



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA STROJNÍHO INŽENÝRSTVÍ

FACULTY OF MECHANICAL ENGINEERING

ÚSTAV AUTOMATIZACE A INFORMATIKY

INSTITUTE OF AUTOMATION AND COMPUTER SCIENCE

OPTIMALIZACE AERODYNAMICKÝCH VLASTNOSTÍ PROFILU

AEROFOIL AERODYNAMIC FEATURES OPTIMIZATION

DIZERTAČNÍ PRÁCE

DOCTORAL THESIS

AUTOR PRÁCE

AUTHOR

Ing. Jan Müller

ŠKOLITEL

SUPERVISOR

prof. Ing. Pavel Ošmera, CSc.

BRNO 2024

Abstrakt

Obsahem předložené disertační práce je pokročilá optimalizace profilu nosné plochy obecného letounu.

Pro optimalizaci je využito pokročilých metaheuristických optimalizačních technik, založených na evolučních výpočtech a rojových algoritmech. Pro tyto algoritmy je typická robustnost optimalizace a inženýrsky přijatelná míra konvergence a optimality řešení.

V rámci řešení byly navrženy a implementovány podstatné modifikace původních optimalizací orientované na profil. Z původního evolučního algoritmu (EA) byla vytvořena nová varianta optimalizace profilu evolučními algoritmy (aEA), následně pak byla z původní optimalizace hejnem částic (PSO) vyvinuta nová varianta optimalizace profilu hejnem částic (aPSO). Dále pak byla vytvořena vlastní hybridizace uvedených metod v paralelní variantě.

Pro proces optimalizace bylo využito modelu parametrizace Bezier-PARSEC 3434, generujícího tvar profilu. Pro optimalizace výchozího profilu byl použit parametrický model založený na B-Spline.

Pro verifikaci optimalizací bylo použito statistických testů tj. Mann-Whitneyův test. Pro použití s parametrickými modely, ale také pro meta-optimalizaci byl použit autokonfigurační nástroj IRACE.

Simulace fluidní dynamiky pro výpočet základních aerodynamických vlastností (vztlak, odpor, moment), byl realizován programovým vybavením Xfoil. Výsledky byly následně verifikovány pomocí simulace dynamiky tekutin (CFD ANSYS Fluent).

Z pohledu optimalizačních úloh, navržených optimalizací a implementací je zřejmé, že jde o komplexní mezioborovou úlohu, jejíž výsledky jsou prezentovány v této práci.

Klíčová slova

Evoluční algoritmy, hejnové algoritmy, paralelní hybridní algoritmy, Bezier-PARSEC model, profil, křídlo, nosná plocha, optimalizace, metaheuristika, Xfoil.

Abstract

The content of the presented thesis is advanced optimization of the aerofoil wing of a general aircraft.

Advanced metaheuristic optimization techniques based on evolutionary calculations and swarm algorithms are used for optimization. These algorithms are characterized by robustness of optimization and engineered degree of convergence and optimality of the solution.

Within the solution, fundamental modifications of the original aerofoil optimizations were designed and implemented. A new variant of aerofoil evolutionary algorithms (aEA) was created from the original evolutionary algorithm (EA), followed by a new variant of aerofoil particle swarm optimization (aPSO) developed from the original particle swarm optimization (PSO). Then the hybridization of the mentioned methods was created in a parallel variant.

The Bezier-PARSEC 3434 parameterization model that generates the aerofoil shape was used for the optimization process. A parametric model based on B-Spline was used to optimize the original aerofoil.

Statistical tests, i.e. the Mann-Whitney test, were used to verify optimizations. The IRACE autoconfiguration tool was used for use with parametric models, but also for meta-optimization.

Fluid dynamics simulation for the calculation of basic aerodynamic features (lift, drag, moment) was realized by Xfoil software. The results are then verified using fluid dynamics simulation (CFD ANSYS Fluent).

From the point of view of optimization tasks developed by optimization and implementation, it is clear that this is a complex interdisciplinary task, the results of which are presented in this thesis.

Keywords

Evolutionary algorithms, swarm algorithms, parallel hybrid algorithms, Bezier-PARSEC model, aerofoil, wing, aerofoil wing, optimization, metaheuristics, Xfoil.

Čestné prohlášení

Prohlašuji, že tato práce je mým původním dílem, vypracoval jsem ji samostatně pod vedením vedoucího práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury.

Jako autor uvedené práce dále prohlašuji, že v souvislosti s vytvořením této práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a jsem si plně vědom následku porušení ustanovení § 11 a následujících autorského zákona c. 121/2000 Sb., včetně možných trestně právních důsledků.

V České dne 10. 5. 2024

Jan Müller

Poděkování

Děkuji svému vedoucímu dizertační práce prof. Ing. Pavlu Ošmerovi, CSc., za jeho rady, připomínky a odbornou pomoc. Rád bych poděkoval také své rodině, která mě při vytváření této práce podpořila a bez jejichž pomoci by nebylo možné práci dokončit.

Obsah

1	Úvod	17
2	Cíle práce	19
3	Přehled optimalizačních algoritmů	21
3.1	Deterministické algoritmy	22
3.2	Stochastické algoritmy	22
4	Podstata metaheuristik	23
4.1	Evoluční algoritmy (Evolutionary Algorithms)	24
4.1.1	Genetické algoritmy (Genetic Algorithms)	24
4.1.2	Genetické programování (Genetic Programming)	25
4.1.3	Evoluční programování (Evolutionary Programming).....	26
4.1.4	Diferenciální evoluce (Differential Evolution).....	26
4.1.5	Evoluční strategie (Evolution Strategy)	27
4.1.6	Gramatická evoluce (Grammatical Evolution).....	27
4.1.7	Dynamická evoluční optimalizace (Dynamic Evolution Optimization)	28
4.2	Rojové (hejnové) algoritmy (Swarm Algorithms).....	29
4.2.1	Optimalizace hejnem částic (Particle Swarm Optimization).....	30
4.2.2	Samo Organizující se Migrační Algoritmus (SOMA).....	31
4.2.3	Optimalizace mravenčí kolonií (Ant Colony Optimization)	32
4.2.4	Algoritmus umělých včelích kolonií (Artificial Bee Colony)	33
4.2.5	Algoritmus svatojánských mušek (Firefly Algorithms)	34
4.3	Výběr metaheuristik	35
4.3.1	Optimalizace evolučních algoritmů.....	35
4.3.2	Optimalizace hejnem částic	39
4.3.3	Optimalizace PSO s vahou setrvačnosti	40
4.3.4	Optimalizace PSO varianta Many Optimizing Liaisons.....	42
5	Parametrizační metody	45
5.1	Přehled parametrizací	45
5.1.1	Hicks-Henneho funkce (Hicks-Henne Function)	45
5.1.2	Radiální bázové funkce (Radial Basis Functions)	45
5.1.3	Volná tvarová deformace (Free-Form Deformation)	46
5.1.4	Parametrizace CST (Class Shape Function)	47
5.1.5	Bezier parametrizace (Bezier Parameterization)	47
5.1.6	B-spline parametrizace (B-spline Parameterization)	48

5.1.7	PARSEC parametrizace (PARSEC Parameterization).....	49
5.1.8	Bezier-PARSEC parametrizace (Bezier-PARSEC Parameterization)	51
5.1.9	Bezier-PARSEC 3333 parametrizace (BP 3333 Parameterization)	52
5.1.10	Bezier-PARSEC 3434 parametrizace (BP 3434 Parameterization)	54
5.1.11	Bézierovy povrchy (Bezier Surfaces).....	56
5.2	Volba parametrizací.....	57
5.2.1	Bezier-PARSEC 3434 parametrizace	57
5.2.2	B-spline parametrizace	57
6	Statistické testy.....	59
6.1	Výčet testů	61
6.1.1	Mediánový test	62
6.1.2	Wilcoxonův párový test.....	62
6.1.3	Mann-Whitneyův test	63
6.1.4	Kruskal-Wallisův test	64
6.1.5	Friedmanův test	65
6.2	Vybrání testu.....	66
6.2.1	Mann-Whitneyův test	66
7	Nástroje pro automatickou konfiguraci	67
7.1	IRACE (Iterated Racing for Automatic Algorithm Configuration)	67
8	Optimalizační algoritmy profilu v historické souvislosti	71
9	Aerodynamické veličiny	73
9.1	Proudění vzduchu kolem profilu	73
9.2	Výpočtové aerodynamické veličiny	74
9.3	Výsledné aerodynamické veličiny.....	75
9.4	Zvolené výpočtové aerodynamické veličiny	76
10	Programy pro výpočet aerodynamických vlastností	77
10.1	Seznam výpočetních programů.....	77
10.1.1	Program Xfoil	77
10.1.2	Program ANSYS Fluent	85
10.1.3	OpenFOAM.....	90
10.2	Srovnání programů.....	90
10.2.1	Srovnání aerodynamických charakteristik profilu NACA 2412	90
10.2.2	Srovnání optimalizovaného profilu na extrém	91
10.2.3	Srovnání výsledků výpočetních metod u optimalizovaného profilu NACA 2412	92
10.3	Výběr programu	93

10.3.1	Program Xfoil	93
11	Vytvoření nových variant optimalizací.....	95
11.1	Varianta optimalizace aEA	95
11.2	Varianta optimalizace aPSO	97
11.3	Varianta optimalizace aPSO s výchozím profilem	99
12	Vytvoření Paralelně hybridní optimalizace PHO	101
13	Použití autokonfiguratoru IRACE.....	103
13.1	Autokonfigurace parametrizací profilu.....	103
13.1.1	IRACE použitý na parametrizaci Bezier-PARSEC 3434.....	103
13.1.2	IRACE použitý na parametrizaci B-spline	104
13.2	Autokonfigurace optimalizací.....	105
13.2.1	IRACE použitý na optimalizaci aEA.....	105
13.2.2	IRACE použitý na optimalizaci aPSO.....	105
13.2.3	IRACE použitý na optimalizaci aPSO s výchozím profilem	105
14	Výsledky optimalizačních algoritmů.....	107
14.1	Výsledky optimalizace aEA.....	108
14.1.1	Velikost turnaje 3 a 6 jedinců	108
14.1.2	Velikost turnaje 6 a 9 jedinců	109
14.1.3	Znovu vkládání nejlepších jedinců po 10 a 20 minutách	110
14.1.4	Velikost turnaje 6 a vkládání nejlepších jedinců po 20 minutách	111
14.2	Výsledky optimalizace aPSO.....	112
14.2.1	Omega -0.3328 a -0.3085, c2 2.8446 a 2.0273.....	112
14.2.2	Omega -0.3328 a 0.2840, c2 2.8446 a 1.9466	113
14.2.3	Omega -0.3328 a -0.2, c2 2.8446 a 2.8.....	114
14.3	Výsledky optimalizace PHO.....	115
14.3.1	Střídání aEA a aPSO po 15 a 30 minutách.....	115
14.3.2	Střídání aEA a aPSO po 15 a 45 minutách.....	116
14.3.3	Střídání aEA a aPSO po 45 minutách a aEA s 5 min. a aPSO s 25 min.	117
14.4	Srovnání výsledků optimalizačních algoritmů.....	118
14.4.1	Srovnání výsledků aEA a aPSO	118
14.4.2	Srovnání výsledků aPSO a PHO	119
14.4.3	Shrnutí srovnání výsledků optimalizací	119
15	Výsledky autokonfiguratoru IRACE	121
15.1	Výsledky autokonfigurace parametrizací profilu.....	121
15.1.1	Srovnání výsledků aPSO a IRACE použitého na parametrizaci B-spline.....	121

15.1.2	Srovnání výsledků IRACE použitého na parametrizaci Bezier-PARSEC	123
15.2	Výsledky autokonfigurace optimalizací	123
16	Výpočet optimálních aerodynamických vlastností	125
16.1	Výpočet aerodynamických vlastností	125
16.1.1	Výpočet extrému aerodynamické jemnosti	125
16.1.2	Výpočet průměru aerodynamické jemnosti při ustáleném letu	126
16.1.3	Profil NACA 2412 a jeho aerodynamické vlastnosti	128
16.2	Srovnání aerodynamických vlastností	130
16.2.1	Srovnání extrému aerodynamické jemnosti s NACA 2412.....	130
16.2.2	Srovnání průměru aerodynamické jemnosti při ustáleném letu s NACA 2412...	132
17	Výsledky práce	135
17.1	Srovnání výsledků optimalizačních algoritmů.....	135
17.2	Srovnání výsledků autokonfigurátoru.....	136
17.3	Srovnání výsledků aerodynamického odporu	137
17.4	Srovnání optimálních aerodynamických vlastností	139
18	Přínosy práce.....	141
19	Pokračování práce	143
20	Závěr	145
	Použitá literatura.....	147
	Seznam zkratk a symbolů	157

Přílohy

- A Seznam publikací
- B Seznam patentů
- C Výběr částí zdrojových kódů
- D Výsledky optimalizací
- E Výsledky aerodynamických vlastností

1 Úvod

Cílem práce je optimalizace profilu nosné plochy letounu a s tím spojené vytvoření numerického řešení optimalizace profilu.

Vývoj v letectví byl vždy spjat s ekonomikou, ekologií a bezpečností provozu. Díky nově vytvořeným optimalizacím a na nich postavenému programovému vybavení lze však dosáhnout dříve nemyslitelných výsledků.

Vzhledem k tomu, že letecká technika představuje multidisciplinární, mezioborovou oblast, je řešení výše uvedených požadavků značně obtížné a vyžaduje skloubení řady často protichůdných požadavků.

Aerodynamika má v těchto souvislostech klíčový význam a úzce souvisí s geometrií profilu nosné plochy letounu. Jeho optimalizace má pak zásadní význam pro řešení výše uvedených problémů.

Pro optimalizaci profilu není možné použít charakteristiku tvaru profilu někdy až několika set souřadnicemi, je tedy použita parametrizace s několika málo parametry.

Parametrizací míníme kvalitní definování geometrie profilu, kdy při co nejmenším počtu parametru je možné zachovat co největší diverzitu tvaru profilu. Při parametrizaci profilu je možné kontrolovat parametry, tak aby byl zaručen použitelný tvar profilu.

U geometrické parametrizace při co nejmenším počtu parametru musí být možné ovlivňovat poloměr náběžné hrany, polohu maximální tloušťky profilu, tloušťku a úhel odtokové hrany při zachování nezávislých řídicích parametrů na horním i dolním povrchu. [1].

Pro optimalizaci je nejdůležitější rychlost konvergence, udržení diverzity, nejlepší výsledky - tedy nejlepší ocenění účelovou funkcí, vyvarování se elitizmu. Účelová funkce, na které jsou testovány jednotlivé optimalizace, je založena na aerodynamické jemnosti.

Výsledkem zmíněných optimalizací by měl být kvalitativně nejlepší profil s co nejnižším aerodynamickým odporem, snižujícím spotřebu paliva a ekologickou zátěží.

Pro srovnání optimalizací lze použít statistických testů definujících vztah jednotlivých výběrů optimalizací. Důležitou roli při nastavení optimalizace sehrává použití autokonfiguratoru na parametrizaci, resp. na optimalizaci.

Na tuto optimalizaci aerodynamických vlastností profilu přímo navazuje řízení jeho geometrie dle mého vynálezu [2]. Idea vynálezu vychází z měnících se nároků na aerodynamiku [3] letounu. Jedná se o možnost měnit tvar profilu (morphing) za letu tak, aby chování letounu bylo bezpečné a současně i ekonomické.

2 Cíle práce

Primárním cílem je vytvoření nových variant původních optimalizací aplikovaných na aerodynamiku a vytvoření vlastní optimalizace. Prvním krokem bude výběr z ověřených optimalizací používaných na profil. K vybraným optimalizacím profilu bude vyhledán nejvhodnější parametrizační model.

Výsledky z optimalizací budou podrobeny statistickými testy. Dále budou získány výsledky z autokonfiguratorů, které budou později srovnány s výsledky optimalizací.

Dalším krokem je výběr programu pro výpočet aerodynamických vlastností profilu s důrazem na rychlost a přesnost kalkulace. Poté je nutné provést testování různých nastavení optimalizací tak, aby později byly použity nejlepší konfigurace optimalizací.

Při testování budou nejlepší optimalizace ty, které dosahující největší aerodynamické jemnosti. U nejlepších optimalizací budou vznikat nové, extrémně kvalitní profily s unikátními vlastnostmi.

Výčet cílů práce:

- vytvoření nových variant optimalizací z používaných optimalizací - v současné době jsou pro optimalizaci profilu nejčastěji používané optimalizační metody [1] genetických algoritmů (GA), dále jsou často používané algoritmy diferenciální evoluce (DE), optimalizace hejnem částic (PSO), Samo Organizující se Migrační Algoritmus (SOMA) [4] a optimalizace chování bakteriální (BFO). Pro lepší výkon a snadnější implementaci lze vyzkoušet i optimalizaci Many Optimizing Liaison (MOL),
- volba parametrického modelu profilu - cílem bude vybrat a případně adaptovat parametrizaci pro definici geometrie profilu. Nejčastější parametrizace pro zjednodušení tvaru profilu jsou Hicks-Henneho, Radiální bázové funkce, Volná tvarová deformace, parametrizace CST, Bezier, B-spline, PARCEC, Bezier-PARSEC parametrizace,
- generování nových profilů nebo modifikace existujících profilů - vybraní optimalizace a parametrického modelu může být použito buďto na vytvoření nových profilů za pomoci náhodné počáteční populace, resp. hejna nebo na pozměnění stávajících profilů,
- výběr ze statistických testů - jako jsou Mediánový, Wilcoxonův párový, Mann-Whitneův, Kriskal-Wallisův a Friedmanův test,
- využití nástrojů pro automatickou konfiguraci - IRACE, SPOT, SQUIRREL pro srovnání s novými variantami optimalizací a vlastní optimalizací,
- výběr programového vybavení pro výpočet aerodynamických vlastností - pro výpočet aerodynamických parametrů je hojně využíváno programové vybavení Xfoil a také výpočetní program pro simulaci dynamiky tekutin CFD ANSYS Fluent, dále pak např. program OpenFOAM.

Klíčovou částí v oblasti optimalizace profilu pomocí evolučních a rojových algoritmů je zejména vytvoření takových nových modifikovaných algoritmů, které budou schopny generovat a optimalizovat komplexní strukturu aerodynamických parametrů profilu.

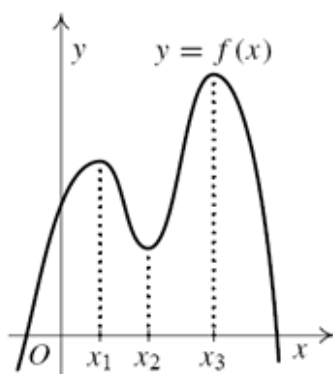
3 Přehled optimalizačních algoritmů

Optimalizace

Optimalizace je proces výběru nejlepšího objektu [5] podle daných kritérií z množiny dostupných objektů. V nejjednodušším případě jsou vstupní hodnoty systematicky dosazovány do funkce, kde dochází k výpočtu výstupních hodnot. Je vybráno řešení, kdy funkce dosahuje maxima nebo minima výstupních hodnot [6].

Optimalizační problém

Optimalizační problém spočívá v nalezení řešení, tj. nejlepších vstupních hodnot funkce. Mějme funkci $f(x)$, kde $x \in A$ a A je množina vstupních hodnot. Hledáme pak x takové, aby hodnota funkce $f(x)$ byla maximální, popřípadě minimální. Hovoříme pak o minimalizačním nebo maximalizačním problému [7].



Obr. 1: Příklad funkce $f(x)$ s více extrémy, která pro hodnotu x_3 dosahuje maxima.

Globální nebo také absolutní maximum, minimum

Globální, nebo také absolutní maximum je největší hodnota z maxim, globální, nebo také absolutní minimum je nejmenší hodnota z minim.

Lokální maximum, minimum

Pokud je mimo daný prostor větší nebo menší hodnota než maximum nebo minimum uvnitř prostoru, pak jde o lokální maximum nebo lokální minimum.

3.1 Deterministické algoritmy

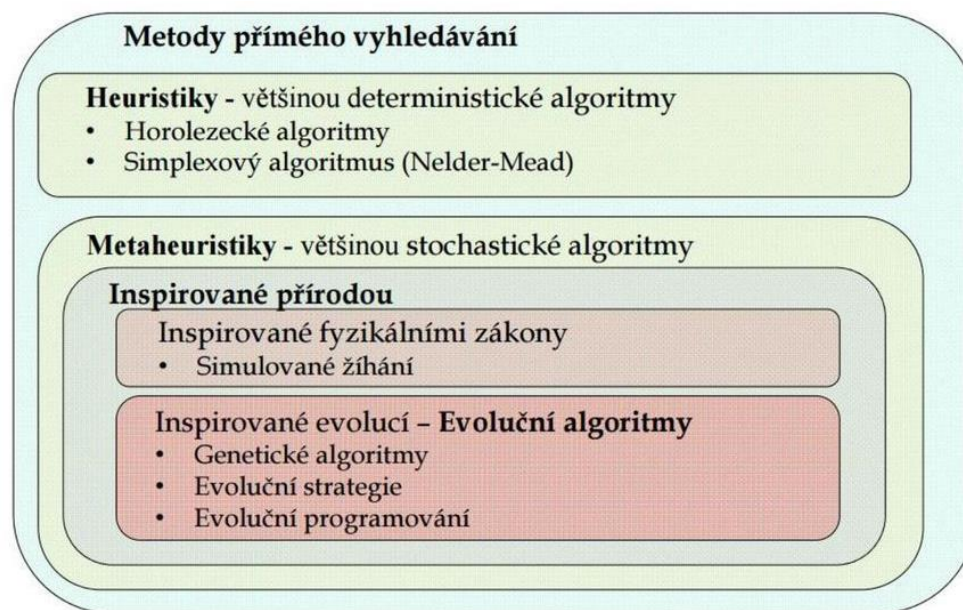
Deterministické algoritmy [8] vždy ze stejných vstupních hodnot svým během - průběhem iterací vytvoří stejné řešení, je tedy předvídatelné, tj. deterministické. Každý aktuální i následující krok vykonávání algoritmu je vždy jednoznačně definován.

3.2 Stochastické algoritmy

Stochastický (nedeterministický) algoritmu [9] v některých krocích může volit z několika možností dalších kroků. Stochastický algoritmus při stejném vstupu může dávat rozdílné výsledky. Stochastický algoritmus sice negarantuje nalezení řešení [10] v konečném počtu kroků, ale často může nalézt byť přibližné, tak prakticky použitelné řešení.

Heuristické algoritmy

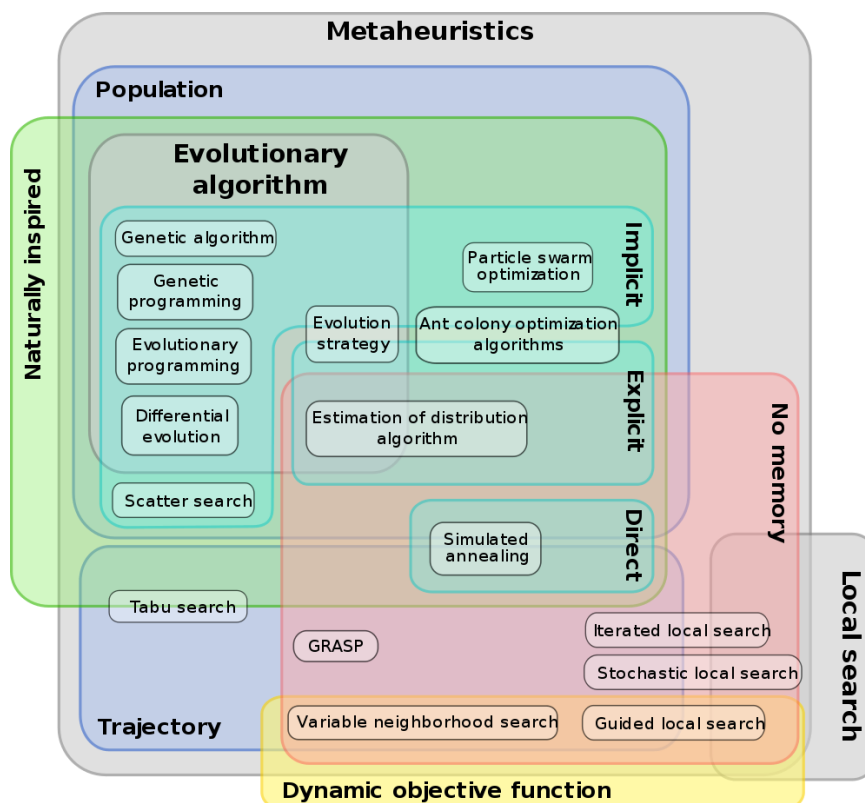
Heuristické algoritmy [10] jsou takové, které najdou řešení mezi všemi možnými, ale nezaručují, že naleznou to nejlepší, a mohou být proto považovány za přibližné, nikoliv přesné algoritmy. Tyto algoritmy však obvykle rychle naleznou řešení, které je blízko řešení optimálnímu.



