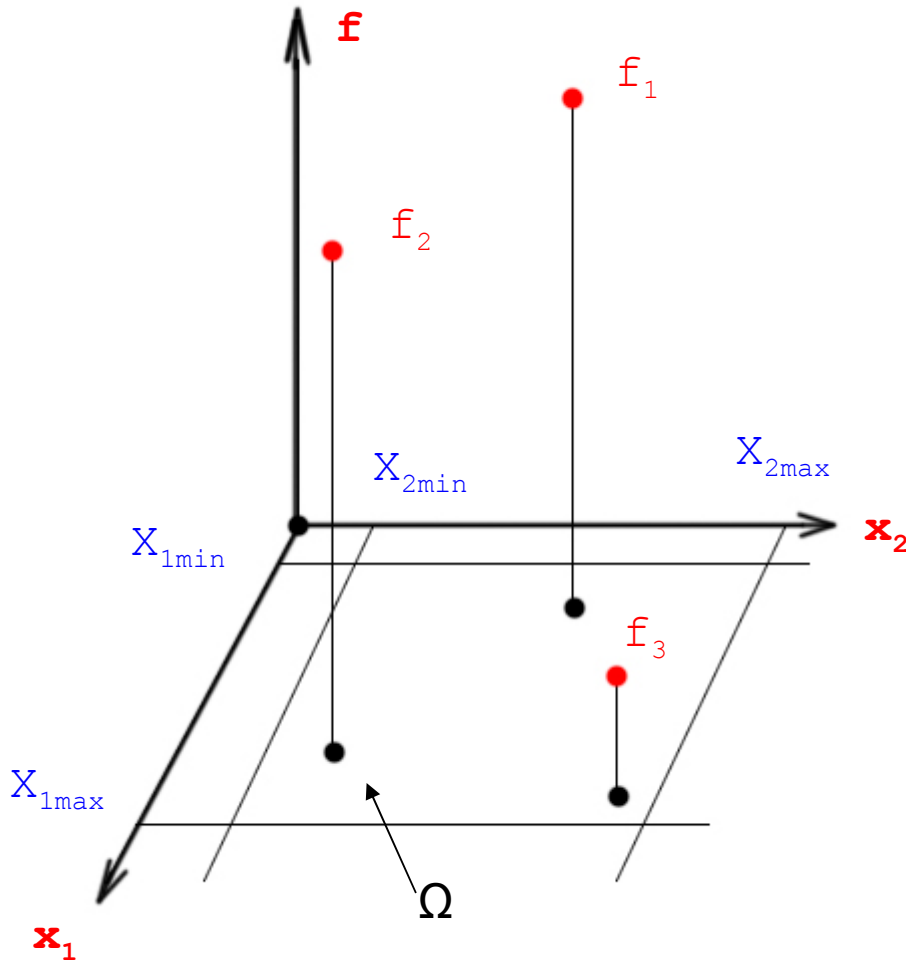
$$\frac{\partial f}{\partial x_k} = \mathbf{0}$$

Analytické vyjádření funkce  $f$  je ve většině inženýrských problémů nemožné.



Je nutné využít nějaký prostředek, který usnadní nalezení extrému cílové funkce  $f$ .

Je zadán optimalizační problém, kde  $f$  je funkce 2 proměnných:



- $D = 2$
- def. na oblasti  $\Omega$
- $x_1 \in \langle x_{1\min}, x_{1\max} \rangle$
- $x_2 \in \langle x_{2\min}, x_{2\max} \rangle$
- $f_1, f_2, f_3$  jsou hodnoty cílové funkce, jejíž extrém hledáme

**Pozn.** V praxi bývá inženýrský optimalizační problém funkcí více než dvou proměnných, např. v řádu jednotek i desítek. Rovněž nemusí být definován na jednoduché (ve 2D obdélníkové) oblasti. Oblast může být rovněž funkcí, např. kružnicí, koulí atd. Pak mluvíme o vázaném extrému funkce. Některé optimalizační parametry mohou nabývat diskrétních hodnot.

Podle sestavení optimalizačního problému mohou být optimalizačními parametry  $x_i$  různé fyzikální i nefyzikální veličiny:

- **geometrické** - délky, úhly
- **kinematické** - rychlosti, zrychlení
- **technologické** - posuv, řezná rychlost
- **ekonomické** - cena, náklady na provoz
- **termodynamické** - teplota, tlak

Cílovou funkcí může být rovněž cokoli, např.

- celková hmotnost, objem
- max. rychlost, čas na kolo
- strojní čas
- životnost
- aerodynamický odpor, vztlak

**Pozn.** V běžných optimalizačních problémech většinou není optimalizována jen jedna cílová funkce, ale kombinaci více funkcí. Vytváří se tzv. pareto množina, tj. křivka(2D), plocha 3D nejlepších řešení.

O tvaru cílové funkce  $f$  na oblasti  $\Omega$  většinou neexistují žádné korektní informace. Proto je výhodné mít nástroj, který pomůže extrém nalézt.

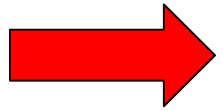
## obecný postup:

1. Určení prvních několika zkušebních bodů z oblasti  $\Omega$
2. Testování cílové funkce  $f$  pro vybrané zkušební body
3. Určení nových zkušebních bodů pomocí optimalizační metody

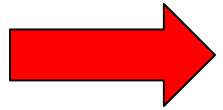
Poté se cyklus 2. – 3. opakuje až do nalezení extrému cílové funkce  $f$

**Pozn.** Termín „nalezení extrému“ není přesný. Místo a hodnota extrému funkce, kterou vyhodnotí optimalizační metoda, je jen určitým přiblížením ke skutečnému extrému. Kde leží skutečný extrém, lze prohlásit jen při analytickém nalezení extrému matematicky zadané funkce. Přesnost, se kterou metoda lokalizuje extrém, je však většinou dostatečná.