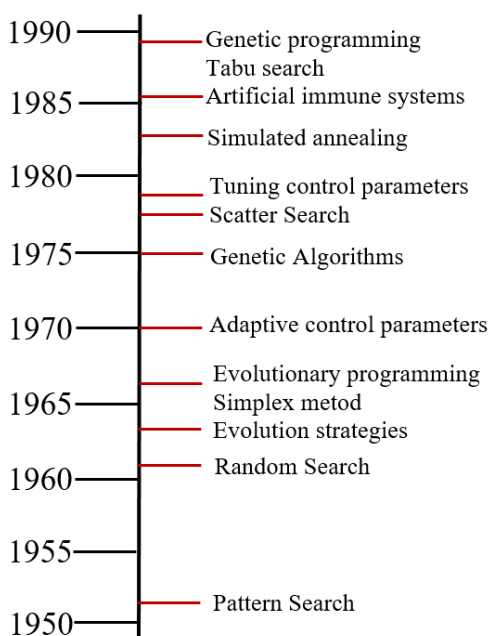
Obr. 2: Optimalizační algoritmy.

4 Podstata metaheuristik

Metody, které dokáží přímo řešit libovolnou třídu optimalizačních úloh, jsou označovány jako heuristiky, kdežto pokud je potřeba metody nějak funkčně upravovat, jedná se o metaheuristiky [11]. Pomocí metaheuristických algoritmů lze prohledávat stavový prostor úlohy, avšak bez záruky nalezení dostatečně kvalitního řešení v rozumném čase [12].



Obr. 3: Přehled metaheuristických algoritmů [13].



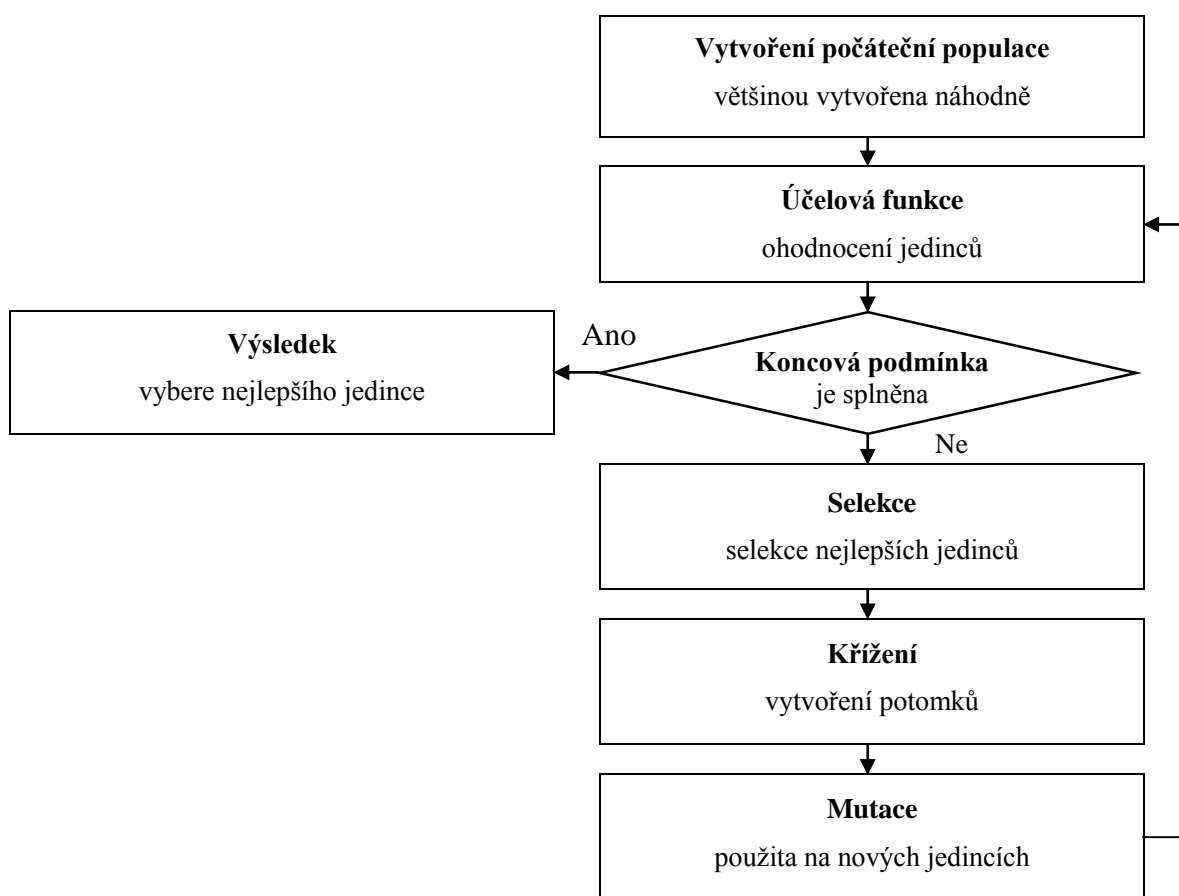
Obr. 4: Historie optimalizačních algoritmů.

4.1 Evoluční algoritmy (Evolutionary Algorithms)

Evoluční algoritmy používají techniky napodobující evoluční procesy známé z biologie [14] - dědičnost, mutace, přirozený výběr a křížení.

4.1.1 Genetické algoritmy (Genetic Algorithms)

Genetické algoritmy (GA) [15] jsou zřejmě nejpoužívanějšími metaheuristickými algoritmy. Jsou založeny na Darwinově teorii o vývoji druhů a simulují boj jednotlivých jedinců o přežití. Je to simulace přirozeného výběru [16], kdy přežívají nejsilnější jedinci. Nejsilnější jedinci se lépe přizpůsobují svému okolí, žijí déle, více se rozmnožují. Mají více potomků, kterým předávají svou genetickou informaci.



Obr. 5: Obecné schéma genetického algoritmu.

Genetické algoritmy [17] se převážně používají u složitých úkolů, většinou se jedná o rozsáhlé nelineární problémy, při rozhodování s velkými neurčitostmi s mnoho faktory ovlivňujícími výsledek řešení. Jsou používány v případě, kdy klasické metody neznají algoritmus řešení nebo jsou nepoužitelné pro datovou náročnost.

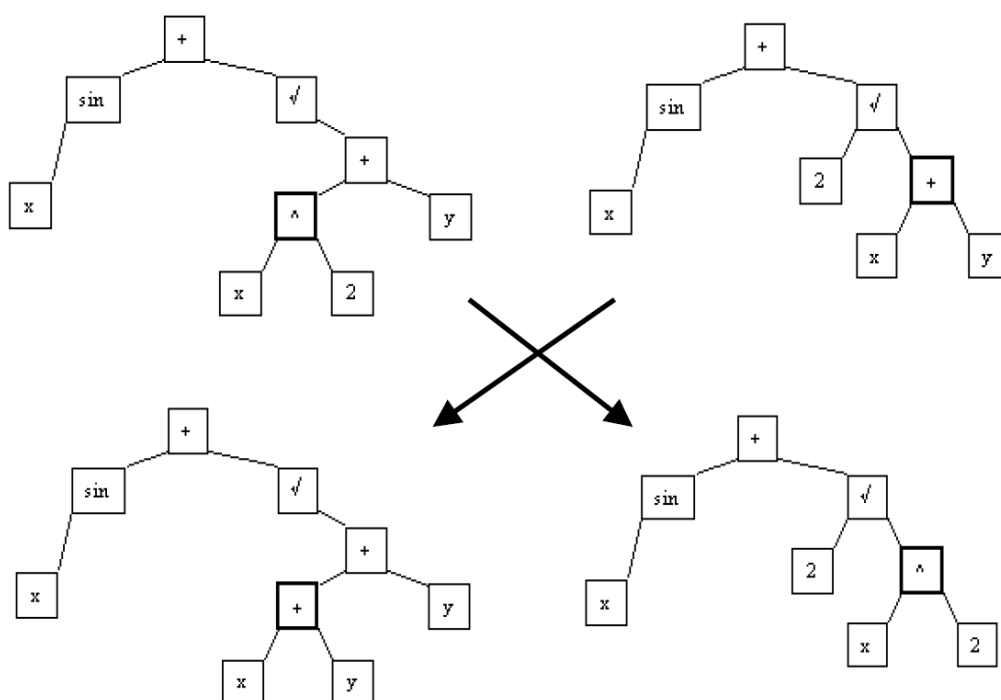
Genetickým algoritmem se nehledá konkrétní řešení, ale skupina možných řešení, z které je vybráno nejlepší řešení. Genetické algoritmy se používají v složitých problémech s mnoha parametry.

4.1.2 Genetické programování (Genetic Programming)

Genetické programování [18] je samostatnou skupinou v evolučních algoritmech, protože řeší úplně jiné problémy. Nehledá účelovou funkci, ale funkci samotnou, nějakou strukturu. Genetické programování lze považovat za rozšíření genetického algoritmu, přičemž kostra a princip jsou u obou algoritmů podobné. Rozdíl mezi oběma algoritmy [19] jsou ve vyjádření jedinců, na jiné třídy úloh genetického programování jsou jedinci vyjádřeni stromem s proměnlivou délkou chromozomu a tedy i jinými operacemi. Výsledkem genetického programování je program, nějaká spustitelná struktura.

Genetické programování je běžně používáno [20] k řešení úloh symbolické regrese, což znamená proložit data vhodnou funkcí, reprezentovanou syntaktickým stromem, jenž je sestaven různými matematickými operacemi, nezávislými proměnnými a konstantami. Výhodou genetického programování je, že k vyřešení problému je nutno pouze velmi malé znalosti o řešeném problému. Ovšem nevýhodou genetického programování je časová náročnost, protože pro každé ocenění řešení je nutné řešení spustit na trénovací množině, což bývá časově náročné. Další možnou nevýhodou genetického programování je výsledek algoritmu, tedy řešení nezaručuje plnou, stoprocentní funkčnost, výsledek algoritmu, funkce, program nemusí vždy vracet správná řešení. Pokud však je zapotřebí plné, stoprocentní funkčnosti, nelze optimalizační problémy řešit za pomoci genetického programování, i přesto že se dají tyto problémy genetickým programováním počítat.

Různé verze genetického programování různě kódují jedince a silně rozlišují mezi genotypem a fenotypem. Vhodnost se počítá podle srovnání chování vypočítaného programu a chování požadovaného programu.



Obr. 6: Křížení dvou stromů.

4.1.3 Evoluční programování (Evolutionary Programming)

Evoluční programování, [21] lze považovat za rozšířenou verzi evoluční strategie. Oba druhy algoritmu jsou stejné v tom, že používají pouze mutace a také, že do další populace jsou vybíráni pouze nejlepší rodiče a potomci. Mutací vzniká z každého rodiče jeden potomek a počet jedinců v populaci nemusí být konstantní. Rozdílem mezi těmito dvěma metodami je způsob realizace mutace a pozdější vývoj.

V přírodě hraje stejně jako u evoluční strategie [22] jednu z klíčových rolí vnitrodruhová konkurence, která může být charakterizována jako soutěž konkurujících si jedinců o potravu, zdroje, prostor a partnera. Evoluční programování [23] bylo určeno k předpovědi nových stavů a událostí u konečných automatů.

4.1.4 Diferenciální evoluce (Differential Evolution)

Stejně jako ostatní evoluční algoritmy používá diferenciální evoluce [24] mutaci, křížení a selekci. Rozdíl mezi diferenciální evolucí a genetickými algoritmy je v pořadí operací, napřed mutace, poté křížení, poté selekce. Optimální řešení u diferenciální evoluce se zakládá na mutaci, naproti tomu u genetických algoritmů nejlepší výsledky přicházejí použitím křížení.

U diferenciální evoluce pro vytvoření nového jedince je zapotřebí dalších čtyř jedinců. Ke každému jedinci jsou vygenerováni tři náhodně zvolení jedinci, ze kterých se mutací vytvoří šumový vektor. Poté je spočítána diference dvou ze tří zvolených jedinců, pak diferencí navážím a přičtu k třetímu ze tří zvolených jedinců. Křížení probíhá nad vybraným jedincem a šumovým vektorem za vytvoření pokusného vektoru. Pokusný vektor se vždy vytváří výběrem mezi parametry šumového vektoru parametry náhodného, čtvrtého jedince. Poté porovnáním mezi vzniklým pokusným vektorem a původně vybraným jedincem vyberu vhodnější, tedy provedu selekci. V základní verzi je algoritmus diferenciální evoluce daným počtem iterací.

Výhodou algoritmu diferenciální evoluce [25] je poměrně jednoduchá konstrukce a pouze čtyři řídicí parametry, tedy velikost populace, práh křížení, mutační konstanta a počet generací. Nevýhodou algoritmu diferenciální evoluce je častá stagnace, to znamená, že nedochází k zlepšování vhodnosti jedinců. Algoritmus diferenciální evoluce není inspirován přírodou, v přírodě neexistuje ekvivalent tohoto algoritmu.

Diferenciální evoluce má vícero verzí [26], lišících se hlavně způsobem zvolení jedinců k mutaci, vytvořením pokusného vektoru a mechanismu operátoru křížení. Pomocí algoritmu diferenciální evoluce je možné hledat globální minimum vícekritériálních funkcí.

4.1.5 Evoluční strategie (Evolution Strategy)

Algoritmus evoluční strategie [22] se historicky vyvíjel současně s genetickými algoritmy. Poprvé byl použit k řešení inženýrských úloh, později byl tento algoritmus použit pro numerickou optimalizaci.

Algoritmus evoluční strategie se řadí do populačních algoritmů, tudíž je inspirován vývojem konkrétní populace v čase. Taková populace je v genetice označována pojmem lokální populace. Pokud je vymezena oblast evolučního zájmu, lze zkonkretizovat význam v takto vymezené evoluci. V evoluci lokální populace je jednou z klíčových rolí vnitrodruhová konkurence, která může být charakterizována jako soutěž konkurujících si jedinců o potravu, zdroje, prostor a partnera. Mnoho lidí se mylně domnívá, že soutěžícím jedincům jde nejprve o přizpůsobení svému okolí. Nejprve však jde soutěžícím jedincům o přežití, rozmnožování a až pak o přizpůsobení svému okolí. V evoluci jedincům nejde nejprve být nejlepší vůči okolí a fyzikálním podmínkám, ale být lepší než jejich konkurenti.

V evoluční strategii [27] se oproti genetickým algoritmům [28] používá reálné kódování návrhových parametrů genů, není nutný převod z fenotypů do genotypů. Další možné rozdíly mezi evoluční strategií a genetickými algoritmy jsou v četnosti mutací a selekčním mechanismem. Evoluční strategie také podporují křížení více jedinců, které však genetické algoritmy mohou také používat. Původní verze evoluční strategie dokonce žádné křížení nepoužívaly. Rozdílem je, že v evoluční strategii vzniká křížením dvou jedinců jeden, kdežto v genetických algoritmech křížením dvou jedinců vznikají dva potomci.

Hlavní výhodou evolučních strategií je možnost upravovat své řídicí parametry, hlavně pak parametr udávající sílu mutace. Základní verze má tři řídicí parametry: sílu mutace a parametr k adaptaci síly mutace a maximální počet generací.

4.1.6 Gramatická evoluce (Grammatical Evolution)

Gramatickou evoluci [29] je možné považovat za jeden z nejnovějších typů genetického programování, v které lze vytvořit programy. Gramatická evoluce se od genetického programování liší tím, že je rozšířena o gramatiku. Proti genetickému programování má výhodu v tom, že je obecnější a použitelná i k vytvoření programů v libovolném jazyce, tedy pokud je dodržena Backus-Naurova forma [30] zápisu gramatiky. Jedinci jsou reprezentováni přímo binárními řetězci na rozdíl od stromů v genetickém programování.

Backusova-Naurova forma je způsob zápisu bezkontextových gramatik používaných pro popis formálních jazyků. Backus-Naurovu formu vytvořil John Backus pro popis syntaxe programovacího jazyka ALGOL a zdokonalil Peter Naur. Backus-Naurova forma se používá k zápisu gramatik počítačových programovacích jazyků, sad instrukcí a komunikačních protokolů. Dále lze využít i pro gramatiky skutečných jazyků.

Terminály a neterminály jsou základem gramatiky. Terminál je rozvíjen, dokud se z něho nestane nedeterminál. Startovací symbol a pravidla transformace nedeterminálu na další stav také patří do gramatiky.

4.1.7 Dynamická evoluční optimalizace (Dynamic Evolution Optimization)

Optimalizace v dynamických prostředích [31] je náročný úkol, protože mnoho problémů s optimalizací v reálném světě se v průběhu času mění. Evoluční výpočty jsou dobrými nástroji pro řešení problémů s optimalizací v dynamických prostředích díky jejich inspiracím z přirozených samoorganizovaných systémů a biologické evoluce, které vždy podléhaly měnícím se prostředím.

Každá dynamická funkce sestává z několika statických funkcí, kde pro každou statickou funkci je dáno pravidlo dynamiky, popisující, jak se v průběhu času mění podíl této komponenty na celém problému. Pravidlo dynamiky je definováno sledem transformací souřadnic, faktorů protažení souřadnic a fitness rescalings.

V případě dynamických problémů však úkol nalezení jednoho optima přechází na adaptivní proces, který aproximuje nejlepší řešení v každém kroku. Vzhledem k různým druhům dynamiky existuje mnoho možných aspektů adaptace, např. roje, sexuální adaptace, stíny a kernel metoda. Stínování představuje distribuci evolučně relevantní fenotypové variace zavedené do populace, např. důležitost každého z těchto aspektů přímo související s cíli, kterých chce dynamická optimalizace dosáhnout.

Kernel metoda [32] řeší konfiguraci evolučního algoritmu, který je v praxi empirický, nákladný a ad hoc proces. Znalost problémových domén má potenciál vést k cílenějšímu a efektivnějšímu návrhu EA. Většina stochastických vyhledávacích algoritmů pracuje na implicitním předpokladu, že „podobné“ body mají tendenci mít podobnou hodnotu. Funkce jádra pro konkrétní problémovou doménu je pak vytvořena učením se na základě explicitního a konkrétního modelu podobnosti. Na funkci jádra lze nahlížet jako na funkční podobnost, která dává pro hledané body v prostoru pravděpodobně podobné hodnoty jejich účelové funkce. Přizpůsobení funkce jádra na základě vzorku vyhledávacích bodů a z několika hodnot účelových funkcí je pak velmi efektivní a značně zkracuje dobu výpočtu.

4.2 Rojové (hejnové) algoritmy (Swarm Algorithms)

Historie

V dnešní době je čím dál tím větší důraz kladen na to, aby vše bylo provádělo rychleji, efektivněji a úsporněji. Jedním z možných východisek je používání takzvané rojové inteligence, která je odvozena z chování různých společenství v přírodě. Už v minulosti bylo pozorováno zvláštní chování rojů hmyzu, stád zvířat nebo hejn ptáků či ryb [33].

První výzkumy chování organizovaného společenství provedl v 60. letech minulého století Brian Partridge na univerzitě v Miami. Jeho výzkum se zabýval chováním hejn tresek v nádrži. Zjistil rozbořem nafilmovaných záběrů, že se tresky řídí dvěma jednoduchými pravidly: následovat tresku vpředu a udržovat rychlost s rybou vedle. Podobnými pravidly se řídí i lidé v davech, které by se daly také považovat za superorganizmy. Společenstva organismů, které se chovají jako jeden celek, se nazývají superorganizmus.

Rojová inteligence

Rojová inteligence [34] je technikou umělé inteligence, která je založená na studiu kolektivního chování decentralizovaných samo organizačních systémů. Systémy rojové inteligence jsou typicky tvořeny populacemi jednoduchých jedinců, ovlivňujících se, nebo komunikujících buďto přímo mezi sebou nebo nepřímo skrze okolní prostředí. Přestože každý jedinec provádí jen jednoduché akce, tak (každý jedinec) přispívá ke správnému fungování celku. Lokální interakce mezi jednotlivci a jednoduchými vzory chování jednotlivců často vedou k emergenci globálního chování. Emergencí je míněna nelineární interakce jednoduchých pravidel, vedoucích k složitějšímu chování celku [35]. Lokální změna chování části hejna přispívá ke změnám chování celku.

Fungování algoritmu [36] je ovlivněno pozitivní zpětnou vazbou, negativní zpětnou vazbou, zesílením změn a vícenásobnou interakcí. Pozitivní zpětná vazba znamená, že jedinec s výhodnější hodnotou k sobě přitahuje další jedince a ti přitahují další, čímž dochází k zahušťování řešení v okolí lokálního optima a zároveň může dávat vznik novým strukturám. Negativní zpětná vazba znamená, že jedinec se přesouvá od méně výhodného jedince k těm jedincům, kteří dávají výhodnější hodnotu. Zesilováním změn rozumíme, když jedinci náhodně hledají nové lokální optima. V přírodě, když jedinec, třeba mravenec, opustí již nalezenou cestu k potravě, je možnost, že najde lepší, kratší cestu. Vícenásobnou interakcí rozumíme, když jedinec může vytvářet stabilní samo organizační strukturu díky interakci s ostatními jedinci, čímž dochází k posilování dané struktury.

4.2.1 Optimalizace hejnem částic (Particle Swarm Optimization)

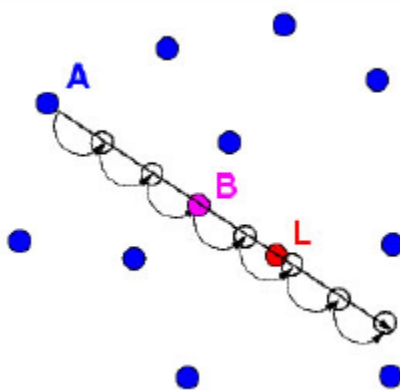
Optimalizace hejnem částic [37], je optimalizační meta-heuristická technika v oblasti umělé inteligence, inspirovaná chováním hejna ptáků nebo ryb. Optimalizace vychází ze simulace chování rojů částic při hledání potravy. Každá částice je definována svojí polohou, rychlostí a pamětí předchozích úspěchů při hledání. Částice jsou ovlivňovány ostatními úspěšnějšími částicemi hejna. Optimalizace spočívá v synchronním prohledávání prostoru přípustných řešení skupinou spolupracujících částic. Pozice částice v prostoru je dána hodnotami hledaných parametrů a rychlostí pohybu částic. Během jednotlivých iterací optimalizace si částice nezávisle na sobě nastavují směr a rychlost pohybu, o níž hovoříme jako o rychlostním vektoru. Rychlostní vektor si částice nastavují podle částice, která má v populaci dosud nejlepší nalezenou vhodnost, svou nejlepší dosaženou pozici v prostoru, a částice, která má nejlepší vhodnost ve vztahu k jejich okolí. Okolí částic je ve většině případů definováno jako neorientovaný graf, určující, které částice spolu mohou komunikovat a které ne. Formálně užívané komunikaci se přezdívá topologie sousedství [38]. Sousedství částic způsobuje, že částice nejsou oddělenými jednotkami, ale navzájem propojenou sítí, v níž se všechny navzájem ovlivňují. Celá populace částic se pohybuje prostorem přípustných řešení a je přitahována do slibných míst. Základní verze optimalizace [39] nemá zavedeno sousedství, které zmírňuje tendenci algoritmu předčasně konvergovat. Tato rozšířená verze je dnes považována za standard. Jelikož pohyb částic je stochastický, do úpravy jejich pozice je začleněn i prvek náhody.

Nevýhodou optimalizace hejnem je vyšší počet řídicích parametrů, což je počet částic, vliv setrvačnosti, vliv nejlepšího řešení souseda, maximální rychlost částic, počet částic v sousedství a maximální počet iterací. Existuje značné množství modifikací optimalizace hejnem, zvláště pak pro vícekritériální optimalizaci. Typ Speciation Particle Swarm Optimization [40] vytváří více skupin hejn, ve kterých se vyskytuje nějaký vedoucí jedinec, podle kterého se všechny částice řídí. Příslušnost částic k hejnu i vedoucímu jedinci se mění v každé iteraci. Ve verzi Nicheing Particle Swarm Optimization se pomalu během iterací na základě sousedství oddělují skupinky, které prohledávají slibná místa prostoru přípustných řešení individuálně.

4.2.2 Samo Organizující se Migrační Algoritmus (SOMA)

Pro optimalizační algoritmus SOMA [1], [4], [41], [42], [43] je důležitá práce s populací hlavně její interakce, strategie a spolupráce mezi jedinci. Algoritmus SOMA má pořád stálou populaci, kdy jedinci pouze mění svou polohu. Samoorganizace prostoru vzniká v důsledku vzájemné interakce mezi jedinci v průběhu jejich pohybu.

Souřadnice jedince ve stavovém prostoru jsou chápány jako hodnoty parametrů, což umožňuje dobrou geometrickou interpretovatelnost a detailní vykreslení průběhu algoritmu. Je vybrán náhodný počet jedinců, kterých se vyhodnotí jejich ocenění. Jedinec s nejlepším oceněním se definuje jako lídr. Pak je vytvořena další populace, kdy každý jedinec, kromě lídra, se pokusí najít vhodnější polohu skoky určité délky po přímce směrem k lídrovi. Skoky mají normální rozdělení se střední hodnotou, kterou je délka skoku a rozptylem způsobujícím nárůst diverzity v populaci. Po absolvování celé cesty je jedinec v další populaci usadí na nejvýhodnějším místě z celé trasy.



Obr. 7: Znáornění průběhu algoritmu SOMA [43]

Jedinec (A) migruje směrem k lídrovi (L).