Optimalizačních metod existuje velké množství, např:

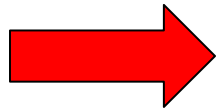


stochastické



geometrické

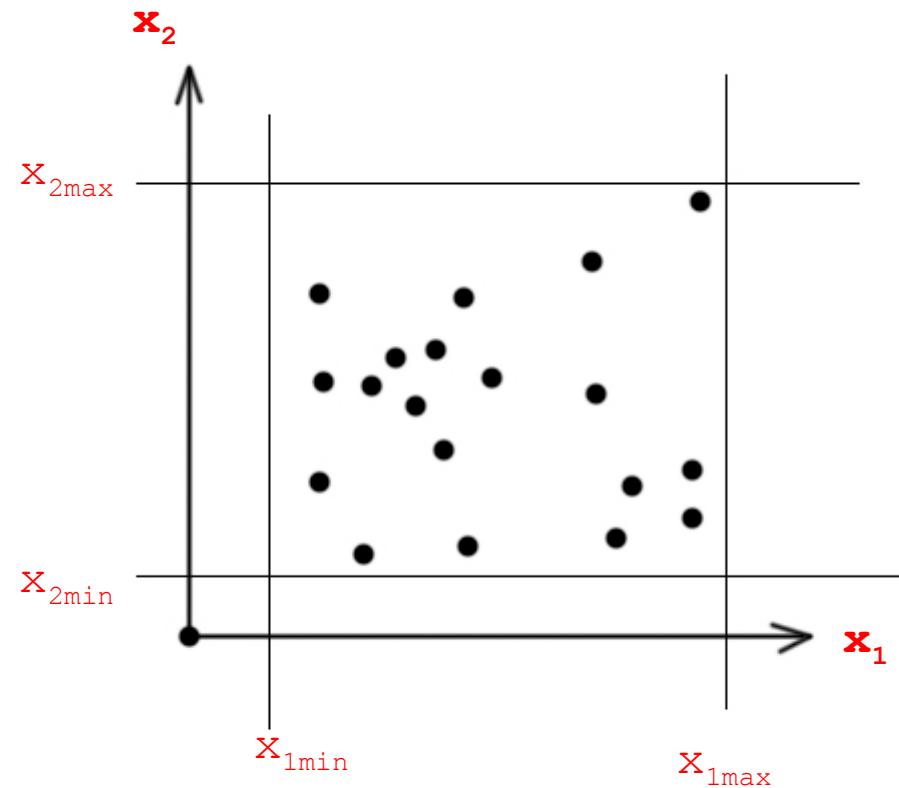
– **Simplexová**



evoluční

– **Diferenciální evoluce**

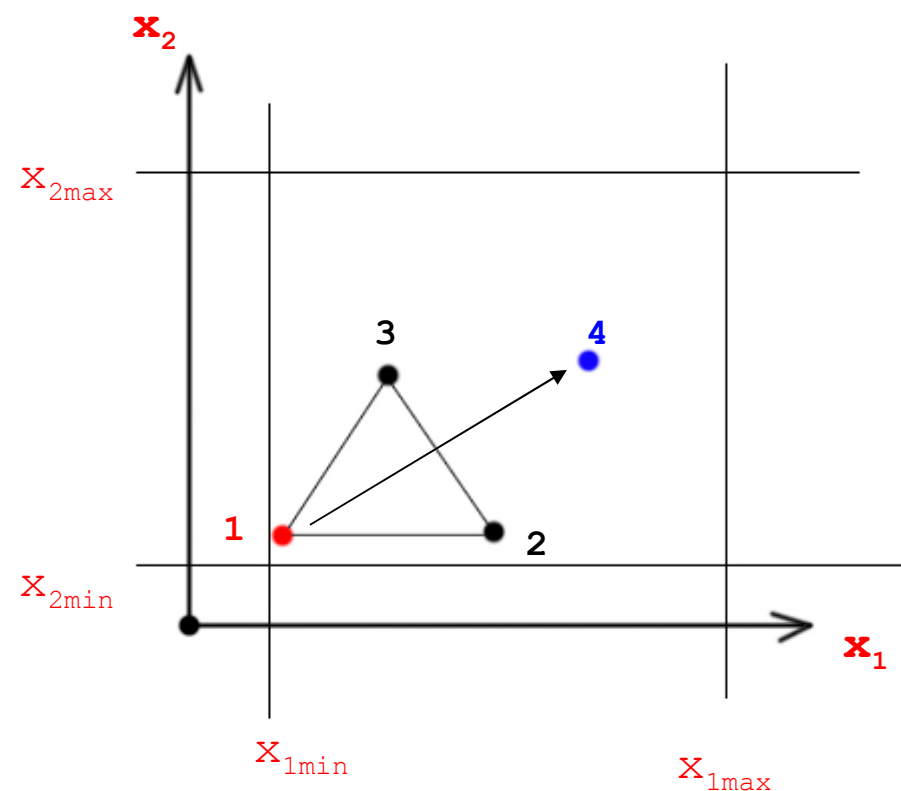
Nejjednodušší metoda stochastická využívá pro nalezení extrému náhody. Generuje dvojice čísel z oblasti  $\Omega$ , ve kterých se testuje cílová funkce  $f$ . Její maximum / minimum je poté prohlášeno za výsledek.



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

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

Metoda simplexová využívá pro určení nových testovacích bodů jednoduchých geometrických útvarů – simplexů. Ve 2D se jedná o rovnostranný trojúhelník ( v 3D o pravidelný čtyřstěn). Při určování nově testovaných bodů se řídí se dvěma pravidly:



1. Bod s nejhorší hodnotou  $f$  (**1**), se vypustí a nahradí se novým vrcholem, tak aby opět vznikl rovnostranný trojúhelník (**4**).
2. Není dovoleno se vracet do právě vypuštěného bodu.

**Nevýhoda:** Podobně jako ostatní geometrické metody se zastaví v lokálním extrému, který nemusí být extrémem globálním

**Pozn.** Simplexová metoda nezohledňuje výhodnost směru, ve kterém se pohybuje. Toho např. využívá metoda Hooka a Jeavese, která při opakovaném postupu v daném směru zvětšuje krok a tím se snaží o urychlení nalezení extrému.

Evoluční metody jsou velmi mladé optimalizační algoritmy. První metodou tohoto typu je genetické žíhání (1994, Price, Storn, USA).

Evoluční metody implementují do optimalizace zákony přírody. Základním pravidlem je: „Do nové generace se nedostane žádný slabý jedinec.“

## Terminologie

- **jedinec** - vektor o  $D+1$  složkách
  - **populace** - množina  $NP$  jedinců
  - **generace** -  $j$ -tá populace
  - **evoluce** - posloupnost  $G$  generací
- $$J^{i,j} = [x_{k=1}^{i,j}, x_{k=2}^{i,j}, f^{i,j}]$$
- $$i = 1, \dots, NP$$
- $$j = 0, \dots, G$$
- $$k = 1, \dots, D$$

Diferenciální evoluce je jedním z evolučních algoritmů. Každý jedinec z  $j$ -té generace vytváří právě jednoho potomka, který je členem  $j+1$  generace.

## Vlastnosti Diferenciální evoluce

- velmi jednoduchá na naprogramování, velmi rychlá, i pro více rozměrů
- velmi spolehlivě hledá globální extrém cílové funkce  $f$  (na rozdíl od geometrických optimalizačních metod)
- pokud má funkce více globálních extrémů, nalezne je
- může pracovat i s čísly typu `float`, `integer`, s binárními čísly a kombinacemi
- řešením je jedno (popř. více) nejlepší řešení

**Pozn.** Diferenciální evoluci lze použít i pro analyticky přesně definované funkce a hledání jejich globálních extrémů. Pokud je např. provedení parciálních derivací pro analytické určení extrému složité, pak je použití evolučního algoritmu vhodné a velmi rychlé.

## Vlastní algoritmus

1. **Stanovení parametrů** - řídí se jimi reprodukční cyklus
2. **Tvorba populace** - 1.generace
3. **Reprodukční cyklus** - křížení mezi jedinci aktuální generace, tvorba nových jedinců
4. **Testování cílové funkce  $f$  pro zkušební jedince**
5. **Určení nové populace**

Cyklus 3. – 5. se opakuje do splnění ukončovacího podmínky. Nejlepší jedinec je vyhlášen řešením optimalizační úlohy

6. **Vyhodnocení evoluce**

## 1. Stanovení parametrů Diferenciální evoluce

- **NP**     $NP > 3$     - počet jedinců, velikost populace
- **F**     $F \in \langle 0, 2 \rangle$  - mutační konstanta
- **CR**     $CR \in \langle 0, 1 \rangle$     - práh křížení
- **G**                       - počet generací

## 2. Tvorba populace – náhodné stanovení první generace

$$x_k^{i,0} = x_{k \min} + \mathit{rand}(x_{k \min}, x_{k \max}) \quad \text{for } i = 1, \dots, NP$$
$$k = 1, \dots, D$$

### 3. Reprodukční cyklus

Pro každého jedince (rodiče) je vygenerován zkušební jedinec  $z_k$ . Nejprve jsou vybráni tři různí jedinci téže generace. Rozdíl prvních dvou násobený konstantou  $\mathbb{F}$  je přičten ke třetímu jedinci. Výsledkem je tzv. šumový vektor  $v$ .

$$v_k^i = x_k^{r3,G} + F(x_k^{r1,G} - x_k^{r2,G}) \quad \text{for } i = 1, \dots, NP$$

Poté je pro každý rozměr generováno náhodné číslo  $\langle 0, 1 \rangle$ . Pokud je číslo menší než práh křížení  $CR$ , je zkušebnímu jedinci pro aktuální rozměr přiřazen parametr ze šumového vektoru. Je-li větší, pak parametr rodiče.

$$\text{if}(\text{nahoda} < CR) \longrightarrow zk_k^i = v_k^i \quad \text{for } i = 1, \dots, NP$$

$$\text{if}(\text{nahoda} \geq CR) \longrightarrow zk_k^i = J_k^i \quad k = 1, \dots, D$$

**Pozn.** Ze vzorce pro šumový vektor je patrný význam parametru  $\mathbb{F}$ . Čím větší parametr  $\mathbb{F}$  bude, tím „více jiný“ bude parametr šumového vektoru než parametr jedince č. 3. Parametr  $CR$  (práh křížení) určuje pravděpodobnost, se kterou se do zkušebního vektoru dostane parametr šumového vektoru. Důvod názvu diferenciální evoluce je nutno hledat rovněž v prvním vzorci. Název není odvozen od diferenciálního počtu, ale z prostého rozdílu vektorů.

#### 4. Testování cílové funkce $f$ pro zkušební jedince

Pro každého zkušebního jedince je testována cílová funkce  $f$ . Nástroj pro testování cílové funkce většinou není součástí evolučního algoritmu. Např. Fluent, Excel apod.

#### 5. Určení nové populace

Porovnáním hodnot cílových funkcí rodičů a zkušebních vektorů se určí nová generace jedinců. Pokud je cílová funkce  $f$  zkušebního vektoru  $zk$  lepší než jedince - rodiče, postoupí do nové generace zkušební vektor. Pokud ne, postoupí beze změny rodič. Tím je zaručeno, že se do nové generace nedostane horší jedinec než rodič.

$$\text{if}(f(zk_k^i) > f(J^{i,j})) \longrightarrow J^{i,j+1} = zk^i \quad \text{for } i = 1, \dots, NP$$

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

## Poznámka

Při reprodukčním cyklu se velmi často stává, že nově generovaný zkušební jedinec  $z_k$  je alespoň jedním parametrem mimo oblast  $\Omega$ . Řešení:

I. Zastavení jedince na hranici – pravděpodobnost nalezení lokálního extrému

II. Náhodné generování překročeného parametru, stejně jako při tvorbě prvotní populace. Jedná se o standardní typ Diferenciální evoluce.