Jedinec se pohybuje skoky a ohodnocuje každou skokovou pozici (prázdné kroužky). Po průchodu celé trajektorie se usadí na pozici s nejlepším ohodnocením (bod B). Původní populaci tvoří modré body a červený lídr.

Variací SOMA je mnoho, pro příklad dvě základní:

All-to-one - „Všichni k jednomu.“ Základní variace algoritmu, kdy všichni jedinci z populace migrují k statickému lídrovi.

All-to-All – „Všichni ke všem.“ V této variaci neexistuje lídr, každý jedinec migruje ke všem ostatním, což zvyšuje výpočetní náročnost, prohledávaný prostor a pravděpodobnost nalezení globálního extrému.

Dalšími variantami jsou například: All-to-One-Rand – „Všichni k jedinému náhodně“.

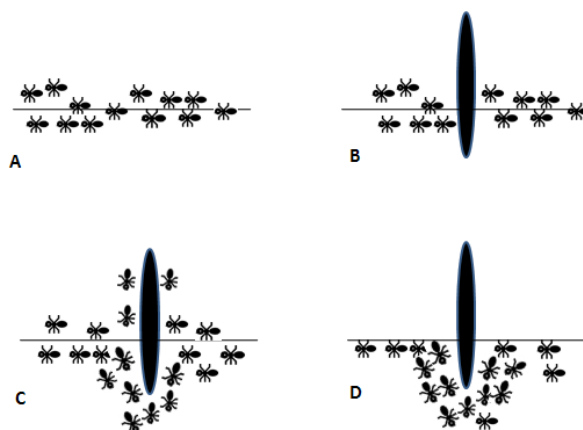
4.2.3 Optimalizace mravenčí kolonií (Ant Colony Optimization)

Optimalizace mravenčí kolonií [44] patří do skupiny rojové inteligence. Používá se pro řešení složitých kombinatorických úloh (například problém obchodního cestujícího).

Optimalizace mravenčí kolonií je inspirována sociálním chováním mravenců v přírodě při shánění potravy. Ti při hledání potravy vylučují feromony. Tyto feromony jsou detekovatelné pro ostatní mravence a slouží jim k vyhledání zdrojů potravy. Tento způsob komunikace založený na feromonech umožňuje mravencům nalézt nejkratší cestu od mraveniště k potravě. Po určitém čase se nejkratší cesta k potravě projevuje vyšší koncentrací feromonů, protože po této cestě se pohybuje většina mravenců vylučujících feromony. Poprvé tento postup popsal v roce 1991 Coloni, Dorigo a Maniezzo.

Optimalizace mravenčí kolonií [45] funguje podobným způsobem jako v přírodě. Umělí mravenci se v této optimalizaci pohybují po neorientovaném ohodnoceném grafu, symbolizujícím daný optimalizační problém. Možné cesty mravenců jsou možnými řešeními. Tento graf je paralelně procházen umělými mravenci, hledajícími různá řešení úlohy. Váha každé hrany grafu odpovídá množství feromonu, který na ni byl nanesen, a preferenci, se kterou se po ní mravenci vydají. Optimalizace je vylepšena zavedením postupného vypařování feromonů, aby nepoužívané hrany grafů mravence příliš nepřitahovaly. Váhy hran grafu se po každé iteraci aktualizují, podle počtu průchodu mravenců danou hranou v závislosti na kvalitě řešení, tj. zdroje, ke kterému daná cesta mravence vedla.

Základní verze algoritmu optimalizace mravenčí kolonií má tři řídicí parametry, vliv feromonů, vliv délky hrany grafu a sílu vypařování feromonů. Dnes se optimalizace mravenčí kolonií používá převážně ve dvou typech problémů. Pro těžké kombinatorické problémy a pro dynamické problémy nalezení nejkratší cesty.



Obr. 8: Principy mravenčího hledání potravy.

- A - mravenci hledají cestu mezi hnízdem a zdrojem potravy,
- B - objevila se překážka na cestě, mravenci se rozhodují, zda se vydají vpravo nebo vlevo, pravděpodobnost obou možností je stejná,
- C - feromon je rychleji ukládán na kratší cestu,
- D - všichni mravenci vybírají kratší cestu.

4.2.4 Algoritmus umělých včelích kolonií (Artificial Bee Colony)

Algoritmus umělých včelích kolonií [46] patří do skupiny algoritmů rojové inteligence.

Algoritmus umělých včelích kolonií [47] je inspirován způsobem, jakým včely získávají potravu v přírodě. Včela medonosná (*Apis mellifera*) je typickou ukázkou eusociálního hmyzu, pro který je charakteristické, že ve společenství se jedinci starají o své potomky, společnost je vícegenerační, jedinci se rozlišují na reprodukční jedince a nereprodukční jedince. Všichni jedinci se starají o blaho celého společenství. Pro tento hmyz je typická dělba práce, která probíhá podle kastovního systému, tedy podle toho, zda je jedinec matka, dělnice nebo trubci, ale také na základě tříd. Nejpočetnější třídou jsou dělnice, které jsou schopné vykonávat vícero povolání, v případě nutnosti mohou dělnice vykonávat úkoly, které jim nejsou vlastní. Dělnice hledají zdroj potravy v okolí úlu, a pokud jsou úspěšné, sdělí ostatním včelám v úlu včelím tancem vzdálenost zdroje od úlu, směr ke zdroji, bohatost potravy a její vůni. A feromony, vylučované z těla dělnice, si značkují zdroj potravy, což napomáhá ostatním včelám v nalezení zdroje.

V algoritmu umělých včelích kolonií [48] množina zdrojů potravy je v zásadě množinou řešení optimalizačního problému a odpovídá kvalitě řešení daného problému. Algoritmus se zabývá pouze způsobem získávání potravy včelami. Algoritmus spočívá ve spolupráci tří druhů umělých včel.

Prvním druhem umělé včely jsou zpravodajky. Každé zpravodajce je přiřazeno jedno řešení. Jejich úkolem je najít v okolí přiřazeného řešení ještě lepší výnosnější řešení, lepší potravu. O nalezených nejvýnosnějších zdrojích potravy, tedy o nejlepším řešení, informují rekrutky včelím tancem. V přírodě by zpravodajky odpovídaly dělnicím, sdělujícím včelím tancem ostatním dělnicím údaje o výnosném zdroji potravy.

Druhým druhem umělé včely jsou rekrutky, jejichž úkolem je následovat zpravodajku a poté hledat v okolí řešení zpravodajky vlastní lepší řešení. Počet rekrutek následujících danou zpravodajku je přímo úměrný kvalitě řešení, výnosnosti zdroje potravy. To znamená, že zpravodajky prezentující nejlepší řešení, tedy nejvýnosnější zdroj potravy, mají také nejvíce následujících rekrutek. Pokud rekrutky najdou lepší řešení nežli řešení zpravodajky, dochází k přesunu zpravodajky k lepšímu řešení rekrutky. V optimalizaci by tedy rekrutky a zpravodajky sloužily jako lokální prohledávací techniky slibných oblastí nejlepších v prostoru přípustných řešení.

Třetím druhem umělé včely jsou průzkumnice, které se v algoritmu nevyskytují pořád. Průzkumnicemi se stávají zpravodajky, kterým se v limitovaném počtu iterací nedaří najít lepší řešení, tedy lepší zdroj potravy než jejich. Pokud k tomu dojde, průzkumnice náhodně zvolí bod v prohledávaném prostoru, který se pak stane jejich novým řešením. Potom průzkumnice promění zpátky v rekrutky. Účelem existence průzkumnic je hledání optima v nových lokalitách. V přírodě jsou průzkumnicemi dělnice, hledající nové a výnosné zdroje potravy, popřípadě založení nového roje. Tyto průzkumnice vyhledávají často jakékoliv řešení, nejen to dobré, tedy zřídka nachází lepší řešení. Důležité ale je, že zvyšují „diverzitu“ známých řešení [49].

4.2.5 Algoritmus svatojánských mušek (Firefly Algorithms)

Algoritmus svatojánských mušek [50] je inspirován bioluminiscencí světlušek. V přírodě svatojánské mušky na sebe poutají pozornost nápadným světélkováním, které slouží k přilákání jedince opačného pohlaví. Podobným způsobem [51] fungují i umělé svatojánské mušky, které se pohybují v prostoru přípustných řešení a světélkováním se k sobě přitahují. Je dobré zmínit, že umělé svatojánské mušky jsou bezpohlavní a všichni přitahují všechny. Jak mnoho jsou k sobě umělé mušky přitahovány, závisí na jejich vhodnosti pro řešení optimalizovaného problému a na vzdálenosti mezi potencionálním párem. V algoritmu [52] se vyskytuje taktéž absorpce světla snižující přitažlivost mušek. Během každé iterace umělé mušky upravují svou pozici k ostatním muškám v populaci a pohybují se k nejkvalitnější mušce. Jak se pohybují, je dále ovlivněno náhodným vektorem, pro jehož generování se používají různá pravděpodobnostní rozdělení. Pokud se v okolí mušky nenachází kvalitnější jedinec, dojde pouze k náhodnému pohybu.

Algoritmus svatojánských mušek má tři řídicí parametry: absorpční koeficient, sílu přitažlivosti a sílu náhodné složky. Původně byl vyvinut pro řešení jednokriteriálních úloh, později byl samotným autorem Xin-She Yangem upraven na řešení vícekriteriálních úloh.

4.3 Výběr metaheuristik

V současné době jsou pro optimalizaci profilu nejčastěji používané optimalizační metody genetických algoritmů (GA), dále jsou často používané algoritmy [1] diferenciální evoluce (DE), optimalizace hejnem částic (PSO), Self-Organizing migrating algorithm (SOMA) optimalizace chování bakteriální (BFO). Pro lepší výkon a snadnější implementaci lze použít i optimalizaci Many Optimizing Liaison (MOL).

4.3.1 Optimalizace evolučních algoritmů

Evoluční algoritmy jsou zastřešujícím termínem pro různé přístupy využívající modely evolučních procesů. Evoluční algoritmy [16] se především používají pro řešení velkých komplexních optimalizačních problémů s mnoha lokálními extrémy, protože je zde menší pravděpodobnost, že uváznou v lokálním extrému, než je tomu u tradičních gradientních metod. Důležitým rysem evolučních algoritmů je jejich generační prohledávací strategie.

Historie

Jean-Baptiste Lamarck [53], vytvořil roku 1801 první evoluční teorii, nazvanou lamarckismus. Předpokládá, že znaky, získané během života na základě častého používání dané vlastnosti, jsou dědičné a přenášejí se bezprostředně na potomky. Charles Darwin [54] vytvořil komplexnější a věrohodnější evoluční teorii v díle O vzniku druhů (1859). Evoluci vysvětloval principem přírodního výběru. Neustálé soupeření o přežití, kde silní (vzhledem k prostředí) přežívají a slabí hynou, vede k adaptaci populace na prostředí.

Za zakladatele genetiky je pokládán Johann Gregor Mendel. 17 let trvajícím experimenty v roce 1865 prokázal dědičnost v živých organizmech. Mendelovy zákony byly však do konce 19. století nepovšimnuty. Neznal je ani Darwin, ani jeho bratranec Francis Galton, zkoumající dědičnost nadání a psychických vlastností v rodových liniích (1865), ale ani trojice Hugo de Vries, Carl Correns, Erich von Tschermak, která v roce 1900 dospěla ke stejným závěrům. K dalšímu vývoji genetiky přispěla řada autorů. Archibald Edward Garrod odhalil příčiny metabolických dědičných chorob (1902), Godfrey Harold Hardy a Wilhelm Weinberg nezávisle na sobě v roce 1908 popsali genetické chování populací v Hardyho-Weinbergově zákoně [55], [56]. Calvin Blackman Bridges objevil roku 1916, že geny jsou v buněčných jádrech uloženy v chromozomech.

V roce 1953 vzniká nový obor - molekulární genetiky, za jejíž zakladatele jsou považováni James Dewey Watson, Francis Crick a Maurice Wilkins. Za popis struktury DNA je jim v roce 1962 udělena Nobelova cena.

První pokusy o aplikaci evoluce v optimalizaci začínají s nástupem výpočetní techniky v 60. letech 20. století. Hans-Paul Schwefel a Ingo Rechenberg se jako první zabývali evolučními strategiemi.

Jedním z nejpoužívanějších evolučních algoritmů je genetický algoritmus, který byl poprvé popsán Johnem Hollandem v roce 1975 v jeho článku *Adaptation in Natural and Artificial Systems*. Je založen na Darwinově teorii o vývoji druhů a simuluje boj jedinců o přežití. K masivnímu rozvoji použití genetického algoritmu dochází v posledních třiceti letech díky prudkému zvýšení výkonu výpočetní techniky.

Selekce (Selection)

Cílem selekce [57] je výběr vhodných rodičovských párů pro vytvoření nových potomků. Je mnoho možných strategií selekce, ale všechny by měly splňovat dvě základní pravidla. První pravidlo je, že jedinci s lepším fitness ohodnocením mají větší šanci se účastnit reprodukce, druhým pravidlem je, že výběr rodičovských párů je podmíněn náhodně.

Dále pak metoda selekce musí být vybrána tak, aby v populaci vybraných jedinců byla zachována rozmanitost jedinců. Při příliš volném selekčním mechanismu evoluční proces postupuje velmi pomalu, vytváří se mnoho generačních cyklů bez zřetelného pokroku.

Ruletový výběr (Roulette Wheel Selection)

Ruletový výběr [58] je zřejmě nejčastěji využívaný způsob selekce. Jedinci jsou vybráni s pravděpodobností, úměrnou svými hodnotami ocenění. Pravděpodobnost $p(i)$, s jakou bude tedy i – tý jedinec vybrán, je dána následovně.

Ruletový výběr byl použit na počátku ověřování výsledků evoluční optimalizace. Následně byl nahrazen turnajovým výběrem, který dával lepší výsledky, oproti dramaticky snížené rozmanitosti populace.

Tento typ výběru může být negativně ovlivněn několika vysoce nadprůměrně ohodnocenými jedinci, kteří se nejčastěji účastní reprodukce. Je to případ přehnaného selektivního tlaku, při kterém dochází k rychlé degeneraci populace, dramaticky se sníží rozmanitost populace.

Turnajový výběr (Tournament Selection)

Turnajový výběr [58] je zřejmě nejčastěji využívaný způsob selekce v aplikacích evolučních algoritmů. Je vybráno více jednotlivců, kteří proti sobě nastoupí v turnaji. Přežívají jen ti nejsilnější, kteří postupují do dalšího kola turnaje. Tedy přežívají jedinci s nejlepším ohodnocením účelové funkce. Ale jedinci jdoucí do soubojů jsou vybíráni náhodně, což zaručuje diverzitu populace. Do soubojů může být vybrán každý jedinec několikrát, tedy za předpokladu, že se jedná o různé souboje.

Seříznutý výběr (Truncation Selection)

Využívá se pro populace s velkým počtem jedinců, kdy jedinci s nejlepším ohodnocením jsou vybráni ke křížení. Podíl populace, který bude vybrán za rodiče, určuje tzv. řezací práh (truncation treshold) a pohybuje se kolem 10% - 50%. Odříznutí jedinci nejsou připuštěni ke křížení [58].

Křížení, rekombinace (Crossover, Recombination)

Křížením [57] jedinců získáme nové jedince v populaci. K výběru jedinců pro křížení může být použito algoritmu výběru (selection). Z částí již existujících řešení se snaží skládat nové řešení. Použitím křížení je snižováno riziko uváznutí v lokálních extrémech.

Mutace (Mutation)

Mutace [57] je v populaci primárním zdrojem variability a vytváří nové jedince (nová řešení problému). Četnost použití mutací se nastavuje podle povahy algoritmu. Rozsah mutací je možno zvětšovat u stagnujících generací, aby bylo opuštěno případné lokální maximum. Četnost mutací je možné zvětšovat po určitém počtu stagnujících generací. Většina mutací má na jedince destruktivní vliv více než pozitivní, protože snižuje jejich vhodnost.

Pokud jsou jedinci definováni binárně, tak mutace znamená změnu alely z hodnoty 0 na 1 a naopak. Při binárním kódování gen nabývá hodnotu 1 nebo 0.

Účelová funkce (Fitness Function)

Za účelem výběru je nutné ohodnotit jedince v populaci. Ohodnocení nebo také ocenění je počítáno hodnotící funkcí [57]. Volba účelové funkce záleží na konkrétním problému, který chceme pomocí genetického algoritmu řešit. Pro křížení jsou vybráni jedinci s lepšími vlastnostmi.

Vyskytují se případy, kdy je příliš velká populace, poté se může používat systém rozdělení populace na dvě pod populace a obě se porovnávají mezi sebou. Podle vzájemného porovnávání jednotlivci dostanou ohodnocení.

Koncová podmínka

Koncová podmínka [57] se používá pro omezení počtu iterací. Maximální počet iterací by měl být vždy zadán, aby předešel uváznutí genetických algoritmů během řešení nebo předešel problému nepříjemně pomalé konvergence. V dalším případě podmínka hlídá, zda neproběhlo několik posledních iterací bez vylepšení dosavadního výsledku. Koncovou podmínkou může být také nepřesáhnutí časového limitu.

Základní pojmy evolučních algoritmů

Genotyp (Genotype)	obecná genetická informace
Fenotyp (Phenotype)	zobrazení genetické informace do konkrétního stavu
Jedinec (Individual)	nositel genetické informace
Chromozóm (Genome)	genetická informace ve tvaru řetězce (mohou to být čísla, znaky nebo jejich kombinace)
Gen (Gene)	konkrétní místo v chromozómu
Alela (Allele)	konkrétní symbol v chromozómu, kterého nabývá gen
Rodič (Parent)	jedinec vstupující do rekombinace, vybraný metodou selekce
Potomek (Offspring)	jedinec, jenž je výsledkem rekombinace dvou nebo více rodičů
Ohodnocení (Fitness)	ohodnocení síly jedince v generaci, odvozuje jeho přežití
Křížení (Crossover)	konstrukce nových jedinců (potomků) dle vybraných jedinců generace (rodičů)
Mutace (Mutation)	změna alel (y) genu (ú) v chromozómu
Rekombinace (Recombination)	sled křížení a mutace
Selekce (Selection)	výběr jedinců, kteří přežívají v generaci nebo jsou určeny pro křížení
Generace (Generation)	skupina jedinců, na kterou je aplikována teorie o vývoji druhů
Populace (Population)	obecné vyjádření pro sled generací
Migrace (Migration)	předávání jedinců mezi různými populacemi
Doplnění, Znovuvložení (Reinsertion)	přechod jedince ze starší generace do nové, bez procesu křížení a mutace
Schéma (Scheme)	prototyp chromozómu, mající některé určené geny

Tab. 1: Základní pojmy evolučních algoritmů [58].

4.3.2 Optimalizace hejnem částic

Intelligence roje, intelligence hejna nebo rojová intelligence, kolektivní intelligence (Swarm intelligence) [59] je intelligence, založená na chování živočichů jednoho druhu v hejnu (nebo roji, skupině). Pojem poprvé použili Jing Wang a Gerardo Beni v roce 1989 v kontextu celulárních robotických systémů.

Například hejno ryb, mravenců vzájemně spolupracuje a spoluprací dokáže překonat překážku nebo odrazit nebezpečí. Jednají stejně podle vlastností prostředí, ve kterém se pohybují, dokáží rychle zareagovat, například rozestoupit se, pokud na ně zaútočí predátor, aniž by do sebe naráželi. Jako celek působí jednolitě.

Hlavními výhodami algoritmů intelligence hejna je jejich robustnost, flexibilita a jednoduchá implementovatelnost. Nevýhodou je, že není vždy garantován optimální výsledek díky práci s náhodnými prvky. Jedním z řešení je vícenásobné spuštění algoritmu [60].

Algoritmus optimalizace hejnem částic

Tato výpočetní technika má původ v umělé inteligenci a sociální psychologii. Byla vyvinuta roku 1995 elektroinženýrem Russelem Eberhartem a sociálním psychologem Jamesem Kennedym [37] a inspirována se sociálním chováním rybích a ptačích hejn [61]. Algoritmus PSO pochází z pokusných modelů, vytvářených biologem Frankem Heppnerem roku 1990. Ten zkoušel vytvářet algoritmy inspirované chováním hejn.

Chování optimalizace znamená, že hejno se přeskupuje, rozptyluje a pokud dojde k ohrožení predátorem, tak mění směr a to bez řízení nějakým vůdcem. Další fakt je, že přes velké množství částic nedochází k jejich srážkám. Reynolds a Heppner došli k závěru, že si částice udržují optimální vzdálenost od ostatních částic hejna. Dále zjistili, že každá částice má tendenci směřovat do středu hejna a tendenci udržovat podobnou rychlost jako jeho sousedé. Základní myšlenka pro vznik PSO algoritmu vyšla z objevu socio-biologa Wilsona, který při popisování shlukování ryb zjistil, že jednotlivci jsou schopni těžit ze zkušeností a objevů všech ostatních členů hejna a že tato výhoda při hledání potravy převažuje i soutěživost mezi jednotlivci. Z toho lze usuzovat, že shlukování jedinců do hejn vzniklo jako evoluční pokrok.

Po svém představení v roce 1995 se stala optimalizace postupně oblíbená a začala nacházet své využití v mnoha odvětvích. Po původní optimalizaci vzniklo více než sto nových verzí, nejčastěji po zveřejnění standardní verze standardní optimalizace hejnem částic (SPSO) 2006.

Principem této optimalizace je prohledávání prostoru hejny částic, hledajících globální řešení problému v prostoru. Každá částice je reprezentovaná bodem se svou polohou a rychlostí, mění se dynamicky v průběhu výpočtu. Každá částice v bodě je ohodnocena účelovou funkcí. Následný pohyb částice je na základě nejlepší pozice částice a zároveň dosud nejlepší pozice hejna. Při každé iteraci všechny částice podstoupí změnu pozice a změnu rychlosti. Nakonec se hejno jako celek pravděpodobně přesune blízko k optimu účelové funkce.

Oproti genetickým algoritmům, kdy špatná řešení vypadávají v následující iteraci, tak PSO zachovává v následujících iteracích diferenciaci částic. Částice nevypadávají, ale řídí se nejlepšími nalezenými řešeními. Výhodami jsou i relativně snazší implementace algoritmu [62] a nižší množství parametrů.

Základní pojmy optimalizace PSO

Částice	je tím míněn jedinec z hejna. Každá částice se pohybuje individuálně, ale jejich řídicí princip je u všech stejný. Pohybují se směrem ke svému pBest a zároveň směrem ke gBest a během toho ověřují pozici, na níž se nachází
Počet částic	udává množství částic, použitých pro optimalizaci, počet je v průběhu celé optimalizace konstantní, beze změny. Čím více jedinců, tím hustěji bude prostor prohledán, ale časová náročnost bude vyšší. Typicky stačí kolem 20 - 50 jedinců. Některé obtížné problémy ale vyžadují i mnohem větší počet (200 a více)
Pozice	je reprezentována polem hodnot, kde každá jednotlivá hodnota definuje pozici. U řešeného problému o n parametrech budeme mít pozici reprezentovanou polem o délce n a optimální řešení
Rychlost	tento operátor jako výsledek mění pozici částice v prostoru. Vstupem je tedy částice a na základě změny pozice částice v prostoru hledáme extrém
pBest	je to pozice s nejlepším nalezeným řešením dané částice. Tedy každá částice má své vlastní pBest a tuto hodnotu při každé iteraci porovnává s aktuálním pBest a případně ji zaměňuje
gBest	je pozice s nejlepším řešením nalezeným celým hejnem. Při každé iteraci porovnává každá aktuální částice ohodnocenou pozici s gBest a případně ji zamění. Všechny částice jsou k této hodnotě přitahovány
Globální sousedství	každou částici ovlivňuje nejlepší soused z úplně celé populace, tedy gBest
Ukončující podmínka	může to být třeba rozdíl ocenění nejlepší a nejhorší částice, případně konvergence posledních částic k relativně stejnému výsledku, nebo nejčastěji počet iterací

Tab. 2: Základní pojmy optimalizace hejname částic [41].

4.3.3 Optimalizace PSO s vahou setrvačnosti

Kvůli neustále zvyšující se rychlosti jedince, kvůli kterému může částice divergovat (blížit se nekonečnu), byla zavedena setrvačnost, která lépe kontroluje rozsah hledání a snižuje důležitost V_{max} . Jde o mírnou úpravu rovnice rychlosti.

Rozdílem následujícího algoritmu je přidání parametru váhy setrvačnosti ω [63], [64], [65].

Výpočet nové rychlosti aktuální částice:

$$v_i(t+1) = \omega \cdot v_i(t) + c_1 \cdot r_1(t) \cdot (pBest_i(t) - x_i(t)) + c_2 \cdot r_2(t) \cdot (gBest(t) - x_i(t)) \quad (1)$$

i = označení jednotlivé částice

t = označení jednotlivé iterace

$v_i(t+1)$ = rychlost částice i v následující iteraci $t+1$

$v_i(t)$ = rychlost částice i v aktuální iteraci t

$x_i(t)$ = pozice částice i v aktuální iteraci t

$pBest_i(t)$ = nejlepší nalezená pozice částice i v aktuální iteraci t

$gBest(t)$ = nejlepší nalezená pozice hejnem v aktuální iteraci t

ω = parametr váhy setrvačnosti

c_1, c_2 = akcelerační koeficienty (učící faktory)

$r_1(t), r_2(t)$ = náhodná čísla v intervalu 0 až 1 v aktuální iteraci t

První část rovnice $\omega v_i(t)$ popisuje setrvačnost pohybu částice, tedy snahu pokračovat stále stejným směrem. Často bývá násobena nějakou vahou, která omezuje vliv předcházející rychlosti na rychlost aktuální.

Rozdílem vůči původní PSO je přidání parametru utlumení ω . Pokud je $\omega = 1$, pohyb částice je plně ovlivněn jejím předchozím pohybem, takže částice může pokračovat ve stejném směru. Na druhou stranu, pokud $0 \leq \omega < 1$, takový vliv je snížen, což znamená, že částice jde spíše do jiných oblastí v doméně vyhledávání.

Bylo zjištěno, že nejlepší výkony PSO vykazuje při lineárním snižování hodnoty ω . Nejprve je prostor prohledáván po velkých skocích a postupně se kroky zmenšují. Hodnota ω je nastavena na hodnotu ω_{max} a snižuje se k hodnotě ω_{min} , většinou $\omega_{max} = 0.9$ a $\omega_{min} = 0.4$. Nevýhodou ovšem je, že pokud jednou ω klesá, hejno ztrácí schopnost hledat nové oblasti.

Výpočet parametru utlumení rychlosti v závislosti na iteraci:

$$\omega = \omega_{max} - \frac{(\omega_{max} - \omega_{min}) \cdot actIter}{numIter} \quad (2)$$

ω = parametr utlumení rychlosti

$actIter$ = aktuální iterace

$numIter$ = celkový počet iterací

Výpočet parametru utlumení rychlosti v závislosti na náhodném čísle:

Inspirací pro následující rovnici byl omezující faktor χ , jehož optimální hodnota je 0.7289. Průměrná hodnota je v následující rovnici 0.75, tedy velmi podobná.

$$\omega = 0.5 + \frac{r}{2} \quad (3)$$

r = náhodné číslo v mezích $[0, 1]$

Výpočet akceleračního koeficientu v závislosti na parametru utlumení rychlosti:

Obecně s růstem akceleračního koeficientu c roste nestabilita (hejno uteče z vymezené oblasti). Při zvýšení parametru ω je vhodné pro stabilitu systému snížit akcelerační koeficient c . Větší stability je dosaženo při menším parametru stability ω a většího akceleračního koeficientu c .

$$c < \frac{2(1 - 2 \cdot \omega + \omega^2)}{1 + \omega} \quad (4)$$

akcelerační koeficient $c = c_1 + c_2$

parametr utlumení rychlosti = $\omega < 1$ a $\omega \neq 0$

Výpočet nové pozice aktuální částice (je stejný jako v původním PSO).

4.3.4 Optimalizace PSO varianta Many Optimizing Liaisons

Byl přizpůsoben zjednodušený typ optimalizace hejnem částic PSO s vahou setrvačnosti [63], na Many Optimizing Liaisons (MOL) [66], který eliminuje nejlepší nalezenou pozici částice pBest tím, že akcelerační koeficient c_1 je roven 0. Ve vnitřní smyčce jsou pak částice náhodně vybrány. Výsledkem je jednodušší implementace fungující stejně dobře jako původní verze optimalizace hejnem částic s vahou setrvačnosti, ne-li lépe. Sociálním psychologem Jamesem Kennedym [67] byla podobná optimalizace nazvána jenom sociální optimalizace hejna částic.

Výpočet nové rychlosti aktuální částice:

$$v_i = \omega \cdot v_i + c_2 \cdot r_2 \cdot (gBest - x_i) \quad (5)$$

i = označení jednotlivé částice

v_i = rychlost částice i

x_i = pozice částice i

$gBest$ = nejlepší nalezená pozice hejnem

ω = parametr váhy setrvačnosti

c_2 = akcelerační koeficient (učicí faktor)

r_2 = náhodné číslo v intervalu 0 až 1

Výpočet nové pozice aktuální částice (je stejný jako v původním PSO).

Pseudokód Many Optimizing Liaisons:

```
1. nejlepší ocenění hejna = 0
2. gBest = []
3. for každou částici i do
4.     vygenerování náhodné rychlosti  $v_i$  a pozice  $x_i$  částice i
5. end for
6. t = 0
7. while t < maximum iterací do
8.     for každou částici i do
9.         nové ocenění částice i = 0
10.        while nové ocenění částice i = 0 do
11.            if nejlepší ocenění hejna <> 0 then
12.                aktualizace rychlosti  $v_i$  částice i:  $v_i = \omega \cdot v_i + c_2 \cdot r_2(gBest - x_i)$ 
13.                aktualizace pozice  $x_i$  částice i:  $x_i = x_i + v_i$ 
14.            end if
15.            nové ocenění částice i je vypočítáno účelovou funkcí
16.            if nové ocenění částice i = 0 then
17.                vygenerování náhodné rychlosti  $v_i$  a pozice  $x_i$  částice i
18.            end if
19.        end while
20.        if nové ocenění částice i < nejlepší ocenění hejna or nejlepší ocenění hejna = 0 then
21.            nejlepší ocenění hejna je nastaveno na nové ocenění částice i
22.            gBest je nastaveno na pozici  $x_i$  částice i
23.        end if
24.    end for
25.    t = t + 1
26. end while
```

5 Parametrizační metody

Tvary aerodynamických profilů jsou reprezentovány jednotlivými souřadnicemi. Při optimalizaci a práci s mnoha profily je nutné vyjádřit tvar profilu pomocí několika parametrů. Reprezentace profilů za pomoci několika málo parametrů se nazývá parametrizace. Výhodou parametrizace je charakteristika jednotlivých profilů v řádech několika hodnot vůči charakterizaci pomocí souřadnic v řádech několika stovek hodnot. Tato část výběru je klíčová pro implementaci optimalizačních algoritmů.

5.1 Přehled parametrizací

5.1.1 Hicks-Henneho funkce (Hicks-Henne Function)

Hicks-Henneho funkce [1], [68], [69], [70] používá základní definici profilu plus lineární kombinaci sady n základních funkcí definovaných mezi 0 a 1 pro určení konečného tvaru profilu. Hicks a Henne navrhli sadu možných základních funkcí na bázi sinusové funkce. Parametry t_1 a t_2 mohou být měněny podle aplikace. Koeficienty α_i jsou návrhové proměnné.

$$0 \leq x \leq 1$$

$$\phi_i(x) = \left[\sin \left(\pi \cdot x^{\frac{\log(0,5)}{\log(t_1)}} \right) \right]^{t_2} \quad (6)$$

$$z = z^{base} + \sum_{i=0}^n \alpha_i \cdot \phi_i(x) \quad (7)$$

5.1.2 Radiální bázové funkce (Radial Basis Functions)

Radiální bázová funkce [71] je matematická funkce přiřazující skutečnou hodnotu každému bodu v prostoru. Je běžně používaná v různých oblastech [72], jako je matematika, informatika. Další z běžných aplikací RBF je parametrizace geometrie profilu při návrhu profilu. RBF lze použít k interpolaci a aproximaci souřadnic profilu umožňující hladké a souvislé znázornění tvaru profilu.

Obecná forma RBF je:

$$\Phi(x) = \Phi(\|x - c\|) \quad (8)$$

Φ je radiální bázová funkce, x je bod v prostoru, c je střed RBF a $\|x - c\|$ představuje euklidovskou vzdálenost mezi x a c .

5.1.3 Volná tvarová deformace (Free-Form Deformation)

Volná tvarová deformace je technika sloužící k úpravě tvaru křivek a ploch počítačovým modelováním [73]. Nejvíce se používá v oblasti počítačové grafiky v tzv. free-form modelování, kde různými nástroji můžeme volně upravovat tvar [74], [75]. FFD je možné použít pro vytváření aerodynamických tvarů v leteckém průmyslu (viz obr. 9) [76].

Princip deformace spočívá v přemístování řídicích bodů [77]. Výhodou volných deformací je jejich všestrannost. FFD lze aplikovat na libovolný geometrický objekt zadaný parametrickou rovnicí či při implicitním vyjádření. FFD má důležitou vlastnost a to, že deformací parametrického povrchu získáme opět parametrický povrch.

Je použita Newtonova metoda mapování objektu do prostoru, kdy původní objekt ve 3D je definován B-spline objemem, charakterizovaným rovnicí 1. s řídicími body $Q_{i,j,k}$ [78], [79], [80].

$$\vec{P}(u, v, w) = \sum_{i=0}^{n_u} N_{i,m_u}(u), N_{j,m_v}(v), N_{k,m_w}(w) Q_{i,j,k} \quad (9)$$

s B-spline bázovými funkcemi $N_{i,m_u}(u)$, $N_{j,m_v}(v)$, $N_{k,m_w}(w)$ stupně m_u , m_v , m_w definovanými uzlovými vektory $N_i : \{0, \dots, 0, u_{m_u+1}, \dots, u_{n_u}, 1, \dots, 1\}$, $N_j : \{0, \dots, 0, v_{m_v+1}, \dots, v_{n_v}, 1, \dots, 1\}$ a $N_k : \{0, \dots, 0, w_{m_w+1}, \dots, w_{n_w}, 1, \dots, 1\}$. Bázové funkce jsou počítány:

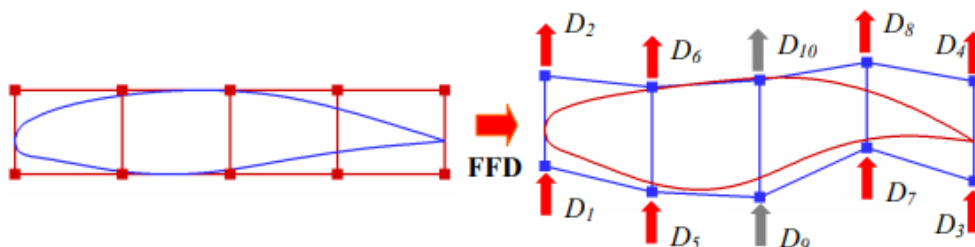
$$N_{i,1u} = \begin{cases} 0, & u_i \leq u < u_{i+1} \\ 1, & \text{jinak} \end{cases} \quad (10)$$

a

$$N_{i,m_u}(u) = \frac{u - u_i}{u_{i+m_u-1} - u_i} \cdot N_{i,m_u-1}(u) + \frac{u_{i+m_u} + u}{u_{i+m_u} - u_{i+1} - u_i} \cdot N_{i+1,m_u-1}(u) \quad (11)$$

Po deformaci je původní B-spline objem změněn na nový B-spline objem s novými řídicími body $Q_{i,j,k}^{new}$.

$$\vec{P}^{new}(u_0, v_0, w_0) = \sum_{i=0}^{n_u} N_{i,m_u}(u_0), N_{j,m_v}(v_0), N_{k,m_w}(w_0) Q_{i,j,k}^{new} \quad (12)$$



Obr. 9: Aplikace volné tvarové deformace ve 2D [81].

5.1.4 Parametrizace CST (Class Shape Function)

CST parametrizace je mimo jiné používána na profily, křídla a jiné části letounů [1]. K výpočtu používá tvarových tříd a tvarových funkcím, kdy $C_{N2}^{N1}(\psi)$ reprezentuje tvarovou třídu a $S(\psi)$ reprezentuje tvarovou funkci.

$$\zeta(\psi) = C_{N2}^{N1}(\psi) \cdot S(\psi) + \psi \cdot \zeta_T \quad (13)$$

kde $\zeta = z/c$, $\psi = x/c$ a $\zeta_T = \Delta\zeta_{TE}/c$. ζ_T je tloušťka odtokové hrany.

Pro třídu profilů NACA má tvarová třída parametry $N1 = 0.5$ a $N2 = 1$:

$$C(\psi) = \psi^{N1} \cdot (1 - \psi)^{N2} \rightarrow C(\psi) = \sqrt{\psi} \cdot (1 - \psi) \quad (14)$$

Tvarová třída je definována:

$$S(\psi) = \sum_{i=0}^p \bar{P}_i \cdot B_{i,p}(\psi) \quad (15)$$

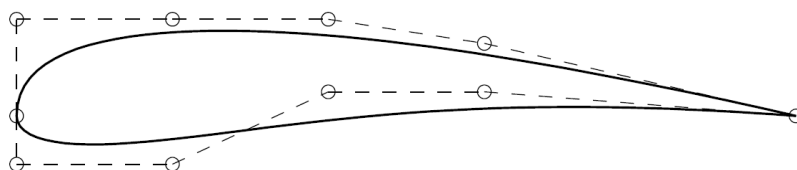
Bernsteinův polynom:

$$B_{i,p} = \binom{p}{i} \cdot \psi^i \cdot (1 - \psi)^{p-i} \quad (16)$$

p je stupeň Bernsteinova polynomu, \bar{P}_i koeficienty Bernsteinova polynomu, za pomoci kterých lze modifikovat tvar profilu.

5.1.5 Bezier parametrizace (Bezier Parameterization)

Parametrizace Bezierovou křivkou [82], [83] je dána jejími řídicími body, které jsou fyzickými body v rovině. První a poslední řídicí bod jsou počáteční a koncový bod samotné křivky. Nicméně ostatní řídicí body nemusí být na křivce, i když určují tvar křivky [84]. Bézierova parametrizace profilu předpokládá i řídicí body, které jsou nepřímo určovány základní aerodynamikou. Profil se skládá ze dvou křivek, jmenovitě zakřivení profilu a rozložení tloušťky profilu. Abychom získali horní a dolní povrch profilu, přidáme a odečteme rozdělení tloušťky od střední křivky.



Obr. 10: Ukázka Bezierovy křivky s řídicími body [85].

Bezierova křivka řádu n je popsána jako:

$$z(t) = \sum_{i=1}^n B_i^n(t) \cdot P_i \quad (17)$$

Směšené funkce Bezierových křivek jsou Bernsteinovy polynomy. Označuje je B_i^n , kde

$$B_i^n = \binom{n}{i} \cdot (1-t)^{n-1} \cdot t^i \quad (18)$$

kde i je – tý Bernsteinův polynom

n je stupeň Bezierovy křivky pro $n - 1$ řídicích bodů pro $t \in \langle 0, 1 \rangle$

poznamenejme, že $\binom{n}{i}$ je binomický koeficient, který se rovná

$$\binom{n}{i} = \frac{n!}{i!(n-i)!} \quad (19)$$

kde P_i je množina řídicích bodů

Příklad Bezierovy křivky se stupněm $n = 3$

$$B_0^3 = (1-t)^3$$

$$B_1^3 = 3t \cdot (1-t)^2$$

$$B_2^3 = 3t^2 \cdot (1-t)$$

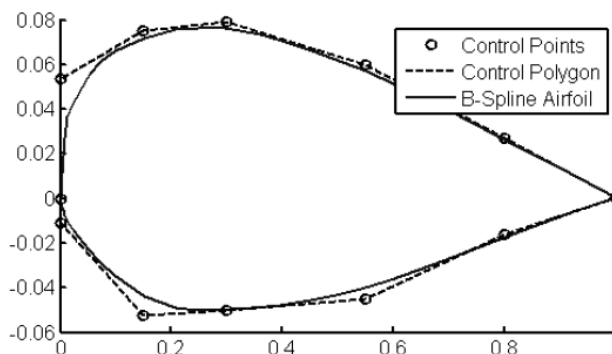
$$B_3^3 = t^3$$

5.1.6 B-spline parametrizace (B-spline Parameterization)

Stejně jak parametrizace Bezierovou křivkou je B-spline dána řídicími body, které jsou fyzickými body v rovině. V zásadě lze tvrdit, že B-spline [83] je zobecněním Bezierovy křivky a je definován jako:

$$z(t) = \sum_{i=0}^n P_i \cdot N_{i,p}(t) \quad (20)$$

kde p je řád polynomu, P_i je množina řídicích bodů, $N_{i,p}$ je základní funkce B-spline.



Obr. 11: B-Spline křivka profilu definovaného 13 řídicími body [83].

5.1.7 PARSEC parametrizace (PARSEC Parameterization)

Tato parametrizace [82] je velmi používaná a vysoce účinná metoda parametrizace profilu. Používá dvanáct základních parametrů pro úplné definování tvaru profilu. Parametrizace [86] je vyjádřena horním (UP) a dolním (LO) povrchem profilu.

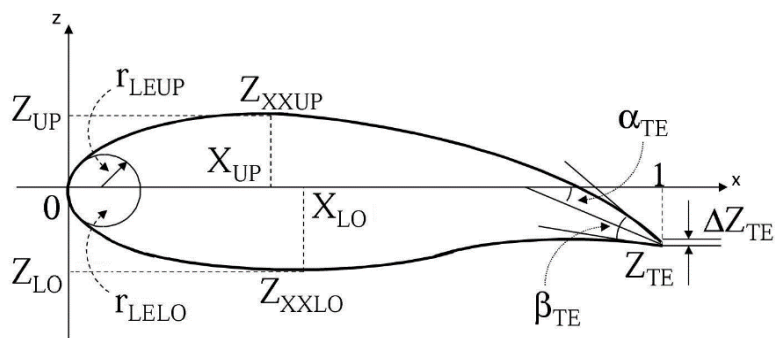
$$Z^{UP} = \sum_{i=1}^6 a_i^{UP} \cdot X^{e_i^{UP}} \quad (21)$$

$$Z^{LO} = \sum_{i=1}^6 a_i^{LO} \cdot X^{e_i^{LO}} \quad (22)$$

Koeficienty e_i jsou zde jako konstanty, oproti a_i tomu jako proměnné. Změnou koeficientů e_i lze získat různé variace parametrizace. Jsou-li splněny základní předpoklady na exponenty e_i , lze pak od 12 návrhových proměnných přejít k analytickému vyjádření řešením malé lineární soustavy 12 rovnic o 12 neznámých, nebo dvou soustav 6 rovnic o 6 neznámých. Původní PARSEC parametrizace Helmuta Sobieczkeho používá následujících koeficientů e_i .

$$e_i^{UP} = e_i^{LO} = i - \frac{1}{2} \quad (23)$$

Význam jednotlivých proměnných parametrizace je zřejmý z obr. 12.



Obr. 12: Návrhové proměnné parametrizace [82].

r_{LEUP}, r_{LELO} jsou horní a dolní poloměry křivosti náběžné hrany

$X_{UP}, X_{LO}, Z_{UP}, Z_{LO}$ jsou maxima obou povrchů

Z_{XXUP}, Z_{XXLO} jsou reciproční křivosti (druhá derivace) maxima obou povrchů

r_{LEUP}, r_{LELO} jsou horní a dolní poloměry křivosti náběžné hrany

α_{TE} je úhel odtokové hrany

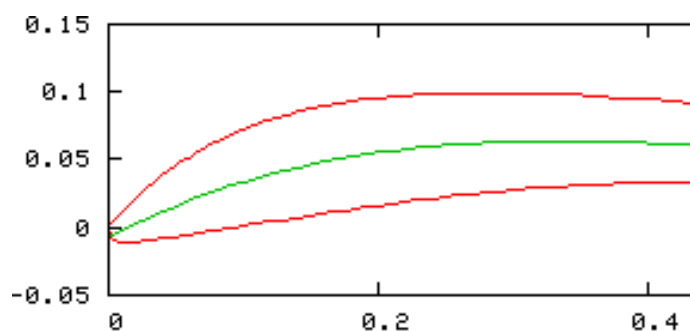
β_{TE} je sklon odtokové hrany

Z_{TE} je zetová souřadnice

ΔZ_{TE} je tloušťka

Problémem parametrizace je nevelký počet stupňů volnosti, neumožňující velkou geometrickou různorodost. Ovšem pokud je nutnost udržovat počet proměnných co nejnižší, je naopak výhodou.

Geometrický problém je omezující podmínka, že vrchol náběžné hrany je v počátku souřadnic. Příkladem, že vrchol náběžné hrany není v počátku souřadnic, je profil 2032crev.

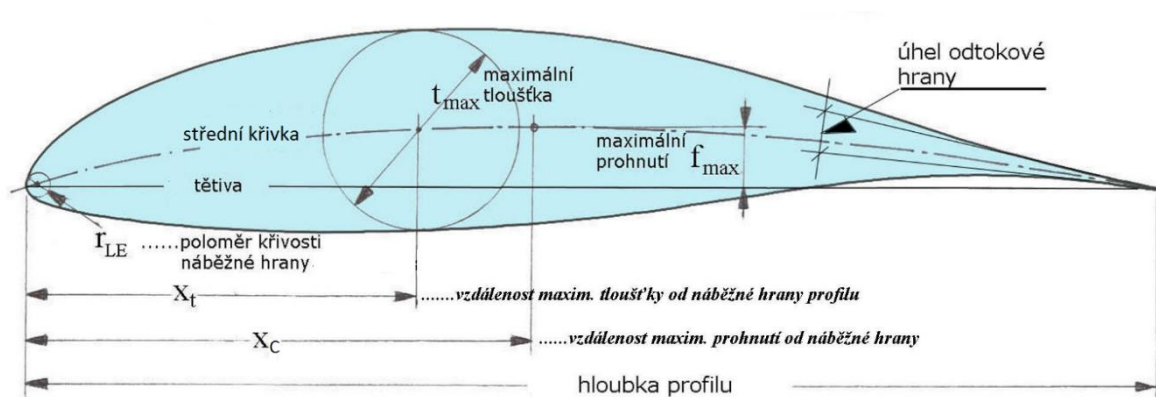


Obr. 13: Profil 2032crev.

5.1.8 Bezier-PARSEC parametrizace (Bezier-PARSEC Parameterization)

Tato metoda je kombinací parametrizačních technik Bezier a PARSEC. Sloučením byly získány výhody, poskytované oběma zmíněnými metodami. Bezier-PARSEC parametrizace [82] využívá PARSEC proměnné jako parametry, které jsou použity k vytvoření čtyř jednotlivých Bezier křivek. Tyto čtyři křivky tvoří střední křivku náběžné a odtokové hrany a rozložení tloušťky náběžné a odtokové hrany. Bezier-PARSEC parametrizace používá druhého stupně spojitosti pro spojení náběžné a odtokové hrany.

Bezier-PARSEC parametry



Obr. 14: Základní geometrie profilu.

r_{le} je poloměr křivosti náběžné hrany

α_{te} je úhel střední křivky odtokové hrany

β_{te} je sklon odtokové hrany

z_{te} je svislé posunutí odtokové hrany

γ_{le} je úhel směru náběžné hrany

x_c je vzdálenost maximálního prohnutí od náběžné hrany

y_c je maximální prohnutí

κ_c je zakřivení střední křivky

x_t je vzdálenost maximální tloušťky od náběžné hrany

y_t je maximální tloušťka

κ_t je zakřivení tloušťky

dz_{te} je polovina tloušťky odtokové hrany

$b_0, b_2, b_8, b_{15}, b_{17}$ jsou Bezier parametry

5.1.9 Bezier-PARSEC 3333 parametrizace (BP 3333 Parameterization)

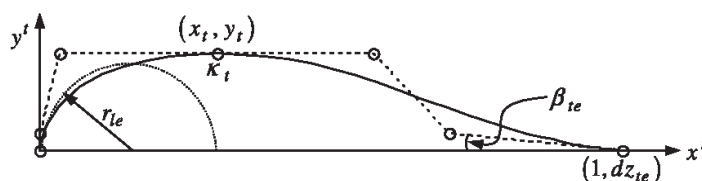
Všechny čtyři křivky jsou polynomy třetího stupně. Parametrizace Bezier-PARSEC 3333 [82] závisí na 12 aerodynamických parametrech. V Bezier-PARSEC 3333 nejsou žádné volné body Bezier křivek. Bezier křivka třetího stupně je parametricky dána:

$$x(u) = x_0 \cdot (1 - u)^3 + 3 \cdot x_1 \cdot u \cdot (1 - u)^2 + 3 \cdot x_2 \cdot u^2 \cdot (1 - u) + x_3 \cdot u^3 \quad (24)$$

a

$$y(u) = y_0 \cdot (1 - u)^3 + 3 \cdot y_1 \cdot u \cdot (1 - u)^2 + 3 \cdot y_2 \cdot u^2 \cdot (1 - u) + y_3 \cdot u^3 \quad (25)$$

Bezier-PARSEC 3333 tloušťka



Obr. 15: Bezier-PARSEC 3333 tloušťka.