**Pozn.** Řešení problému, když jedinec alespoň jedním svým parametrem opustí oblast, může být libovolné. Např. navrácení do první třetiny od překročené hranice. Může však nastat situace, kdy dojde ke zhoršení různorodosti populace. To může mít vliv na správné nalezení extrému. U řešení č. II je toto nebezpečí nižší.

# konkrétní optimalizační problém

Oblast	<b>Aerodynamika, užití CFD</b>
Cílová funkce	<b>Vztlak</b>
Metoda	<b>Diferenciální evoluce</b>
Popis	<b>Nalezení nejméně vhodného umístění a tvaru přitlačného křídla modelu závodního vozu</b>
Preprocesor	<b>Gambit</b>
CFD solver	<b>Fluent</b>
Program	<b>DevC++</b>

**Pozn.** Cílem práce bylo testování využití evolučního optimalizačního algoritmu v aerodynamice za použití CFD softwaru. Tedy odpovědět na otázku, zda je diferenciální evoluce vhodná při používání CFD výpočtů.

odpověďt na otázku: **Která konfigurace je aerodynamicky méně výhodná?**  
**Tj. která vykazuje vyšší vztlak?**

Lancer WRC

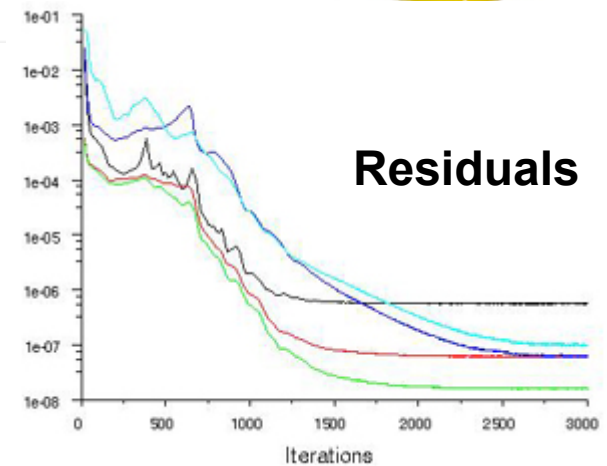
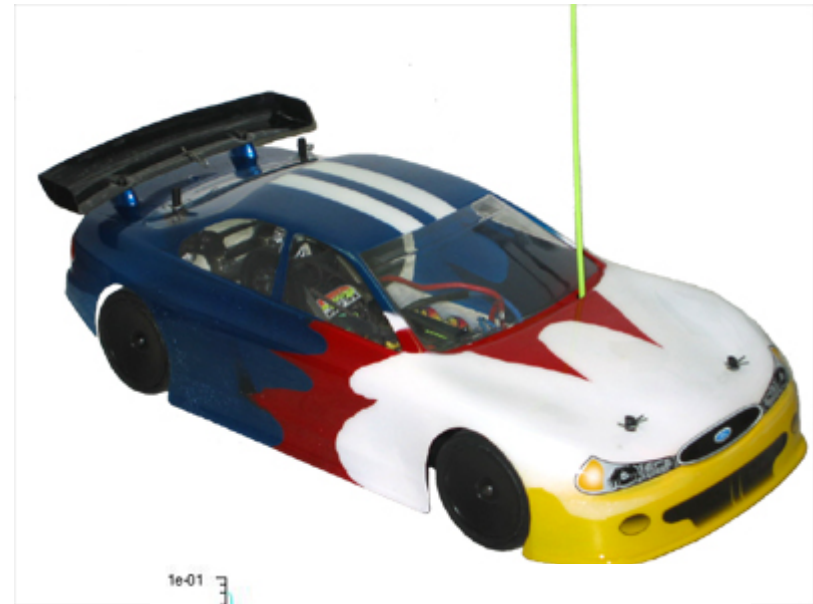
 MITSUBISHI MOTORS