Řídící body náběžné hrany

$$x_0 = 0$$

$$y_0 = 0$$

$$x_1 = 0$$

$$y_1 = \frac{3 \cdot \kappa_t \cdot (x_t - b_9)^2}{2} + y_t$$

$$x_2 = b_9$$

$$y_2 = y_t$$

$$x_3 = x_t$$

$$y_3 = y_t$$

Parametr b_9 je kořenem

$$\frac{27 \cdot \kappa_t^2 \cdot b_9^4}{4} - 27 \cdot \kappa_t^2 \cdot x_t \cdot b_9^3 + \left(9 \cdot \kappa_t \cdot y_t + \frac{81 \cdot \kappa_t^2 \cdot x_t^2}{2} \right) \cdot b_9^2 +$$

$$(2 \cdot r_{ie} - 18 \cdot \kappa_t \cdot x_t \cdot y_t - 27 \cdot \kappa_t^2 \cdot x_t^3) \cdot b_9 + \left(3 \cdot y_t^2 + 9 \cdot \kappa_t \cdot x_t^2 \cdot y_t + \frac{27 \cdot \kappa_t^2 \cdot x_t^4}{4} \right) = 0$$

(26)

musí být v daných mezích

$$\max \left(0, x_t - \sqrt{-\frac{2 \cdot y_t}{3 \cdot \kappa_t}} \right) < b_9 < x_t \quad (27)$$

Řídící body odtokové hrany

$$x_0 = x_t$$

$$y_0 = y_t$$

$$x_1 = 2 \cdot x_t - b_9$$

$$y_1 = y_t$$

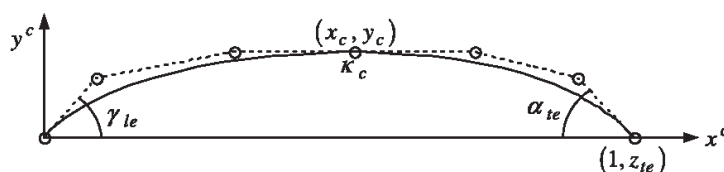
$$x_2 = 1 + \left[dz_{te} - \left(\frac{3 \cdot \kappa_t \cdot (x_t - b_9)^2}{2} + y_t \right) \right] \cdot \cot(\beta_{te})$$

$$y_2 = \frac{3 \cdot \kappa_t \cdot (x_t - b_9)^2}{2} + y_t$$

$$x_3 = 1$$

$$y_3 = dz_{te}$$

Bezier-PARSEC 3333 střední křivka



Obr. 16: Bezier-PARSEC 3333 střední křivka.

Řídící body náběžné hrany

$$x_0 = 0$$

$$y_0 = 0$$

$$x_1 = b_1 \cdot \cot(\gamma_{te})$$

$$y_1 = b_1$$

$$x_2 = x_c - \sqrt{\frac{2 \cdot (b_1 - y_c)}{3 \cdot \kappa_c}}$$

$$y_2 = y_c$$

$$y_3 = y_c$$

$$x_3 = x_c$$

Parametr b_1 je vypočítán z

$$b_1 = \frac{[16 + 3 \cdot \kappa_c \cdot (\cot(\gamma_{le}) + \cot(\alpha_{te})) \cdot (1 + z_{te} \cdot \cot(\alpha_{te}))]}{[3 \cdot \kappa_c \cdot (\cot(\gamma_{le}) + \cot(\alpha_{te}))]} \pm$$

$$4 \cdot \sqrt{16 + 6 \cdot \kappa_c \cdot (\cot(\gamma_{le}) + \cot(\alpha_{te})) \cdot (1 - y_c \cdot (\cot(\gamma_{le}) + \cot(\alpha_{te})) + z_{te} \cdot \cot(\alpha_{te}))}$$
(28)

a musí být v následujících mezích

$$0 < b_1 < y_c$$

Řídící body odtokové hrany

$$\begin{aligned} x_0 &= x_c & y_0 &= y_c \\ x_1 &= x_c + \sqrt{\frac{2 \cdot (b_1 - y_c)}{3 \cdot \kappa_c}} & y_1 &= y_c \\ x_2 &= 1 + (z_{te} - b_1) \cdot \cot(\alpha_{te}) & y_2 &= b_1 \\ x_3 &= 1 & y_3 &= z_{te} \end{aligned}$$

5.1.10 Bezier-PARSEC 3434 parametrizace (BP 3434 Parameterization)

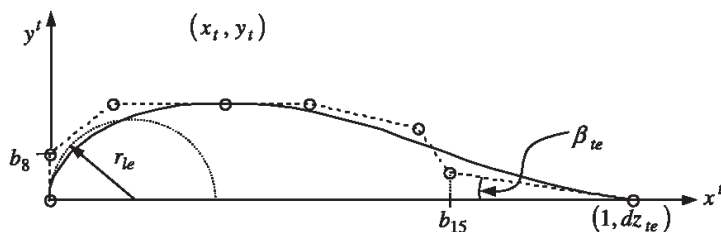
Obě křivky náběžné hrany jsou polynomy třetího stupně, zatímco obě křivky odtokové hrany jsou polynomy čtvrtého stupně. Parametrizace Bezier-PARSEC 3434 [82] závisí na deseti aerodynamických parametrech plus pěti parametrech Bézierových křivek. Bezier křivka čtvrtého stupně je parametricky dána:

$$x(u) = x_0 \cdot (1 - u)^4 + 4 \cdot x_1 \cdot u \cdot (1 - u)^3 + 6 \cdot x_2 \cdot u^2 \cdot (1 - u)^2 + 4 \cdot x_3 \cdot u^3 \cdot (1 - u) + x_4 \cdot u^4$$
(29)

a

$$y(u) = y_0 \cdot (1 - u)^4 + 4 \cdot y_1 \cdot u \cdot (1 - u)^3 + 6 \cdot y_2 \cdot u^2 \cdot (1 - u)^2 + 4 \cdot y_3 \cdot u^3 \cdot (1 - u) + y_4 \cdot u^4$$
(30)

Bezier-PARSEC 3434 tloušťka



Obr. 17: Bezier-PARSEC 3434 tloušťka.

Řídící body náběžné hrany

$$x_0 = 0$$

$$y_0 = 0$$

$$x_1 = 0$$

$$y_1 = b_8$$

$$x_2 = -\frac{3 \cdot b_8^2}{2 \cdot r_{le}}$$

$$y_2 = y_t$$

$$y_3 = y_t$$

$$x_3 = x_t$$

Parametr b_8 podléhá následujícímu omezení

$$0 < b_8 < \min \left(y_t, \sqrt{-\frac{2 \cdot r_{le} \cdot x_t}{3}} \right) \quad (31)$$

Řídící body odtokové hrany

$$x_0 = x_t$$

$$y_0 = y_t$$

$$x_1 = \frac{\left(7 \cdot x_t + \frac{9 \cdot b_8^2}{2 \cdot r_{le}} \right)}{4}$$

$$y_1 = y_t$$

$$y_2 = \frac{(y_t + b_8)}{2}$$

$$x_2 = 3 \cdot x_t + \frac{15 \cdot b_8^2}{4 \cdot r_{le}}$$

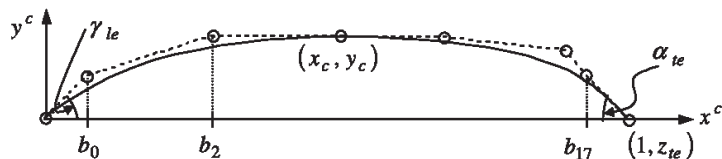
$$y_3 = dz_{te} + (1 - b_{15}) \cdot \tan(\beta_{te})$$

$$x_3 = b_{15}$$

$$y_4 = dz_{te}$$

$$x_4 = 1$$

Bezier-PARSEC 3434 střední křivka



Obr. 18: Bezier-PARSEC 3434 střední křivka.

Řídící body náběžné hrany

$$x_0 = 0$$

$$y_0 = 0$$

$$x_1 = b_0$$

$$y_1 = b_0 \cdot \tan(\gamma_{le})$$

$$x_2 = b_2$$

$$y_2 = y_c$$

$$x_3 = x_c$$

$$y_3 = y_c$$

Řídící body odtokové hrany

$$x_0 = x_c$$

$$y_0 = y_c$$

$$x_1 = \frac{(3 \cdot x_c - y_c \cdot \cot(\gamma_{le}))}{2}$$

$$y_1 = y_c$$

$$x_2 = \frac{(-8 \cdot y_c \cdot \cot(\gamma_{le}) + 13 \cdot x_c)}{6}$$

$$y_2 = \frac{5 \cdot y_c}{6}$$

$$x_3 = b_{17}$$

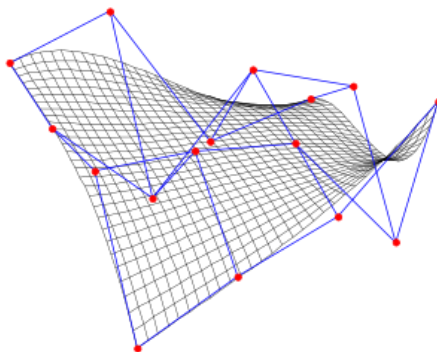
$$y_3 = z_{te} - (1 - b_{17}) \cdot \tan(\alpha_{te})$$

$$x_4 = 1$$

$$y_4 = z_{te}$$

5.1.11 Bézierovy povrchy (Bezier Surfaces)

Pro úplnost jsou uvedeny Bézierovy povrchy [69], které jsou druhem matematického spline používaného v počítačové grafice, CAD a modelování konečných prvků. Stejně jako u Bézierovy křivky je Bézier povrch definován sadou řídicích bodů [70]. Povrch obecně neprochází řídicími body; spíše je "napnutý" směrem k nim. Jsou vizuálně intuitivní a pro mnoho aplikací matematicky výhodné.



Obr. 19: Bézierovy povrchy.

5.2 Volba parametrizací

V konečném důsledku je volba parametrizace často rovnováhou mezi několika faktory:

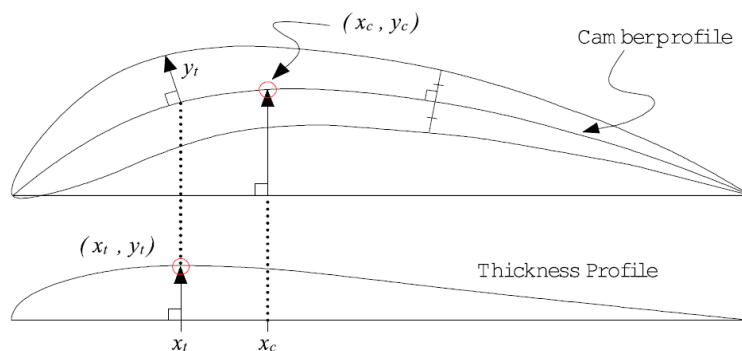
- **přesnost** - některé parametrizace jsou bližší ve srovnání s experimentálními a reálnými výsledky, zatímco jiné charakterizují zjednodušeně,
- **složitost** - díky ní lze detailněji definovat geometrii profilu, většinou za cenu výpočetní náročnosti,
- **výpočetní efektivita** - jednodušší parametrizace mohou být méně výpočetně náročné, ale jsou většinou méně přesné,
- **výběr podle požadavků řešení konkrétního problému**, často podle možnosti aplikování a implementace.

5.2.1 Bezier-PARSEC 3434 parametrizace

Jedná se o jednu z nejnovějších parametrizací, kdy prvotně tvar profilu vychází z Bézierových křivek definovaných řídicími body, dále pak z PARSEC parametrizace charakterizované dvanácti parametry.

Kombinace Bézierových křivek s metodou PARSEC zajišťuje, že tvar profilu je zobrazen hladce bez jakýchkoli ostrých diskontinuit nebo náhlých změn. Flexibilita této parametrizace umožňuje návrhářům prozkoumat širokou škálu tvarů profilu s různými charakteristikami.

Tato parametrizace generuje geometrii profilu na základě parametrů střední křivky a tloušťky (viz obr. 20). Pokud jsou však parametry libovolné, pak výsledkem parametrů mohou být profily s nepoužitelnými tvary. Do Bezier-PARSEC parametrizace tak bylo přidáno hlídání rozpětí jednotlivých parametrů tak, aby výsledkem každé parametrizace byl reálný tvar profilu.



Obr. 20: Tvar profilu ze střední křivky a tloušťky [87]

5.2.2 B-spline parametrizace

Pokud jsou k dispozici řídicí body B-spline křivek, pro horní a dolní části profilu, lze manipulací těchto bodů měnit geometrii profilu. Změnou tvaru se mění i aerodynamické vlastnosti, avšak většinou nové tvary profilu mají podobný tvar vztaženo k výchozímu profilu, tedy i podobné aerodynamické vlastnosti.

S vyšším počtem řídicích bodů je možné definovat složitější tvary a zvyšují různost výsledného tvaru. Tvarově jednodušší profil NACA 2412 má 14 řídicích bodů, ale při složitějších geometriích profilu není výjimkou i přes 30 řídicích bodů.

6 Statistické testy

Hladina významnosti (alfa, α)

Je důležité brát v potaz, že hladina významnosti 0.05 je konvenční úroveň statistické významnosti a může být zvolena jiná hladina v závislosti na konkrétním výzkumném kontextu a požadavcích [88], [89].

P-hodnota (p-value)

Je číselná hodnota používaná při statistickém testování hypotéz. Reprezentuje pravděpodobnost, že výsledek (např. rozdíl pozorovaný mezi dvěma skupinami) mohl vzniknout náhodou [90], [91], [92].

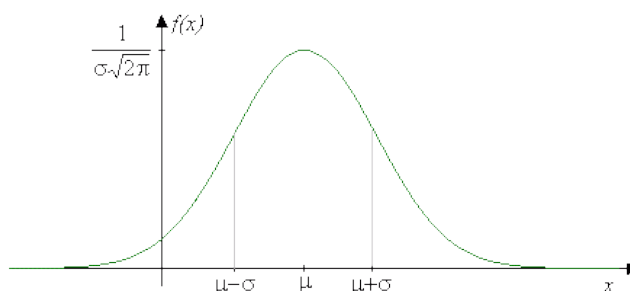
Signifikance rozdílů výsledků (Statistická významnost výsledků)

Srovnání p-hodnot s hladinami významnosti	Výsledek z hlediska significance
p-hodnota > 0.10	výsledek není signifikantně rozdílný
p-hodnota \leq 0.10	výsledek je okrajově signifikantně rozdílný
p-hodnota \leq 0.05	výsledek je signifikantně rozdílný
p-hodnota \leq 0.01	výsledek je velmi signifikantně rozdílný

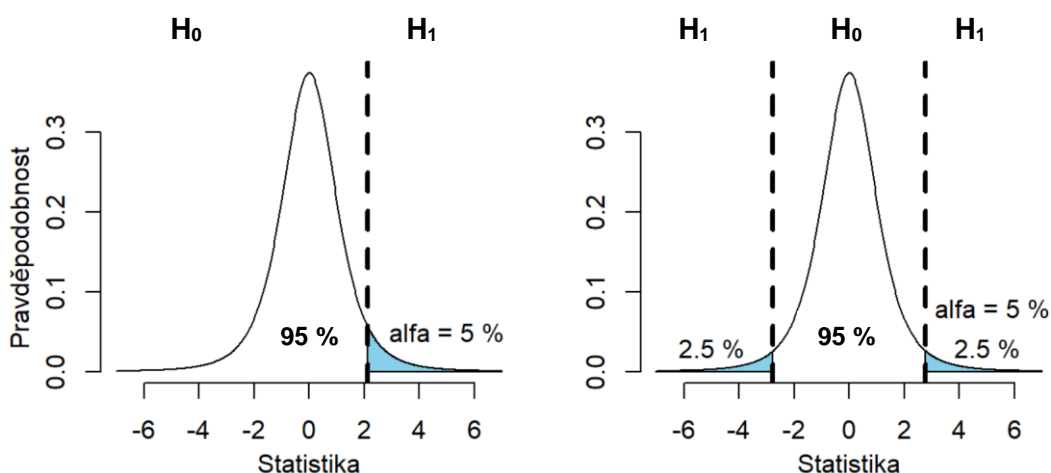
Tab. 3. Signifikance rozdílů výsledků podle srovnání p-hodnot s hladinou významnosti [93], [94], [95], [96].

Normální rozdělení (Bell curve)

Normální rozdělení někdy též Gaussovo rozdělení [97] je definováno dvěma parametry. Parametr střední hodnoty μ určuje maximum křivky. Parametr směrodatné odchylky σ určuje, jak jsou po obou stranách od hodnoty μ vzdáleny inflexní body (křivka se mění z konkávní na konvexní a naopak), tedy jak je křivka roztažena do šířky [98].



Obr. 21: Gaussova křivka tj. graf hustoty pravděpodobnosti [98].



Obr. 22: Normální rozdělení při jednostranném (pravostranném) a oboustranném testu [99].

Nulová hypotéza (H_0)

Pokud je p -hodnota větší než hladina významnosti 0.05, lze zpravidla přijmout nulovou hypotézu a prohlásit, že neexistují dostatečné důkazy pro to, aby byla zpochybněna. To znamená, že v zkoumaném statistickém testu nebyly nalezeny statisticky významné rozdíly nebo efekty mezi daty nebo skupinami [95], [96].

Nulová hypotéza je používána k testování, zda jsou pozorované statistiky významné nebo zda může být výsledek náhodný. Nulová hypotéza je klíčovým prvkem při interpretaci výsledků statistických testů [100].

Alternativní hypotéza (H_1)

Alternativní hypotéza navrhuje existenci nějakého efektu, rozdílu nebo změny v datech k ověření statistickými testy. Pokud platí alternativní hypotéza, tak pozorovaný efekt nevznikl jen náhodou [89].

Pokud je zamítnuta nulová hypotéza, můžeme tvrdit, že existuje statisticky významný efekt nebo rozdíl, tj. platí alternativní hypotéza.

Chyby α a β při testování hypotéz

Chyba 1. druhu α nastává, pokud je zamítnuta nulová hypotéza H_0 i když platí. Chyba 2. druhu β je při nesprávném přijetí nulové hypotézy H_0 i když neplatí [101].

6.1 Výčet testů

Statistický test

Statistický test je matematická procedura umožňující zhodnotit, zda existují statisticky významné rozdíly, efekty nebo souvislosti mezi zkoumanými daty. Používá se k rozhodnutí o tom, zda zamítnout nulovou hypotézu H_0 na základě dat [101].

(Ne)parametrické statistické testy

U parametrických statistických testů data pocházejí z určitého známého rozdělení, obvykle z normálního rozdělení. To znamená, že parametrické testy jsou citlivé na tvar rozdělení dat [102]. Neparametrické statistické testy nevyžadují, aby data pocházela z konkrétního rozdělení a jsou vhodné pro data, která nejsou normálně distribuována, tedy vychází z neznámého rozdělení.

Parametrické	Neparametrické
Testy pro jeden výběr - přiléhavosti (Kolmogorov-Smirnovův) - o hodnotě průměru, rozptylu a korelačním koeficientu	Testy pro jeden výběr - přiléhavosti (χ^2 - test) - náhodnosti založených na iteracích
Testy pro dva výběry - o rozdílu dvou průměrů, rozptylů a korelačních koeficientů	Testy pro dva výběry - závislé (Bowkerův, McNamarův a Wilcoxonův test) - nezávislé (Mann-Whitneův, mediánový test)
Analýza rozptylu (ANOVA) - jednoduchá - pro opakovaná měření	Testy závislosti dvou znaků (χ^2 - test nezávislosti dichotomických znaků, Fisherův test a test nezávislosti Spearmanova koeficientu pořadové korelace)
	Testy pro více výběrů - nezávislých (χ^2 - test homogenity, Kruskal – Wallisův test) - závislých (Cochranův test, Friedmannův test)
	Testy závislosti více znaků - Kendallův koeficient konkordance

Obr. 23: Výčet statistických testů podle aplikace [103].

6.1.1 Mediánový test

Nechť Y_{i1}, \dots, Y_{in_i} jsou veličiny, kdy r náhodných na sobě nezávislých výběrů má rozsahy n_1, \dots, n_r , přičemž všechny tyto výběry pochází ze spojitého rozložení s distribuční funkcí F_i . Pokud platí nulová hypotéza H_0 , tak všechny distribuční funkce jsou navzájem totožné, tedy $F_1 = \dots = F_r$. Obecně platí $i = 1, \dots, r$. Oproti tomu za platnosti alternativní hypotézy H_1 jsou distribuční funkce navzájem rozdílné [104], [105].

Velichiny Y_{ij} vytvoří dohromady sdružený výběr o rozsahu $N = n_1 + \dots + n_r$. Poté jsou uspořádány do rostoucí posloupnosti, ve které mají veličiny Y_{11}, \dots, Y_{1n_1} pořadí R_1, \dots, R_{n_1} , veličiny Y_{21}, \dots, Y_{2n_2} pořadí $R_{n_1+1}, \dots, R_{n_1+n_2}$ a tak dále.

Mediánový test je založen na testové statistice:

$$Q_M = 4 \cdot \sum_{i=1}^r \frac{1}{n_i} \cdot \left(A_i - \frac{n_i}{2} \right)^2 = 4 \cdot \sum_{i=1}^r \frac{A_i^2}{n_i} - N \quad (32)$$

kde

$$A_i = \sum_{j \in g_i} \frac{1}{2} \cdot \left\{ \text{sign} \cdot \left[R_j - \frac{1}{2} \cdot (N + 1) \right] + 1 \right\} \quad (33)$$

udává počet veličin i -tého výběru, které jsou větší než medián sdruženého výběru. Pokud je N liché a medián patří do i -tého výběru pak se k tomuto počtu připočte $\frac{1}{2}$.

Nulová hypotéza H_0 platí, pokud má Q_M asymptoticky rozložení $\chi^2 \cdot (r - 1)$ a pokud rostou rozsahy nade vše meze. Tedy pro $Q_M \geq \chi_{1-\alpha}^2 \cdot (r - 1)$ je zamítnuta nulová hypotéza H_0 na hladině významnosti, která se s rostoucím n_i blíží číslu α .

6.1.2 Wilcoxonův párový test

Wilcoxonův párový test (Wilcoxonův test se závislými výběry) [106], je neparametrická alternativa k párovému t-testu.

Nechť X_1, \dots, X_n a Y_1, \dots, Y_n jsou dva závislé výběry. Pokud platí nulová hypotéza H_0 , tak oba soubory mají shodné rozdělení. Oproti tomu za platnosti alternativní hypotézy H_1 oba soubory nemají shodné rozdělení [107].

Nechť $(X_1, Y_1), \dots, (X_n, Y_n)$ jsou závislé páry. Párový test je převeden na jednovýběrový nahrazením každé dvojice (X_i, Y_i) jejich diferencí $X_i - Y_i$, $i = 1, \dots, n$. Pro jednovýběrový test jsou hodnoty z rozdílů definovány X_1, \dots, X_n . Tyto jsou seřazeny do rostoucí posloupnosti podle absolutní hodnoty, tedy $|X_{(1)}| < \dots < |X_{(n)}|$. Nechť je R_i je pořadí X_i při tomto uspořádání [108].

Součty kladných a záporných pořadí jsou:

$$S^+ = \sum_{X_i > 0} R_i \quad \text{a} \quad S^- = \sum_{X_i < 0} R_i \quad , \text{přičemž platí} \quad S^+ + S^- = \frac{n \cdot (n + 1)}{2} \quad (34)$$

Pro malá $n \leq 25$ lze kritickou hodnotu pro statistiku vypočítat:

$$T = \min(S^+, S^-) \quad (35)$$

Pokud je hodnota T menší než stejná jako kritická tabulková hodnota, pak lze nulovou hypotézu H_0 zamítnout na zvolené hladině významnosti [109].

Pro větší $n > 25$

$$U = \frac{T - \frac{n \cdot (n + 1)}{4}}{\sqrt{\frac{n \cdot (n + 1) \cdot (2 \cdot n + 1)}{24}}} \quad (36)$$

Stejně tak jestliže je hodnota U menší než stejná jako kritická tabulková hodnota, pak lze nulovou hypotézu H_0 také zamítnout na zvolené hladině významnosti [109].

6.1.3 Mann-Whitneyův test

Mann-Whitneyův test (dvouvýběrový Wilcoxonův test) [105], [110], [111] se řadí k nejsilnějším neparametrickým testům a je používán jako obdoba dvouvýběrového t-testu.

Nechť X_1, \dots, X_m je náhodný výběr ze základního souboru se spojitou distribuční funkcí F . Nechť Y_1, \dots, Y_n je druhý nezávislý náhodný výběr ze základního souboru se spojitou distribuční funkcí G (rozsahy m a n jsou obecně různé). Pokud platí nulová hypotéza H_0 , tak distribuční funkce F je totožná s distribuční funkcí G , tedy $F = G$. Oproti tomu za platnosti alternativní hypotézy H_1 jsou obě distribuční funkce rozdílné, tedy $F \neq G$.

Všechny hodnoty obou výběrů ($n + m$ hodnot) jsou seřazeny vzestupně podle velikosti, poté je sečteno pořadí všech hodnot X_1, \dots, X_m a je označeno jako S_1 . Obdobně je sečteno pořadí všech hodnot Y_1, \dots, Y_n a označeno jako S_2 .

Jsou vypočítány testovací statistiky U_1 a U_2 na základě následujících vzorců:

$$U_1 = m \cdot n + \frac{m \cdot (m + 1)}{2} - S_1 \quad a \quad U_2 = m \cdot n + \frac{n \cdot (n + 1)}{2} - S_2 \quad (37)$$

Přičemž platí následující vztahy:

$$U_1 + U_2 = m \cdot n \quad (38)$$

Pokud menší z hodnot U_1 a U_2 je menší nebo rovné tabulkové kritické hodnotě (pro dané rozsahy m a n k dané hladině významnosti α) pak je nulová hypotéza H_0 zamítnuta a na dané hladině významnosti α je potvrzena alternativní hypotéza.

6.1.4 Kruskal-Wallisův test

Kruskal-Wallisův test [105], [110], [112] je neparametrická alternativa k analýze rozptylu. Jedná se o zevšeobecnění Mann-Whitneyova testu, kdy Kruskal-Wallisův test se používá pro dva a více nezávislých výběrů.

Nechť Y_{i1}, \dots, Y_{in_i} jsou veličiny, kdy r náhodných na sobě nezávislých výběrů má rozsahy n_1, \dots, n_r , přičemž i -tý výběr pochází ze spojitého rozložení s distribuční funkcí F_i . Pokud platí nulová hypotéza H_0 , tak všechny distribuční funkce jsou navzájem totožné, tedy $F_1 = \dots = F_r$. Obecně platí $i = 1, \dots, r$. Oproti tomu za platnosti alternativní hypotézy H_1 jsou distribuční funkce navzájem rozdílné.

Velichiny Y_{ij} vytvoří dohromady sdružený výběr o rozsahu $N = n_1 + \dots + n_r$. Poté jsou uspořádány do rostoucí posloupnosti, ve které mají veličiny Y_{11}, \dots, Y_{1n_1} pořadí R_1, \dots, R_{n_1} , veličiny Y_{21}, \dots, Y_{2n_2} pořadí $R_{n_1+1}, \dots, R_{n_1+n_2}$ a tak dále. Součet pořadí těchto hodnot patřících do i -tého výběru je označen jako T_i . Tedy musí platit $T_1 + \dots + T_r = N \cdot (N + 1)/2$.

Testová statistika má tvar:

$$Q_K = \frac{12}{N \cdot (N + 1)} \cdot \sum_{i=1}^r \frac{T_i^2}{n_i} - 3 \cdot (N + 1) \quad (39)$$

Nulová hypotéza H_0 platí, pokud má statistika Q_K asymptoticky rozložení $\chi^2 \cdot (r - 1)$, rostou-li rozsahy výběrů nade všechny meze. Tedy pro $Q_K \geq \chi_{1-\alpha}^2 \cdot (r - 1)$ je zamítnuta nulová hypotéza H_0 na hladině významnosti, která se s rostoucím n_i blíží číslu α .

Je-li zamítnuta nulová hypotéza H_0 , je nutné se rozhodnout, které dvojice výběru se významně liší. K tomu se používají metody mnohonásobného porovnávání.

Pokud výběry mají stejný rozsah n lze použít Neményiho metodu: Je-li $|T_l - T_k| \geq$ tabulková kritická hodnota, je zamítnuta na hladině významnosti α hypotéza, že l -tý a k -tý výběr pocházejí ze stejného rozložení.

Pokud výběry mají různé rozsahy, je použita obecná metoda mnohonásobného porovnávání: Je-li

$$|T_l - T_k| \geq \sqrt{\frac{1}{12} \cdot \left(\frac{1}{n_l} + \frac{1}{n_k}\right) \cdot N \cdot (N + 1) \cdot h_\alpha} \quad (40)$$

je zamítnuta na hladině významnosti α hypotéza, že l -tý a k -tý výběr pocházejí ze stejného rozložení. Při větších rozsazích výběrů je možné nahradit kritickou hodnotu h_α kvantilem $\chi_{1-\alpha}(r - 1)$.

6.1.5 Friedmanův test

Friedmannův test [104], [105] se používá namísto Wilcoxonova testu v případě více než dvou závislých výběrů [113].

Nechť Y_{ij} jsou nezávislé náhodné veličiny se spojitými distribučními funkcemi F_{ij} , kdy $i = 1, \dots, r$ a $j = 1, \dots, k$. Celkem $r \cdot k$ je počet pozorování na r objektech s k opakováními. Testuje se nulová hypotéza H_0 , zda jsou distribuční funkce F_{i1}, \dots, F_{ik} totožné.

Test ověřuje, zda úroveň sledovaného znaku závisí nebo nezávisí na změně podmínek. V závislosti na výsledku se určí, zda vyšetření mají na každý blok stejný efekt. Někdy nejde přímo o vyšetření, ale nějaká veličina je na každém sledovaném objektu r opakována v k časových okamžicích.

Pro každé i se zvlášť určí pořadí R_{ij} veličiny Y_{ij} . Jde tedy o určení pořadí každého bloku, mezi veličinami Y_{i1}, \dots, Y_{ik} .

i	j			
	1	2	...	k
1	Y_{11}	Y_{12}	...	Y_{1k}
2	Y_{21}	Y_{22}	...	Y_{2k}
...
r	Y_{r1}	Y_{r2}	...	Y_{rk}

 \Rightarrow

i	J			
	1	2	...	k
1	R_{11}	R_{12}	...	R_{1k}
2	R_{21}	R_{22}	...	R_{2k}
...
r	R_{r1}	R_{r2}	...	R_{rk}

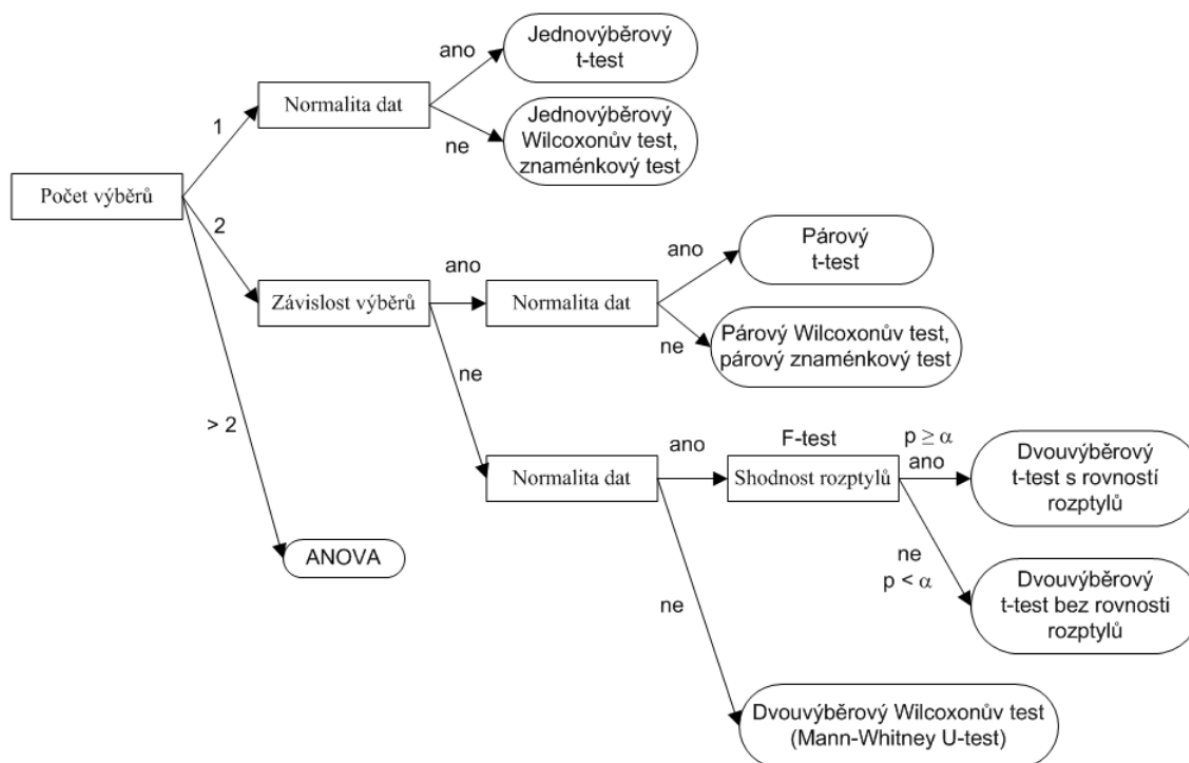
Tab. 4: Nezávislé náhodné veličiny Y_{ij} a jejich pořadí R_{ij} .

Testová statistika má tvar:

$$Q_F = \frac{12}{r \cdot k \cdot (k + 1)} \cdot \sum_{j=1}^k \left(\sum_{i=1}^r R_{ij} \right)^2 - 3 \cdot r \cdot (k + 1) \quad (41)$$

Nulová hypotéza H_0 platí, pokud má statistika Q_F asymptoticky rozložení $\chi^2 \cdot (k - 1)$ a pokud rostou rozsahy nade vše meze. Tedy pro $Q_F \geq \chi_{1-\alpha}^2 \cdot (k - 1)$ je zamítnuta nulová hypotéza na hladině významnosti, která se s rostoucím n_i blíží číslu α .

6.2 Vybrání testu



Obr. 24: Diagram pro výběr správného statistického testu [114].

Kritéria k výběru statistického testu [115] :

- **počet výběrů** - pokud jsou srovnávány dva výběry lze použít jednoduché testy, jako je Studentův t-test nebo Mann-Whitneyův test, pro srovnání tří anebo více výběrů lze vybrat analýzu rozptylu (ANOVA) nebo ekvivalentní neparametrický Kruskal-Wallisův test anebo neparametrický Friedmanův test,
- **závislost výběrů** - nastává pokud jsou páry výběrů závislé. Mohou vyžadovat specifické statistické testy, jako je párový t-test nebo Wilcoxonův párový test,
- **normalita dat** - normální data s normálním rozdělením lze analyzovat parametrickými testy, zatímco náhodně rozdělená data s neznámým rozdělením vyžadují neparametrické testy. Normalitu dat lze vyčíst z pomoci grafů histogramy, Q-Q grafy a statistických testů tj. Shapiro-Wilkův test pro normalitu,
- **rozptyl** - pokud je předpoklad homoskedasticity (shodnost rozptylů) mezi výběry, lze použít parametrické testy. V opačném případě lze zvážit neparametrické testy,
- **statistická síla a množství dat** - množství dat ovlivňuje sílu analýzy, může se stát, že s malým množstvím zkoumaných dat nebude možno odhalit skutečné rozdíly.

6.2.1 Mann-Whitneyův test

Původně byl vybrán pro optimalizace Wilcoxonův párový test a to kvůli stejné počáteční populaci resp. hejna. Tento předpoklad se nakonec ukázal jako irelevantní. Protože páry dvou výběrů navzájem nesouvisí, ale naopak jsou nezávislé, ukázal se po konzultaci s odborníky na statistiku jako nejvhodnější Mann-Whitneyův test.

7 Nástroje pro automatickou konfiguraci

Autokonfiguratory, také známé jako nástroje pro automatickou konfiguraci, jsou softwarové systémy určené k automatické konfiguraci různých nastavení a parametrů aplikací, systémů nebo zařízení. Tyto nástroje jsou schopny vyladit vše co má parametry od algoritmů, simulátorů, databází, optimalizací, kompilátorů.

Autokonfiguratory jednoduše posílají parametry do objektů, z kterých se jim vrací hodnota, na základě které se pak modifikují parametry, tedy provádí automatickou konfiguraci. Automatická konfigurace je tudíž obecná bez nutnosti zásahu do autokonfiguratoru nebo do konfigurovaného objektu. Mění se pouze konfigurace parametrů.

Optimalizace se týká procesu hledání nejlepšího možného řešení nebo nastavení pro daný problém nebo úkol. Cílem optimalizace je maximalizovat nebo minimalizovat nějakou hodnotu nebo funkci reprezentující kvalitu řešení. Optimalizace tedy nemá obecné použití, na řešení konkrétního problému lze použít pouze vhodné optimalizační algoritmy.

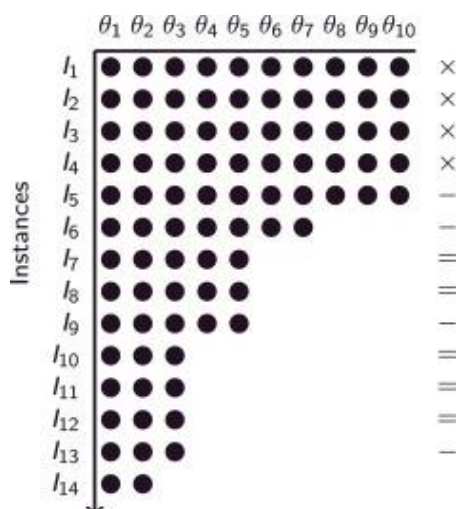
Nejznámější autokonfiguratory jsou IRACE, SPOT, SQUIRREL, ovšem princip autokonfigurace vždy zůstává.

7.1 IRACE (Iterated Racing for Automatic Algorithm Configuration)

IRACE [116], [117], [118] je primárně navržen pro automatické ladění parametrů, tedy pro kritický proces v modelech strojového učení, kde je potřeba najít nejlepší kombinaci parametrů pro konkrétní algoritmus nebo model, aby bylo dosaženo optimálního výkonu na daném úkolu.

Tento nástroj je založen na konceptu závodění, v kterém jsou různé konfigurace (sady parametrů) porovnávány z hlediska jejich výkonu na ověřovací sadě. Konfiguracím, které fungují dobře, je povoleno pokračovat v závodě, zatímco ty méně výkonné jsou eliminovány. Toto závodění pokračuje iterativně, dokud nezůstane pouze nejlepší konfigurace.

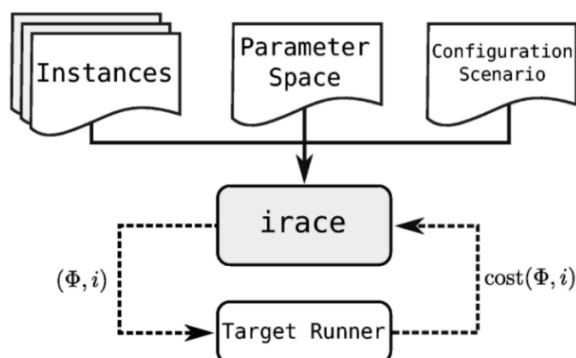
IRACE je zvláště užitečný pro řešení vysokých výpočetních nároků na vyhodnocení různých konfigurací parametrů, protože umožňuje včasné odstranění špatných konfigurací. Iterativním zužováním vyhledávacího prostoru IRACE efektivně najde téměř optimální konfiguraci pro daný model strojového učení.



Obr. 25: Závod pro automatickou konfiguraci algoritmu [119].

Iterované závodění je metoda automatické konfigurace, která se skládá ze tří kroků: 1. vzorkování nových konfigurací podle konkrétní distribuce, 2. výběr nejlepších konfigurací z nově vzorkovaných pomocí závodů a 3. aktualizace vzorkování distribuce, aby se vzorkování posunulo směrem k nejlepším konfiguracím. Tyto tři kroky se opakují, dokud není splněno kritérium ukončení.

Závod začíná konečnou sadou kandidátských konfigurací. V příkladu na obrázku 25 je deset konfigurací parametrů θ . V každém kroku závodu jsou kandidátské konfigurace vyhodnoceny na jedné instanci I_j . Po několika krocích jsou ty kandidátské konfigurace, které vykazují statisticky horší výsledky než alespoň jiná, vyřazeny a závod pokračuje se zbývajících přeživšími konfiguracemi [119].



Obr. 26: Schéma komponent požadovaných IRACE pro konfiguraci algoritmu [120].

Obrázek 26 ukazuje obecné schéma toho, jak IRACE funguje. Obdrží jako vstup parametry, definující prohledávaný prostor, odpovídající parametrům cílového algoritmu k ladění. Vstupem je také sada instancí, pro které musí být parametry vyladěny.

```

# Parameters for Evolutionary Algorithms
#
# name          switch          type    values
populationsize "--popsize "    i      (50, 300)
crossoverprobability "--crossprob "  r      (0.6, 0.9)
tournamentsize "--toursize "    i      (3, 20)
  
```

Obr. 27: Parametry pro aEA.

Je poskytnuta specifická konfigurace parametrů θ a instance i , konkrétně pro evoluční algoritmy budou konfigurovány tři parametry. Je hledána nejlepší velikost populace, pravděpodobnost křížení a velikost turnaje.

```
maxExperiments = 500
debugLevel = 3
digits = 5
trainInstancesFile = "instances.txt"
targetRunner = "target-runner.bat"
```

Obr. 28: Scénář konfigurace pro aEA.

Nejdůležitější částí scénáře je definice targetRunneru, kdy pro Microsoft Windows je nastaven batch soubor na obrázku 29, pro příklad u Linuxu je to bash skript.

```
@echo off
SET "pythonexe=python.exe"
SET "targetrunnerpy=./main.py"
SET parameters=%*
%pythonexe% %targetrunnerpy% %parameters%
```

Obr. 29: Bat soubor target-runner.bat.

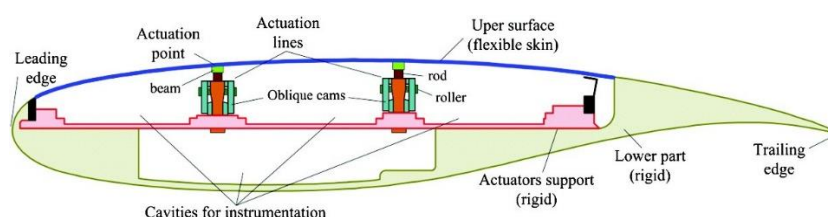
Nástroj IRACE je napsán v R programovacím jazyku speciálně určeném na statistiku. Díky targetRunneru je možná spolupráce s jakýmkoli objekty. V tomto případě IRACE komunikuje s programovým kódem jazyka Python.

8 Optimalizační algoritmy profilu v historické souvislosti

Od počátku letectví až po dnešek se řada autorů a společností pokouší zlepšit letové vlastnosti jak dopravních, tak i vojenských letounů. S přechodem na reaktivní pohon a s ním spojené zvyšování užitečného zatížení, zkracování doby přeletu, snižování spotřeby pohonných hmot a zvyšování bezpečnosti se letové charakteristiky staly velice důležitou, základní složkou návrhů nových letounů.

Optimalizace v projektu ATR-42-4

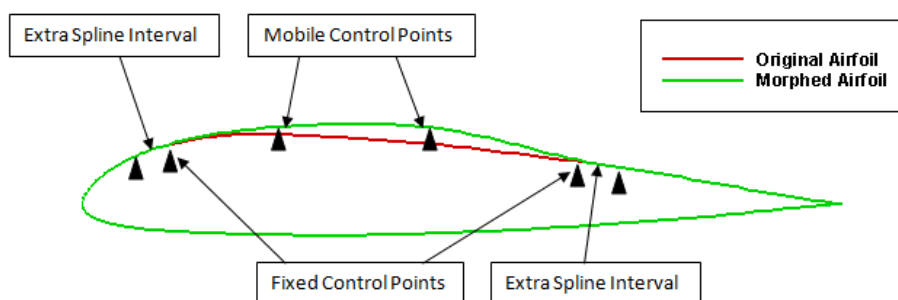
Letoun ATR-42-4 byl jedním z prvních letounů s aplikací optimalizace profilu. Byl použit morphing horní strany profilu [121]. Potah horní strany křídla je elastický, umožňující deformaci [122] dle vystavení dvou akčních prvků, jak je zřejmé z obr. 30.



Obr. 30: Projekt ATR-42-4.

K optimalizaci je použito genetických algoritmů (GA). Každý jedinec v populaci je definován dvěma reálnými hodnotami, představujícími zdvih akčních členů. Tento zdvih nabývá hodnot mezi originálním profilem a zdviženým potahem. K výpočtu jedinců v populaci byla vytvořena účelová funkce, jež zohledňuje minimalizaci odporu profilu.

Aproximace horní deformované části je provedena kubickými splíny. Následně jsou vlastnosti profilu (Mach číslo, Reynoldsové číslo, úhel náběhu a datový soubor profilu (x, y hodnoty)) zavedeny přes bat soubor do programu Xfoil, který poskytuje hodnoty vztlaku a odporu profilu.



Obr. 31: Deformace horní strany profilu.

Výsledkem této úpravy geometrie profilu bylo možné posunout bod přechodu na horní straně povrchu profilu až o 24,81 % a snížit součinitel odporu až o 26,73 %. Redukce odporu vede ke snížení spotřeby paliva o více než 20 %. Také vztlak výrazně vzrostl, zejména v rozmezí malých, pozitivních a negativních úhlů náběhu.

9 Aerodynamické veličiny

9.1 Proudění vzduchu kolem profilu

V 2. polovině 19. století bylo zjištěno anglickým fyzikem Osbornem Reynoldsem, že se tekutina může pohybovat dvěma kvalitativně zcela odlišnými typy proudění, které se nazývají laminární a turbulentní. Hranici mezi oběma druhy proudění udává Reynoldsovo kritické číslo [123].

Laminární proudění

Přísluší nižším rychlostem plynu, při kterých se částice plynu pohybují po paralelních drahách, jednotlivé vrstvy se navzájem nemísí.

Turbulentní proudění

Je naopak spjato s vyšším rychlostem proudění. Setrvačné síly jsou v tomto případě dominantní, což má za důsledek kromě přímočarého pohybu nahodilý pohyb částic ve všech směrech.

Mezní vrstva

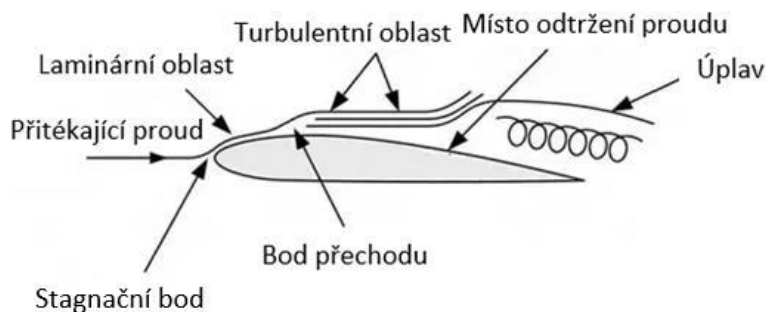
Je oblastí proudění vzduchu v těsné blízkosti tělesa. Rychlost proudění se snižuje v blízkosti povrchu křídla, na povrchu je nulová.

Održení proudů

Při vyšším úhlu náběhu, obvykle obvykle nad $12^\circ - 15^\circ$, již proudnice na horní straně nestačí sledovat zakřivení povrchu profilu a odtrhnou se od něho.

Úplav

Na náběžné hraně dochází ke stlačování plynů, tedy oblast s vyšším tlakem a laminárním prouděním, zatímco na odtokové hraně je oblast s nižším tlakem a turbulentním prouděním tzv. úplav.



Obr. 32: Znázornění obtékání profilu [123].

9.2 Výpočtové aerodynamické veličiny

Reynoldsovo číslo

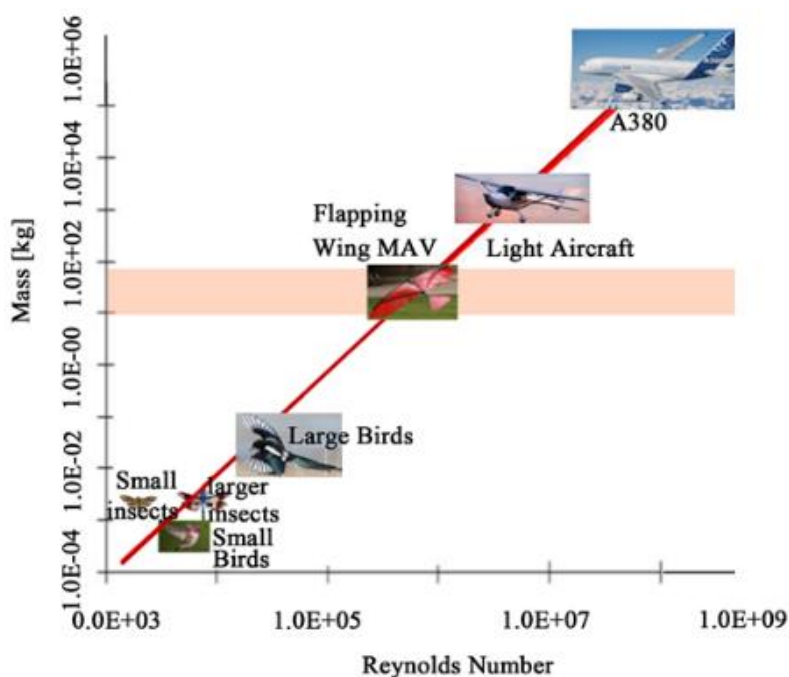
Reynoldsovo číslo je bezrozměrná hodnota, která vyjadřuje poměr setrvačných sil a sil vzniklých vnitřním třením (viskózních sil) a popisuje stupeň laminarity nebo turbulentního toku. Systémy používající stejné Reynoldsovo číslo budou mít stejnou charakteristiku obtékání.

$$Re = \frac{\text{setrvačná síla}}{\text{viskózní síla}} = \frac{\rho \cdot v \cdot l}{\mu} = \frac{v \cdot l}{\nu} [-] \quad (42)$$

ρ = hustota tekutiny [$\text{kg} \cdot \text{m}^{-3}$], μ = dynamická viskozita tekutiny [$\text{Pa} \cdot \text{s}$],

v = rychlost tekutiny [$\text{m} \cdot \text{s}^{-1}$], l = hloubka křídla [m],

ν = kinematická viskozita tekutiny [$\text{m}^2 \cdot \text{s}^{-1}$]



Obr. 33: Vztah mezi hmotnostmi objektů a Reynoldsovým číslem [124].

Machovo číslo

Machovo číslo Ma je bezrozměrná fyzikální veličina používaná v aerodynamice. Udává poměr rychlosti pohybu tělesa určitým prostředím k rychlosti šíření zvuku v témže prostředí. Hodnota Ma menší než 1 vyjadřuje rychlost podzvukovou, je-li Ma větší než 1, jedná se o rychlost nadzvukovou. Ma nese název po významném fyzikovi 19. století Ernstu Machovi, rodákovi z Chrlic.

$$Ma = \frac{v}{c} [] \quad (43)$$

v = rychlost pohybu tělesa [$m \cdot s^{-1}$]

c = rychlost zvuku v daném prostředí [$m \cdot s^{-1}$]

Pro výpočet je také nutné znát nejnižší a nejvyšší úhel náběhu a krok mezi jednotlivými úhly náběhu.