Impreza WRC

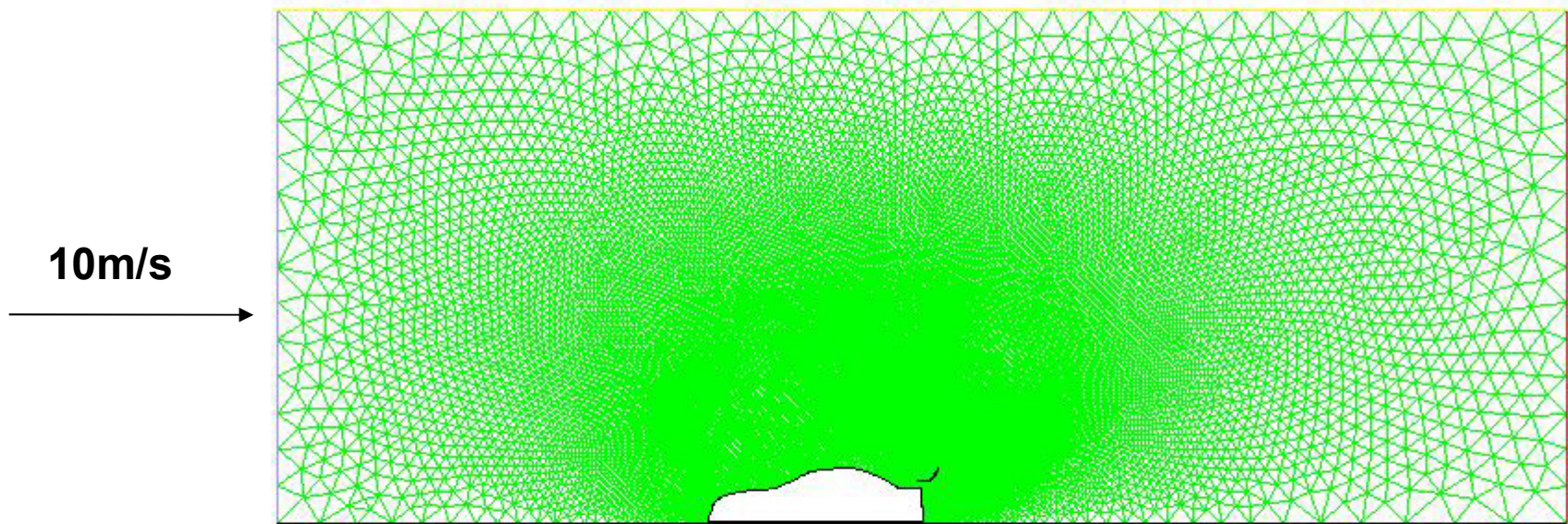


**Pozn.** Vyšetřovaným objektem bylo přítláčné křídlo a jeho tvar a umístění na voze.

Vůz:	<b>Ford Mondeo</b>
Měřítko:	<b>1/10</b>
Rychlost	<b>10m/s</b>
Model turbulence	<b>k – <math>\epsilon</math></b>
Počet iterací	<b>3000</b>
Schéma	<b>Upwind 1. řádu</b>



**Pozn.** CFD simulace byla prováděna pro model závodního automobilu v měřítku 1/10. Rychlost proudění byla volena tak, aby se blížila rychlosti, kterou modely třídy E 1/10 TC dosahují v rychlých zatáčkách závodních tratí.



Grid

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

**Pozn.** Sít' byla generována automaticky v programu Gambit(textový režim). Vstupními daty byly souřadnice bodů karoserie, předmětem optimalizace pak souřadnice bodů křídla.

parametr  $k=1$       **x-ová pozice křídla,  $x_k$**

$$x_k \in \langle 365, 416 \rangle \text{ [mm]}$$

parametr  $k=2$       **úhel natočení svislé plochy křídla  $\alpha_k$**

$$\alpha_k \in \langle 45, 90 \rangle \text{ [}^\circ\text{]}$$

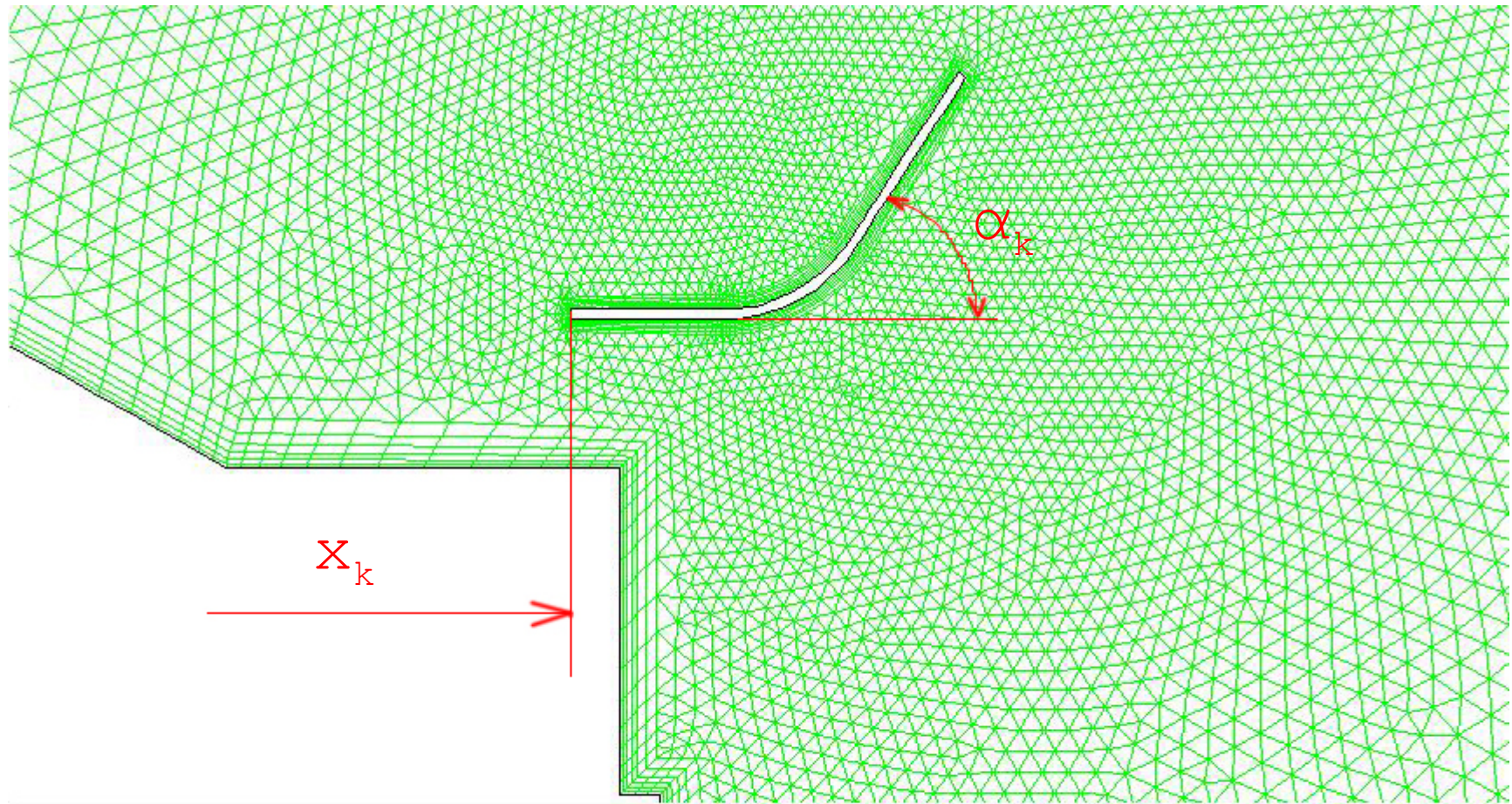
parametry Diferenciální evoluce:

$$NP = 4$$

$$F = 1,23$$

$$CR = 0,45$$

$$G = 4$$



Grid

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

**Pozn.** Vzdálenost  $x_k$  byla měřena od přední části vozu, kde byl umístěn souřadný systém.

# Průběh – nejlepší jedinec

## I. „zastavení na hranici“

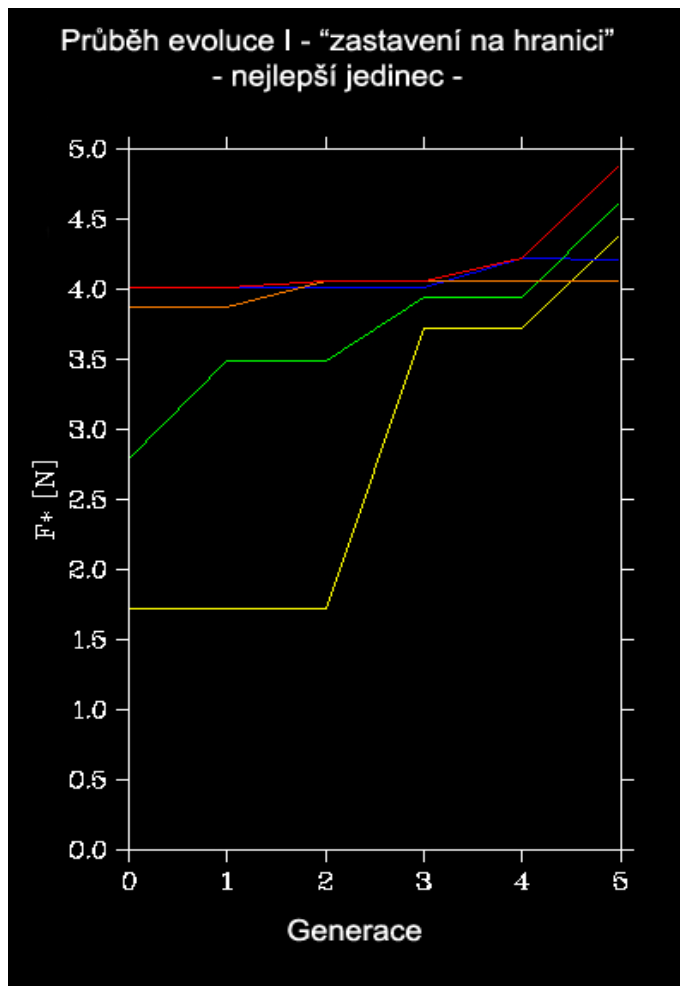
G	$X_k$ [mm]	$\alpha_k$ [°]	$F_y^*$ [N]
1	376.00	45.00	4.01
2	376.00	45.00	4.01
3	365.00	48.00	4.06
3	365.00	48.00	4.06
5	365.00	45.00	4.22

## II. „standard“

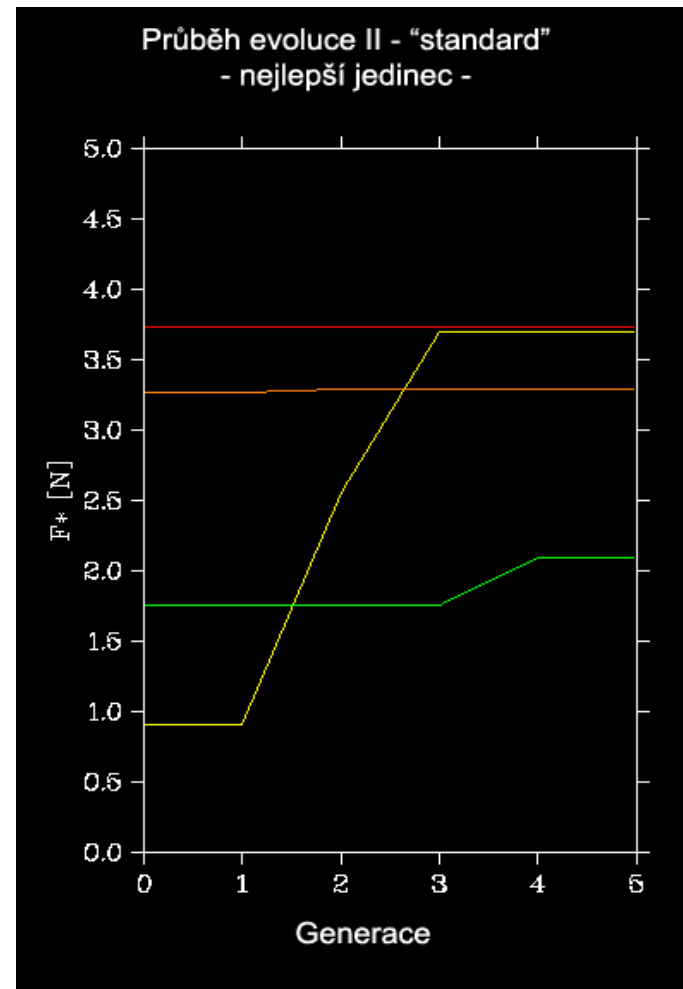
G	$X_k$ [mm]	$\alpha_k$ [°]	$F_y^*$ [N]
1	367.00	60.00	3.73
2	367.00	60.00	3.73
3	367.00	60.00	3.73
4	367.00	60.00	3.73
5	367.00	60.00	3.73

**Pozn.** Nejlepší jedinec v případě I.) byl znám až v poslední generaci, zatímco v II.) již v první. To je dílem náhody, která hraje, podobně jako v přírodě, významnou roli v diferenciální evoluci.