α = úhel náběhu[°]

9.3 Výsledné aerodynamické veličiny

Součinitel vztlaku C_L

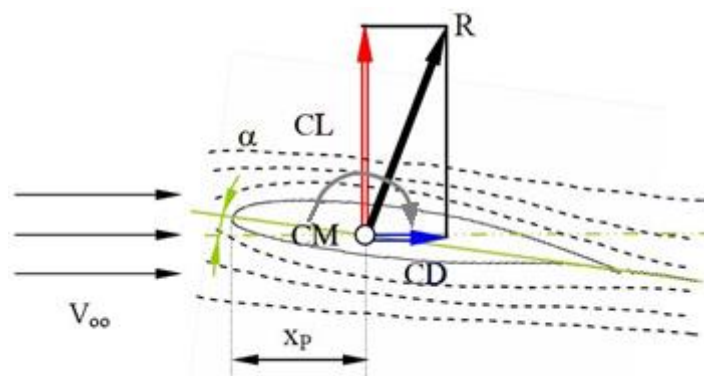
Aerodynamický parametr, vyjadřující vztlakovou sílu, která na tělese působí kolmo na směr přitékajícího proudu vzduchu.

Součinitel odporu C_D

Aerodynamický parametr, vyjadřující odporovou sílu, která na tělese působí ve směru přitékajícího proudu vzduchu.

Součinitel klopivého momentu C_M

Aerodynamický parametr, charakterizující klopivý moment profilu k předem definovanému bodu na jeho těživě. Může být kompenzován opačným momentem ocasních ploch.



Obr. 34: Základní součinitelé profilu.

9.4 Zvolené výpočtové aerodynamické veličiny

Pro porovnání byl zvolen letoun Cessna 172 [125], s profilem NACA 2412. Tento letoun je, do dnešní doby, nejprodávanějším letadlem na světě (dosud prodáno více než 43 000 kusů).

Reynoldsovo číslo

Výpočtem [126] vychází Reynoldsovo číslo pro Cessnu 172 na zhruba $5.0e+06$. Dále pak na základě doporučené výšky letu 2.9 km [127] a rychlosti 225 km/h [128] se dostává taktéž $5.0e+06$.

Výsledné a dále používané ke srovnávání s optimalizovaným profilem je Reynoldsovo číslo $5.0e+06$.

Machovo číslo

$$Ma = \frac{v}{c} = \frac{62.5}{343} = 0.182$$

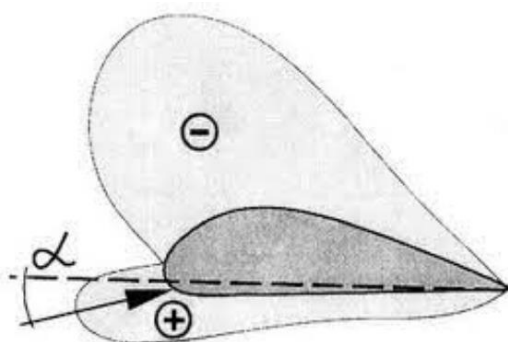
Výsledné Machovo číslo je 0.182.

10 Programy pro výpočet aerodynamických vlastností

Počáteční fáze návrhu každého letounu vytváří potřebu rychlých a spolehlivých metod k aerodynamické analýze zapojené do optimalizačního procesu. Ačkoliv experimentální metody jsou schopny poskytnout všechna požadovaná data, cenově příznivé numerické metody mají zásadní důležitost [129].

Použité přímé metody řeší profily panelovou metodou, ale může být použito složitějších metod, např. metodou konečných objemů nebo metodou konečných prvků [130]. Přímá metoda je založena na spočívání ve vkládání geometrie profilu do programu za účelem získání aerodynamických vlastností.

Naproti tomu inverzní metoda je postupem iteračního hledání takové geometrie, která generuje předepsané rozložení tlaku po hloubce profilu na spodním a dolním povrchu. Tedy je zadáno rozložení součinitele tlaku a program iteračně určí geometrii profilu, která nejlépe odpovídá danému rozložení.



Obr. 35: Rozložení tlaku na povrchu profilu, α - úhel náběhu, (+) přetlak, (-) podtlak [131].

10.1 Seznam výpočetních programů

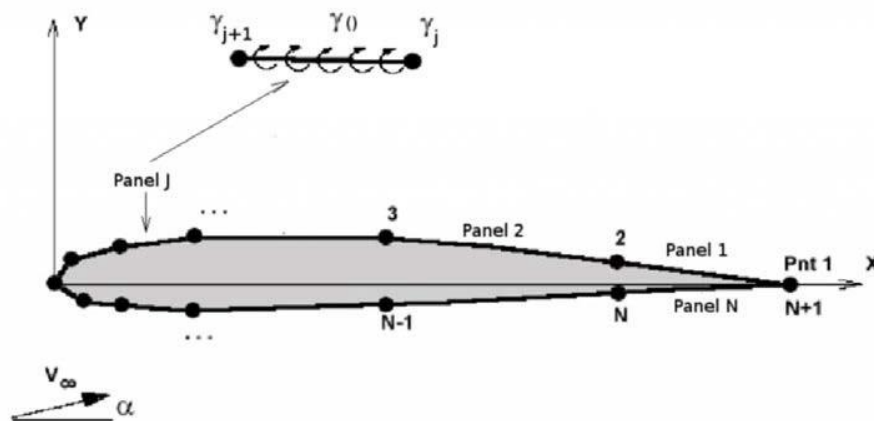
10.1.1 Program Xfoil

Xfoil je program sloužící pro analýzu a výpočet parametrů profilu při podzvukovém proudění. První verzi Xfoilu 1.0 vyvinul v roce 1986 prof. Mark Drela. Program je psaný v programovacím jazyce FORTRAN [123].

Hlavním cílem Xfoilu je provádět numerickou simulaci proudění vzduchu kolem zadaného profilu a vypočítat vztlak, odpor a další aerodynamické charakteristiky profilu na základě zadaných podmínek. Program také umožňuje provádět analýzu vlivu úhlů náběhu, Reynoldsova a Machova čísla na aerodynamické vlastnosti profilu.

Xfoil [132] kombinuje korekci stlačitelnosti s metodami spojenými s viskózními a neviskózními toky. Xfoil používá dvou základních rovnic, udržení mezní vrstvy a panelové metody.

V aerodynamice se od roku 1970 velice často využívá panelové metody, řešící potenciální tok v libovolné, obecné části profilu modelem, založeným na diskretizaci obrysu profilu pomocí singularně navazujících přímých panelů s lineárním rozložením vířivosti mezi koncovými body, jak je zřejmé z obrázku 36 [133].



Obr. 36: Potenciální řešení toku profilu (panelová metoda).

Každý panel je tvořen rovinným segmentem mezi obrysovými body povrchu (j a $j + 1$). Podél panelu se v každém koncovém bodě (γ_j a γ_{j+1}) vytváří vířivé rozložení lineárně se měnící síly (γ). Tyto síly se mění od panelu k panelu, ale hodnoty na koncových bodech na sebe navazují. Vzhledem k tomu, že je známý průřez sekce profilu a jsou známy podmínky volného proudění (v - rychlost, α - úhel náběhu), bude pak přesný model definován rovnicemi hraničních podmínek pro rozložení sil v koncových bodech panelu. Součtem všech sil jednotlivých panelů pak dostaneme celkovou sílu, tj. vztlak.

$$C_L = \sum_{i=1}^N C_{p_i} \cdot \frac{(x_i - x_{i+1})}{c} \quad (44)$$

kde

$$C_{p_i} = \left(1 - \frac{v_i^2}{v_\infty^2} \right) \quad (45)$$

Xfoil poskytuje viskózní a neviskózní analýzy existujících profilů, smíchávání profilů, výpočet odporové poláry s nastaveným Reynoldsovým číslem nebo volitelně Machovým číslem, čtení geometrie profilu a zapisování polár.

QUIT		Exit program
.OPER		Direct operating point(s)
.MDES		Complex mapping design routine
.QDES		Surface speed design routine
.GDES		Geometry design routine
SAVE	f	Write airfoil to labeled coordinate file
PSAV	f	Write airfoil to plain coordinate file
ISAV	f	Write airfoil to ISES coordinate file
MSAV	f	Write airfoil to MSES coordinate file
REVE		Reverse written-airfoil node ordering
LOAD	f	Read buffer airfoil from coordinate file
NACA	i	Set NACA 4,5-digit airfoil and buffer airfoil
INTE		Set buffer airfoil by interpolating two airfoils
NORM		Buffer airfoil normalization toggle
BEND		Display structural properties of current airfoil
PCOP		Set current-airfoil panel nodes directly from buffer airfoil points
PANE		Set current-airfoil panel nodes (140) based on curvature
.PPAR		Show/change paneling
.PPOP		Plotting options
WDEF	f	Write current-settings file
RDEF	f	Reread current-settings file
NAME	s	Specify new airfoil name
NINC		Increment name version number
Z		Zoom (available in all menus)
U		Unzoom

Tab. 5: Výčet základních příkazů a funkcí programu Xfoil.

<cr>		Return to Top Level
!		Redo last ALFA,CLI,CL,ASEQ,CSEQ,VELS
Visc	r	Toggle Inviscid/Viscous mode
.VPAR		Change BL parameter(s)
Re	r	Change Reynolds number
Mach	r	Change Mach number
Type	i	Change type of Mach,Re variation with CL
ITER		Change viscous-solution iteration limit
INIT		Toggle BL initialization flag
Alfa	r	Prescribe alpha
CLI	r	Prescribe inviscid CL
Cl	r	Prescribe CL
ASeq	rrr	Prescribe a sequence of alphas
CSeq	rrr	Prescribe a sequence of CLs
SEQP		Toggle polar/Cp(x) sequence plot display
CINC		Toggle minimum Cp inclusion in polar
HINC		Toggle hinge moment inclusion in polar
Pacc	i	Toggle auto point accumulation to active polar
PGET	f	Read new polar from save file
PWRT	i	Write polar to save file
PSUM		Show summary of stored polars
PLIS	i	List stored polar(s)
PDEL	i	Delete stored polar
PSOR	i	Sort stored polar
PPlo	ii.	Plot stored polar(s)
APlo	ii.	Plot stored airfoil(s) for each polar
ASET	i	Copy stored airfoil into current airfoil
PREM	ir.	Remove point(s) from stored polar
PNAM	i	Change airfoil name of stored polar
PPAX		Change polar plot axis limits
RGET	f	Read new reference polar from file
RDEL	i	Delete stored reference polar
GRID		Toggle Cp vs x grid overaly
CREF		Toggle reference Cp data overlay
FREF		Toggle reference CL,CD.. data display
CPx		Plot Cp vs x
CPV		Plot airfoil with pressure vectors (gee wiz)
.VPlo		BL variable plots

.ANNO		Annotate current plot
HARD		Hardcopy current plot
SIZE	r	Change plot-object size
CPMI	r	Change minimum Cp axis annotation
BL	i	Plot boundary layer velocity profiles
BLC		Plot boundary layer velocity profiles at cursor
BLWT	r	Change velocity profile scale weight
FMOM		Calculate flap hinge moment and forces
FNEW	rr	Set new flap hinge point
VELS	rr	Calculate velocity components at a point
DUMP	f	Output Ue,Dstar,Theta,Cf vs s,x,y to file
CPWR	f	Output x vs Cp to file
CPMN		Report minimum surface Cp
NAME	s	Specify new airfoil name
NINC		Increment name version number

Tab. 6: Výčet základních příkazů a funkcí menu OPER.

NACA2412	
X	Y
1.0000	0.0013
0.9500	0.0114
0.9000	0.0208
0.8000	0.0375
0.7000	0.0518
0.6000	0.0636
0.5000	0.0724
0.4000	0.0780
0.3000	0.0788
0.2500	0.0767
0.2000	0.0726
0.1500	0.0661
0.1000	0.0563
0.0750	0.0496
0.0500	0.0413
0.0250	0.0299
0.0125	0.0215
0.0000	0.0000
0.0125	-0.0165
0.0250	-0.0227
0.0500	-0.0301
0.0750	-0.0346
0.1000	-0.0375
0.1500	-0.0410
0.2000	-0.0423
0.2500	-0.0422
0.3000	-0.0412
0.4000	-0.0380
0.5000	-0.0334
0.6000	-0.0276
0.7000	-0.0214
0.8000	-0.0150
0.9000	-0.0082
0.9500	-0.0048
1.0000	-0.0013

Tab. 7: Soubor dat souřadnic profilu NACA 2412.

Xfoil může být spuštěn z příkazové řádky a postupně interaktivně ovládán přes příkazy a funkce s využitím nabízených menu, nebo je vytvořen bat soubor, který je najednou poslán do programu.

Spuštění Xfoil přes bat soubor bylo využito při návrhu programového vybavení optimalizací.

Xfoil může vypočítat aerodynamické vlastnosti profilu z databáze nebo ze souboru souřadnic. V prvním případě se napíše na první řádek typ profilu, pro ukázkou NACA2412. V druhém případě se název souboru souřadnic se napíše na první řádek za LOAD. Pro ukázkou, pokud se soubor nazývá NACA2412.dat, tak bude na prvním řádku LOAD NACA2412.dat.

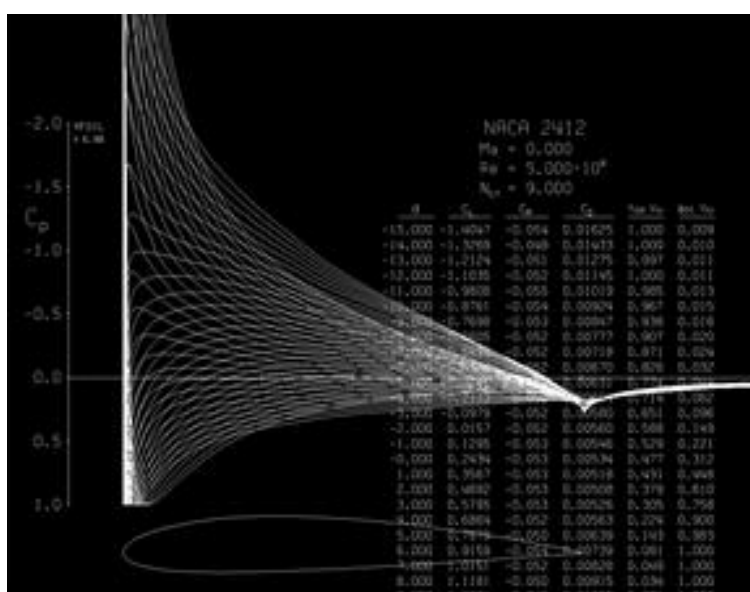
Spustí se přes příkazovou řádku `xfoil.exe < xfoil.bat`

```
LOAD NACA2412.dat ← profil z databáze nebo ze souboru souřadnic
PANE
OPER
VPAR
N 9

VISC 5000000 ← Reynoldssovo číslo
PACC
NACA2412.log ← výstupní soubor poláry
ASEQ -15 15 1.0 ← minimální a maximální úhel náběhu s krokem
QUIT
```

Obr. 37: Ukázkový bat soubor.

Po spuštění příkazu `ASEQ -15 15 1.0` v menu `OPER` vykreslí Xfoil tlakové rozložení, tvar profilu a obsah souboru polary.



Obr. 38: Příklad výstupu z programu Xfoil: součinitelé vztlaku, odporu a klopivého momentu.

Po spuštění příkazu QUIT v menu OPER je uložen soubor poláry NACA2412.log a následně je Xfoil ukončen.

```

XFOIL          Version 6.99

Calculated polar for: NACA2412

1 1 Reynolds number fixed      Mach number fixed

xtrf =  1.000 (top)           1.000 (bottom)
Mach =  0.000      Re =  5.000 e 6      Ncrit =  9.000

  alpha    CL      CD      CDp      CM      Top_Xtr  Bot_Xtr
  -----  -
-15.000  -1.3936  0.01619  0.01215  -0.0558  0.9988  0.0074
-14.000  -1.2867  0.01407  0.00994  -0.0573  0.9959  0.0089
-13.000  -1.1752  0.01235  0.00816  -0.0590  0.9930  0.0108
-12.000  -1.0671  0.01080  0.00658  -0.0595  0.9872  0.0140
-11.000  -0.9451  0.00933  0.00514  -0.0629  0.9835  0.0210
-10.000  -0.8250  0.00836  0.00421  -0.0652  0.9715  0.0277
-9.000   -0.7134  0.00775  0.00359  -0.0653  0.9423  0.0320
-7.000   -0.5049  0.00685  0.00249  -0.0618  0.8722  0.0368
-6.000   -0.4000  0.00657  0.00208  -0.0600  0.8253  0.0387
-5.000   -0.2942  0.00636  0.00173  -0.0584  0.7722  0.0394
-4.000   -0.1885  0.00615  0.00134  -0.0568  0.7124  0.0429
-3.000   -0.0818  0.00604  0.00109  -0.0553  0.6498  0.0466
-2.000    0.0233  0.00581  0.00086  -0.0536  0.5868  0.0912
-1.000    0.1289  0.00559  0.00073  -0.0521  0.5303  0.1658
-0.000    0.2346  0.00544  0.00069  -0.0506  0.4792  0.2579
 1.000    0.3394  0.00534  0.00074  -0.0490  0.4257  0.3737
 2.000    0.4404  0.00515  0.00087  -0.0466  0.3700  0.5653
 3.000    0.5375  0.00521  0.00113  -0.0435  0.2985  0.7481
 4.000    0.6300  0.00544  0.00149  -0.0391  0.2290  0.8987
 5.000    0.7526  0.00621  0.00208  -0.0418  0.1494  0.9785
 6.000    0.8798  0.00743  0.00289  -0.0459  0.0700  0.9927
 7.000    1.0038  0.00829  0.00357  -0.0491  0.0464  0.9974
 8.000    1.1275  0.00903  0.00424  -0.0523  0.0391  0.9998
 9.000    1.2136  0.00989  0.00502  -0.0474  0.0287  1.0000
10.000    1.2902  0.01100  0.00605  -0.0408  0.0174  1.0000
11.000    1.3687  0.01216  0.00719  -0.0349  0.0128  1.0000
12.000    1.4407  0.01341  0.00846  -0.0281  0.0104  1.0000
13.000    1.5045  0.01488  0.00997  -0.0204  0.0085  1.0000
14.000    1.5634  0.01687  0.01203  -0.0131  0.0070  1.0000
15.000    1.6206  0.01917  0.01442  -0.0068  0.0065  1.0000

```

Tab. 8: Součinitelé vztlaku, odporu a klopivého momentu vypočítané programem Xfoil.

Velmi často je pro jeho rychlost a jednoduché předávání příkazů používán program Xfoil. K dalším, velmi hojně používaným programům, patří programy XFLR5, a PANUKL.

10.1.2 Program ANSYS Fluent

ANSYS Fluent [134] je komplexní program využívaný pro počítačovou dynamiku tekutin (CFD - Computational Fluid Dynamics) , tedy pro simulace řešící problematiku proudění tekutin [135]. ANSYS Fluent se používá pro široké spektrum fyzikálních a chemických modelů [136]:

- laminární a turbulentní proudění,
- nestlačitelné a stlačitelné tekutiny,
- sdílení tepla konvekcí, kondukcí a radiací,
- modelování od jednoduchých chemických reakcí až po heterogenní chemické reakce spalování, pro příklad uhlí, plynů,
- vícefázové proudění se sdílením tepla a hmoty pro kombinace fází, tedy kapalina a plyn, tekutina a pevná fáze a interakcí mezi nimi.

Fyzikální modely a jejich kombinace v ANSYS Fluentu jsou schopny řešit téměř všechny problémy v průmyslu [137]:

- letecké inženýrství - letouny, příkladem je proudění vzduchu v okolí křídla, dále proudění tekutiny v okolí vesmírných lodí,
- pozemní doprava - snižování odporu vozidel,
- medicína - proudění krve v umělém srdci, proudění vzduchu při dýchání,
- chemické inženýrství - proudění skrz pumpy či trubky, spalování uhlí v pecích,
- energetické inženýrství - zvyšování účinnosti turbín, vývoj větrných elektráren.

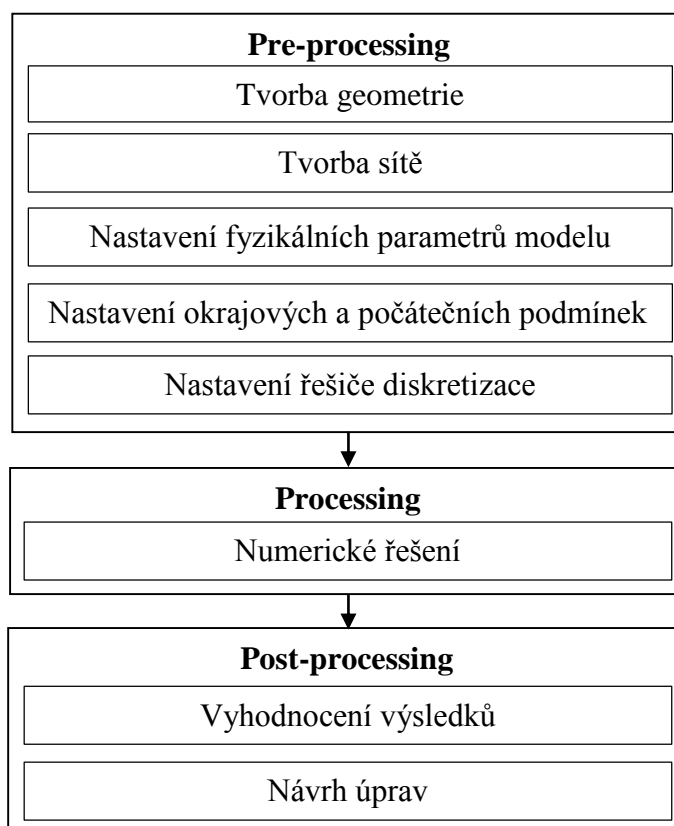
Multifyzikální simulace umožňuje předání výsledků jedné simulace jako vstupní data do další simulace.

ANSYS Fluent je velmi používaný na celém světě, od malých firem až po velké nadnárodní koncerny. Společnosti využívají program k simulacím v návrhové a optimalizační fázi vývoje výrobku, nebo ke kontrole již navržených produktů.

K dosažení snížení času, potřebného pro výpočet simulace, je možné využít paralelizace úloh na více procesorech nebo jádrech. Program umožňuje implementaci nových, uživatelem navržených modelů. Interaktivní nastavení programu ANSYS Fluent umožňuje kdykoliv pozastavit výpočet, analyzovat výsledky, změnit, zkorigovat nastavení a dál pokračovat v řešení.

V současnosti je ANSYS Fluent patří k nejpoužívanějším softwarem pro CFD simulace.

Počítačová dynamika tekutin CFD (Computational Fluid Dynamics)

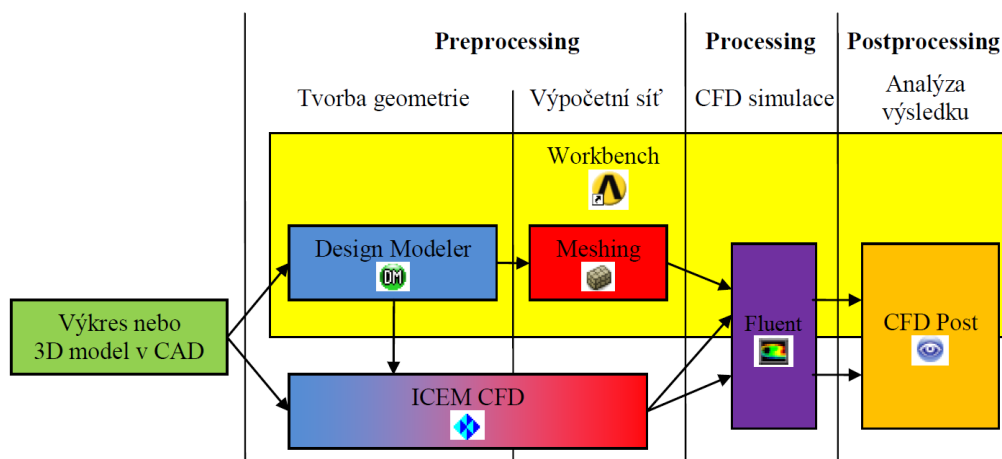


Obr. 39: 3 fáze CFD procesu [138].

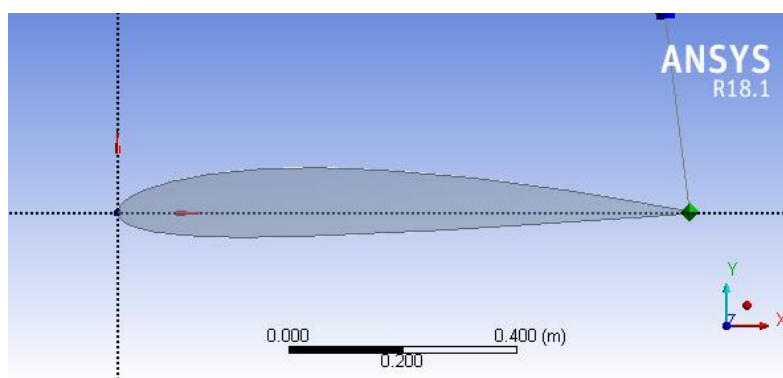
CFD simulace našly již v počátcích svého využívání široké uplatnění především v oblasti vojenských aplikacích jako jsou letecká technika, raketové zbraně a kosmické prostředky. CFD simulací bylo použito u letounu Panair či A502 od společnosti Boeing, Quadpan od společnosti Lockheed Martin atd. [137].