## I. „zastavení na hranici“



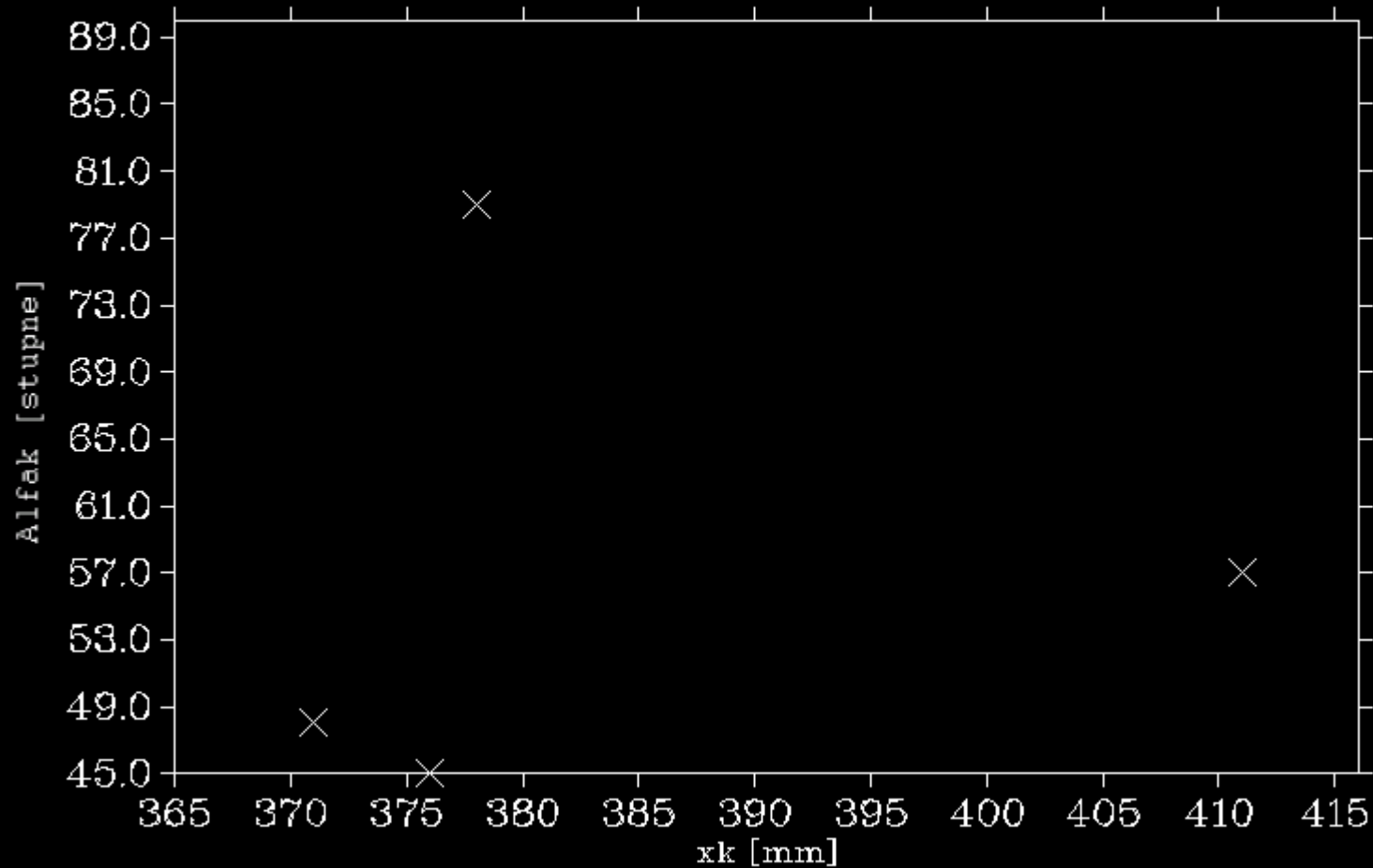
## II. „standard“



**Pozn.** Vývoj každého jedince je zobrazen čarou různé barvy, nejlepší řešení pak má barvu červenou. Z obrázku je vidět, že evoluce č. I. měla zajímavější průběh než evoluce č. II., kde nejlepší jedinec byl znám již v první generaci.

# I. „zastavení na hranici“

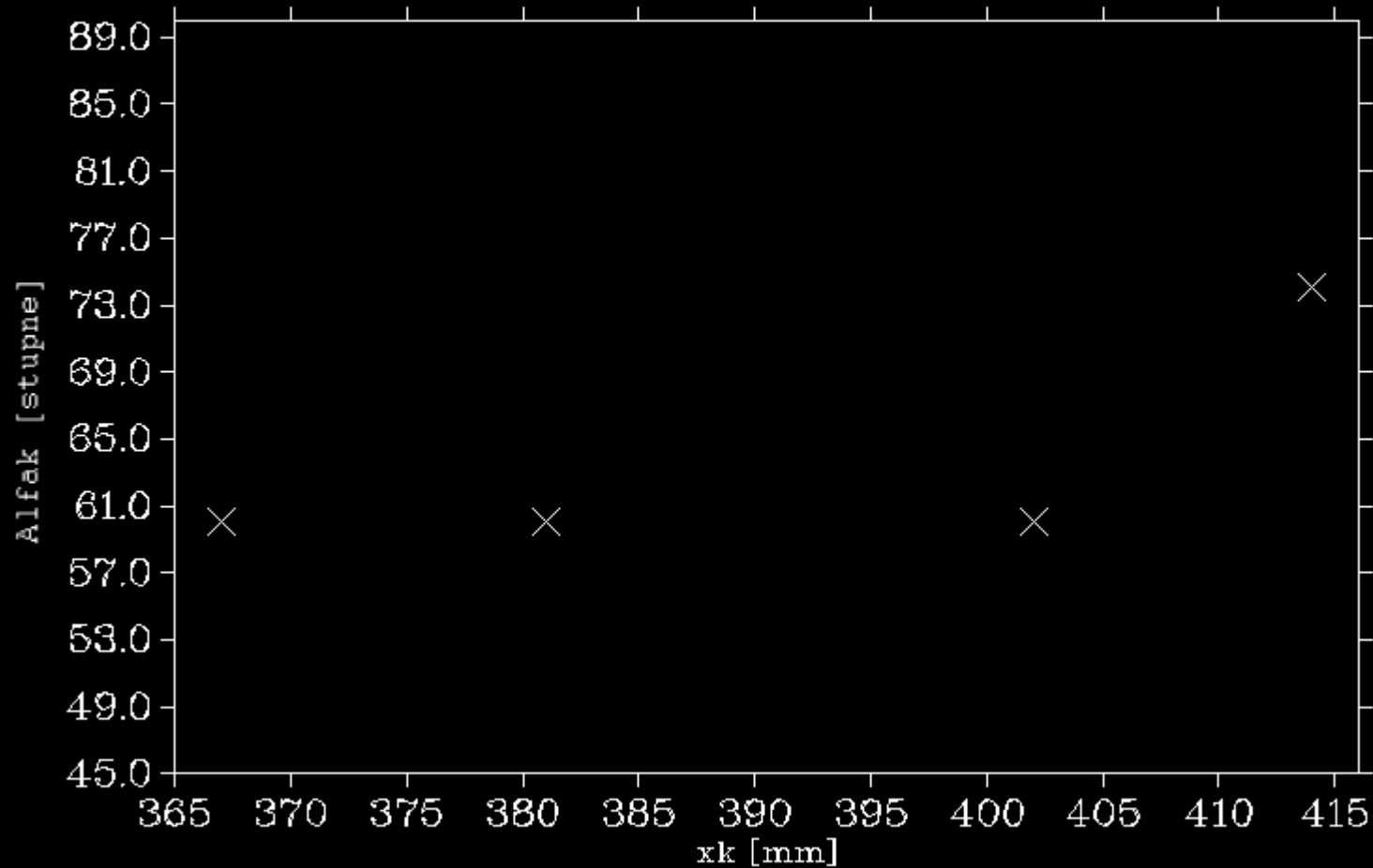
Průběh evoluce | generace č.1



**Pozn.** Obrázek představuje vývoj evoluce po jednotlivých generacích. První frame je 1. generace, tj. rozmístění jedinců při generování počáteční populace. Každý další frame je obrazem další generace. Rozmístění jedinců poslední generace odpovídá nebezpečí zastavení jedince na hranici při opuštění oblasti.

## II. „standard“

Průběh evoluce | generace č.1



**Pozn.** V poslední generaci se jedinci vyrovnají v jedné řadě vzhledem k  $\text{Alfa}_k$ . To by mohlo poukazovat na možnost nalezení optimálního úhlu natočení svislé části přítláčného křídla. Jasnou odpověď by poskytl až výpočet dalších generací.

V PDF nefunguje animace obrázku gif. Ten lze shlédnout na <http://www.suchomelplasty.cz/academics/>

Díky omezeným možnostem testování cílových funkcí v CFD programu a časové náročnosti výpočtu nebylo možné optimalizaci dovést až do konce. Pro jednoduché matematické funkce (rotační paraboloid) je potřeba stovky až tisíce testování cílové funkce.

Diferenciální evoluce je velmi mocný prostředek pro optimalizaci. Avšak velké procento výpočtů zkušebních jedinců je neúspěšné, tj. není generován lepší jedinec.

Proto bych pro aerodynamické optimalizační problémy bych doporučil spíše metody, které berou v potaz rovněž výsledky z předchozích výpočtů.

Mou představou je např. proložení regresní plochy testovanými body, výpočet gradientu a pokračování ve výpočtu ve směru nejvyššího stoupání.

**Pozn.** Data získaná ze špatných jedinců mohou být využita pro studium problému samotného, který je optimalizační metodou řešen.

To však se samotnou optimalizací již nesouvisí.

Ačkoli nebylo možné dovést optimalizaci až do konce, z dosažených výsledků plyne, že horší konfiguraci přitlačného křídla má Mitsubishi Lancer WRC.



**Pozn.** Výsledek optimalizace říká, že nejhorším místem pro montáž křídla je přední část zavazadlového prostoru. Což by mělo Mitsubishi znevýhodňovat. Praxe, tj. výsledky v Mistrovství světa v rally, ukazuje, že Mitsubishi Lancer je velmi konkurenceschopný. Odpověď lze hledat ve vertikálním umístění plochy vytvářející přitlak. Ta je na japonském voze umístěna téměř v rovině se střechou. K přitlačné ploše se dostane "kvalitní" vzduch generující přitlak. Vertikální pozice křídla nebyla předmětem optimalizace.

## **Autor**

Ondřej Suchomel

---

[ondra@suchomelplasty.cz](mailto:ondra@suchomelplasty.cz)

## **Vedoucí projektu**

Ing. Tomáš Hyhlík, FS ČVUT, Odbor Mechaniky tekutin a termodynamiky

## **Aktuální verze prezentace**

<http://www.suchomelplasty.cz/academics>

## **Literatura**

- [1] Mařík a kolektiv: Umělá inteligence (4), Academia, Praha, 2003
- [2] S. Kračmar, J. Vogel: Programovací jazyk C, Fakulta strojní ČVUT, ČVUT, Praha, 2002

## **Další zdroje**

<http://wrc.auto.cz>

<http://www.mitsubishi-motors.com>

<http://www.swrt.com>