ANSYS Workbench

Program ANSYS Workbench nabízí jednotné prostředí pro široké spektrum specializovaných programů. V tomto prostředí lze vytvářet vazby mezi jednotlivými simulačními programy i projekty. Výsledky jednotlivých simulačních programů mohou být navzájem sdíleny a umožňují tak multifyzikální simulace [136].



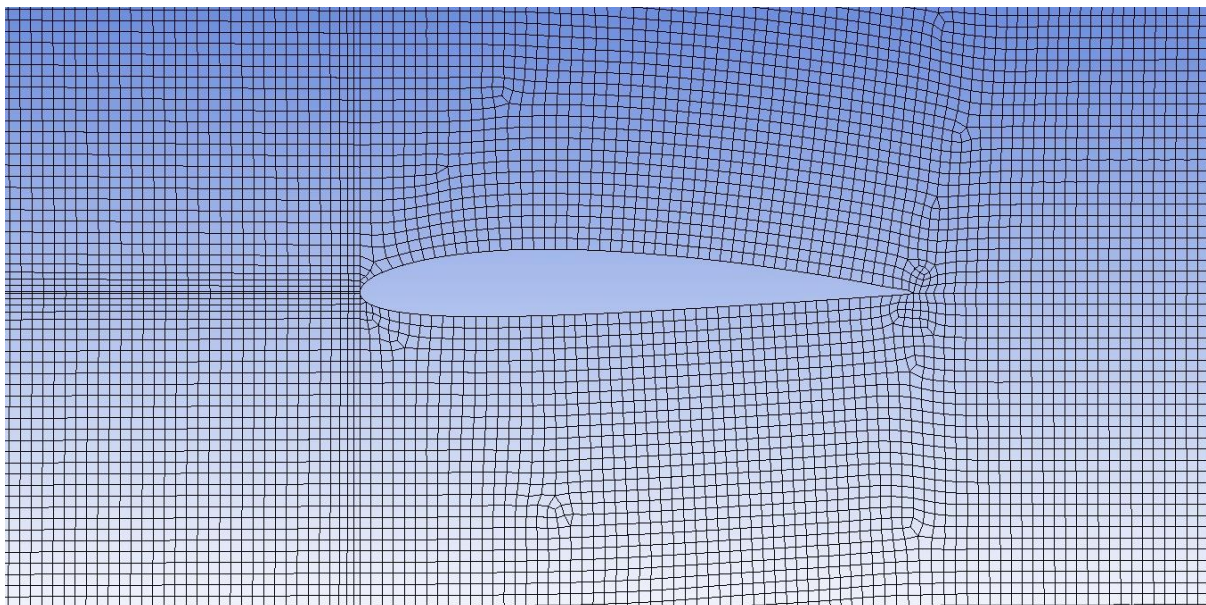
Obr. 40: Schéma numerické simulace a vzájemné vztahy mezi jednotlivými programy.



Obr. 41: Profil vytvořený na základě souboru souřadnic.

Sít' - Mesh - ANSYS Meshing

Program ANSYS Meshing je používán pro vytváření strukturované výpočetní sítě. V programu jsou využívány všechny typy elementů, lze vytvářet jak hybridní, tak uniformní síť, včetně mezní vrstvy. Základním požadavkem na kvalitu sítě buněk je jejich hustota. Ta je definována minimální a maximální velikostí buňky. Na kvalitě sítě velice záleží, protože ovlivňuje přesnost následných výpočtů a také rychlost konvergence iterací. Po spuštění je automaticky vytvořena vazba mezi modelem z programu ANSYS Design Modeler a ANSYS Meshingem (viz obr. 41). Pro 2D síť jsou akceptovatelné čtyřúhelníkové nebo trojúhelníkové buňky (viz obr. 42).

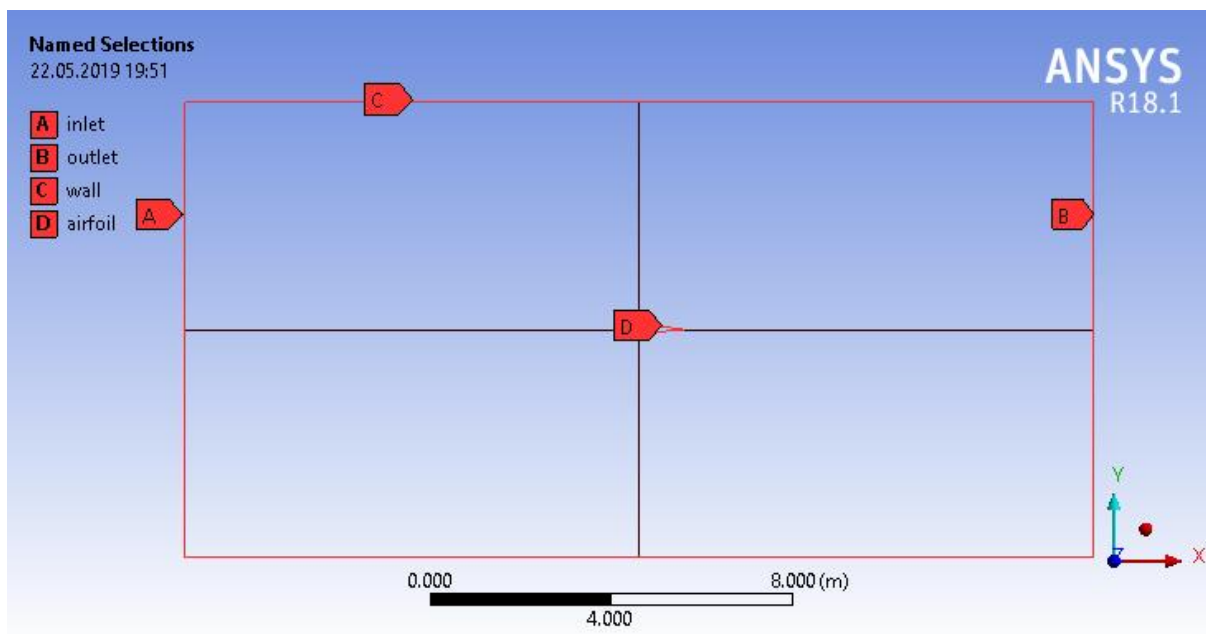


Obr. 42: Detail sítě kolem profilu.

Zavedení mezní vrstvy pomocí příkazu Inflation, přidávané k síti, bylo provedeno, ale při spuštění síťování byl Inflation vyřazen z činnosti. Obdobně tomu bylo i s podmínkou Face Sizing nastavenou na dělení povrchu profilu a s mapováním domén tj. Face Meshing. Uvedené vyloučení podmínek při tvorbě sítě nemělo vliv na přesnost následného výpočtu [139] [140].

Byly otestovány tři typy sítě ve tvaru C-H, H a O které se liší geometrií, ale na výsledcích CFD simulace se neprojevili.

Posledním krokem před vlastním výpočtem je definování jednotlivých částí aerodynamického tunelu.



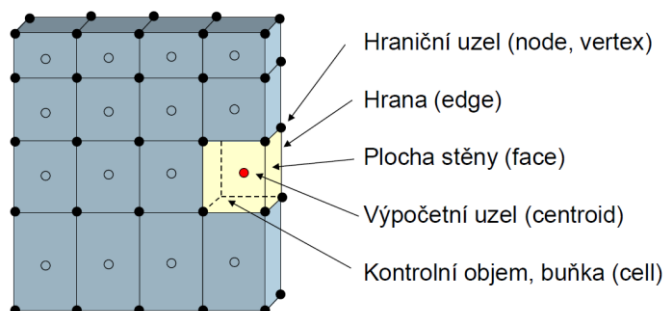
Obr. 43: Definování části aerodynamického tunelu.

Solution – ANSYS Fluent Solver

ANSYS Fluent Solver slouží pro samotné řešení úlohy připravené v ANSYS Fluent Preprocessoru Metodou konečných objemů (MKO) [141].

MKO se řadí do skupiny metod aproximujících diferenciální rovnici. Tyto metody převádí diskretizací diferenciální rovnice (rovnice kontinuity, pohybové, energie, transportní atd.) na soustavu rovnic algebraických.

MKO má širší výběr typů elementů sítě než Metoda konečných prvků (MKP). MKO má nižší paměťové a výpočetní nároky než MKP, je tedy vhodná i pro velké problémy, MKO má však nižší přesnost než MKP.

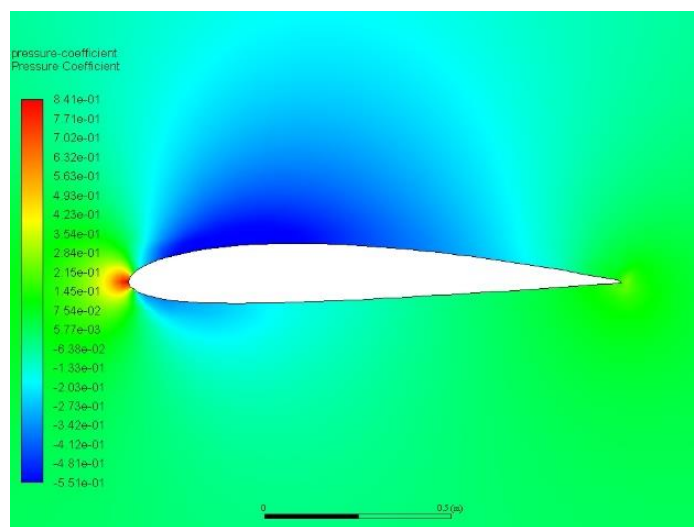


Obr. 44: Výpočetní síť – základní označení.

Proměnné jsou uloženy v těžištích objemů (cell-centered) nebo ve vrcholech (vertex-centered).

Results – ANSYS CFD-Post

ANSYS CFD-Post je program pro zobrazení výsledků z CFD simulace. Program obsahuje veškeré potřebné nástroje pro vizualizaci a analýzu výsledků dynamiky tekutin. Program umožňuje tvorbu konturovaných obrázků, vektorových polí proudnic, analýzu důležitých parametrů atd.



Obr. 45: Vizualizace proudového pole pro úhel náběhu 0° .

10.1.3 OpenFOAM

První verze programu OpenFOAM (Open-source Field Operation And Manipulation) byla vydána v roce 2004. Knihovny psané v programovacím jazyce C++ tvoří základ programu. OpenFOAM je primárně určen pro operační systémy Linux. Díky volnému přístupu k programu a možnosti snadné modifikace má velkou základnu uživatelů jak z komerčního tak i z akademického odvětví [142].

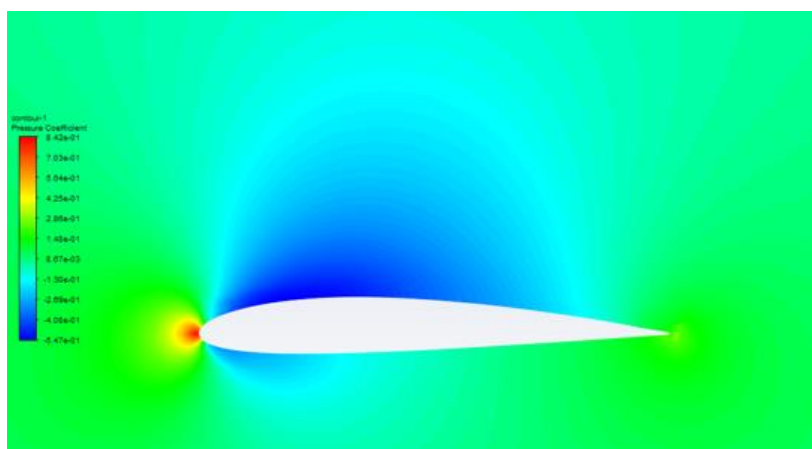
Program je určen k široké škále inženýrských úloh, jako je turbulentní proudění tekutin, chemické reakce, přestup tepla nebo elektromagnetismus.

K řešení CFD problémů využívá algoritmy Navier-Stokesových rovnic např. řešič SIMPLE, dále pak matematické metody konečného objemu a ploch, metody řešení soustav algebraických rovnic nebo linearizace nelineárních diferenciálních rovnic [131].

10.2 Srovnání programů

Ke srovnání programů je použito vlastností profilu vypočítaných v programu Xfoil a ANSYS Fluent na základě souřadnic profilu NACA 2412 a souřadnic profilu s nejlepšími výsledky z Paralelně hybridní optimalizace (kapitola Výsledky optimalizačních algoritmů). Dále je zde použit optimalizovaný profil NACA 2412 (kapitola Použití autokonfiguratoru IRACE).

10.2.1 Srovnání aerodynamických charakteristik profilu NACA 2412

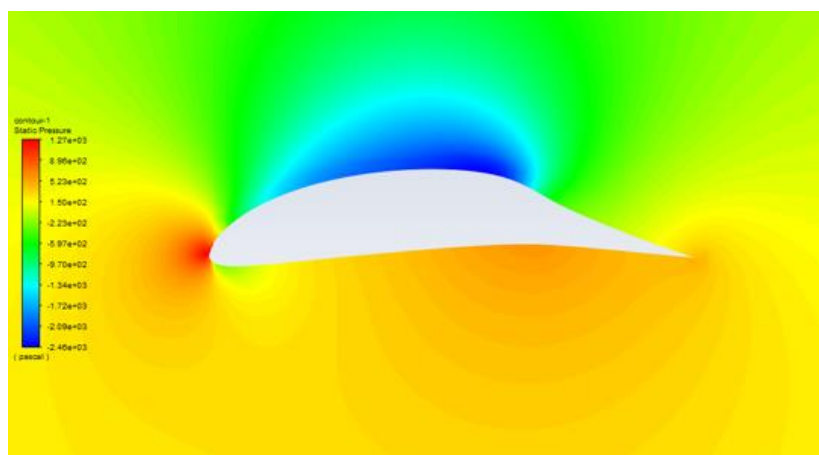


Obr. 46: Vizualizace proudového pole u profilu NACA 2412.

	Re	α [°]	C_L	C_D	C_L/C_D
Xfoil	5.0e+06	0.00	0.2346	0.0054	43.1250
Fluent	5.0e+06	0.00	0.2485	0.0050	49.4880
Rozdíl			+ 5.92 %	- 7.72 %	+ 14.75 %

Tab. 9: Srovnání výpočtových metod pro stanovení aerodynamických charakteristik profilu NACA 2412.

10.2.2 Srovnání optimalizovaného profilu na extrém



Obr. 47: Vizualizace proudového pole u optimalizovaného profilu.

	Re	α [°]	C_L	C_D	C_L/C_D
Xfoil	5.0e+06	0.00	1.0729	0.0065	164.8070
Fluent	5.0e+06	0.00	1.1251	0.0059	192.6520
Rozdíl			+ 4.87 %	- 9.83 %	+ 16.90 %

Tab. 10: Vliv výpočetních metod u optimalizovaného profilu na hodnotu maximální aerodynamické jemnosti.

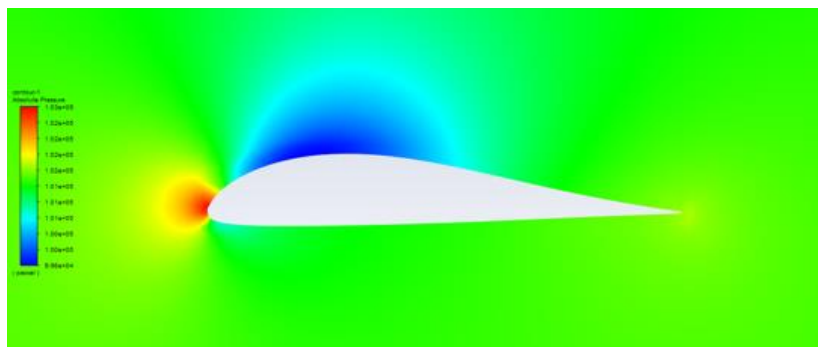
Z tlakových poměrů na horní straně profilu je zřejmé, že došlo k posunu maxima podtlaku k odtokové hraně profilu, což je známkou laminarity profilu. Obdobně je tomu i na spodní straně profilu, kde přetlak je více soustředěn k odtokové hraně.

U výpočtů byl nastavený úhel náběhu profilu na 0° a hodnota Reynoldsova čísla na 5.000×10^6 .

Ze srovnání vypočítaných aerodynamických vlastností jednotlivých profilů z tabulek výše lze konstatovat, že výsledky z výpočetního programu CFD Fluent jsou vždy reálnější než u výpočetního programu Xfoil a to většinou i více než 10 %.

Výjimkou jsou výsledné hodnoty u optimalizovaného profilu NACA 2412, kdy výsledky získané z Fluentu vykazují zvýšení hodnoty aerodynamického vztlaku o 21 % a současně je dosaženo o 80% nižší hodnoty aerodynamického odporu. Toto chování CFD Fluent je obtížně vysvětlitelné, je nezávislé na modelech obtékání, které jsou dostupné pro ANSYS Fluent. V ostatních případech je rozdíl mezi výsledky výpočetních programů minimální, což umožňuje použití Xfoil pro optimalizaci profilu.

10.2.3 Srovnání výsledků výpočetních metod u optimalizovaného profilu NACA 2412



Obr. 48: Vizualizace proudového pole u optimalizovaného profilu NACA 2412.

	Re	α [°]	C_L	C_D	C_L/C_D
Xfoil	5.0e+06	0.00	0.3128	0.0062	50.8610
Fluent	5.0e+06	0.00	0.3785	0.0012	307.2000
Rozdíl			+ 21.00 %	- 80.00 %	+ 504.00 %

Tab. 11: Srovnání aerodynamických charakteristik optimalizovaného profilu NACA 2412.

Z obrázku 48 je zřejmé, že po optimalizaci se zvětšila tloušťka profilu v důsledku většího zakřivení horní strany profilu a snížilo se zakřivení spodní strany profilu. Uvedené změny v geometrii profilu se výrazně promítly do výsledných hodnot aerodynamického součinitele vztlaku a odporu. (viz tab. 11). Současně se zúžila oblast podtlaku na horní straně profilu a zcela vymizel podtlak na spodní straně profilu.

Nastavení geometrie, vyjma profilu samotného, nastavení sítě a stejně tak nastavení fyzikálních konstant a výběr modelu jsou stejné a i přesto jsou výsledky různé (viz tab. 9, 10, 11).

10.3 Výběr programu

Program Xfoil

Výhody:

- zadání za pomoci pouze tří vstupů pro výpočet,
- vypočítá najednou i více úhlů náběhu,
- vypočítání aerodynamických charakteristik profilu v řádu sekund.

Nevýhody:

- slouží pouze k výpočtu aerodynamických charakteristik profilu,
- méně přesné vypočítané aerodynamické charakteristiky.

Program ANSYS Fluent

Výhody:

- je schopen řešit téměř všechny problémy v průmyslu,
- je přesnější, poskytuje reálnější vypočítané aerodynamické charakteristiky.

Nevýhody:

- poměrně složité zadání pro výpočet,
- vypočítá najednou pouze jeden úhel náběhu,
- časová náročnost pro stanovení aerodynamických charakteristik profilu je v řádu desítek minut.

10.3.1 Program Xfoil

I přesto, že program ANSYS Fluent počítá aerodynamické charakteristiky profilu přesněji, je vybrán program Xfoil, protože při optimalizačních výpočtech profilů je primární rychlost výpočtu. ANSYS Fluent je možné použít pro verifikaci výsledků pocházejících z programu Xfoil.

11 Vytvoření nových variant optimalizací

Jsou vytvořeny nové varianty optimalizací původních optimalizací modifikovaných pro aplikaci aerodynamikou.

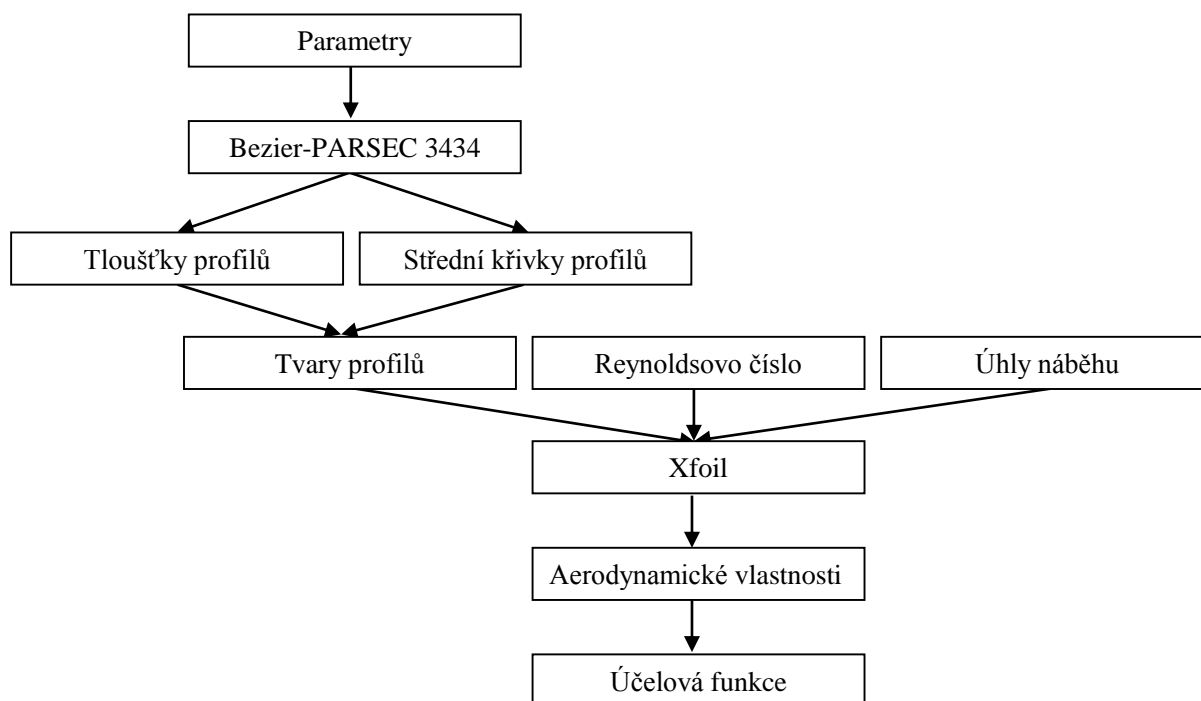
11.1 Varianta optimalizace aEA

Bylo použito nové varianty Evolučních algoritmů, které jsou atypické způsobem křížení, který je uzpůsoben k implementaci parametrizace.

Parametrizace

Tvar profilu byl získán parametrizací Bezier-PARSEC 3434 [82] modelu na základě vstupujících parametrů. Na počátku jsou vytvořeny náhodné profily náhodnými parametry, ze kterých se vychází. Náhodné parametry střední křivky a tloušťky jsou omezeny, tak aby výsledný profil byl použitelný. Při každé iteraci jsou změněny parametry, z kterých jsou vypočítány nové tvary profilů s novými aerodynamickými vlastnostmi (viz obr. 49).

Program Xfoil z tvaru profilu, Reynoldsova čísla a úhlů náběhu vypočítá aerodynamické vlastnosti tvaru profilu, tj. odpor, vztlak a moment.



Obr. 49: Výpočet aerodynamických vlastností na základě parametrů.

Účelová funkce

Je hledán ze všech profilů takový, který má největší aerodynamickou jemnost.

$$\max_{1 \leq j \leq N} (C_L/C_D)_j \quad (46)$$

N = všechny optimalizované úhly náběhu

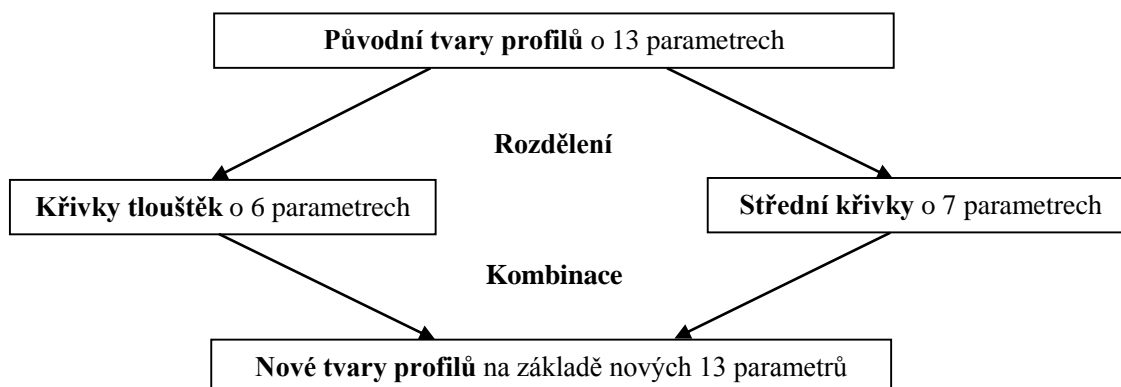
C_L = součinitel vztlaku

C_D = součinitel odporu

Optimalizace

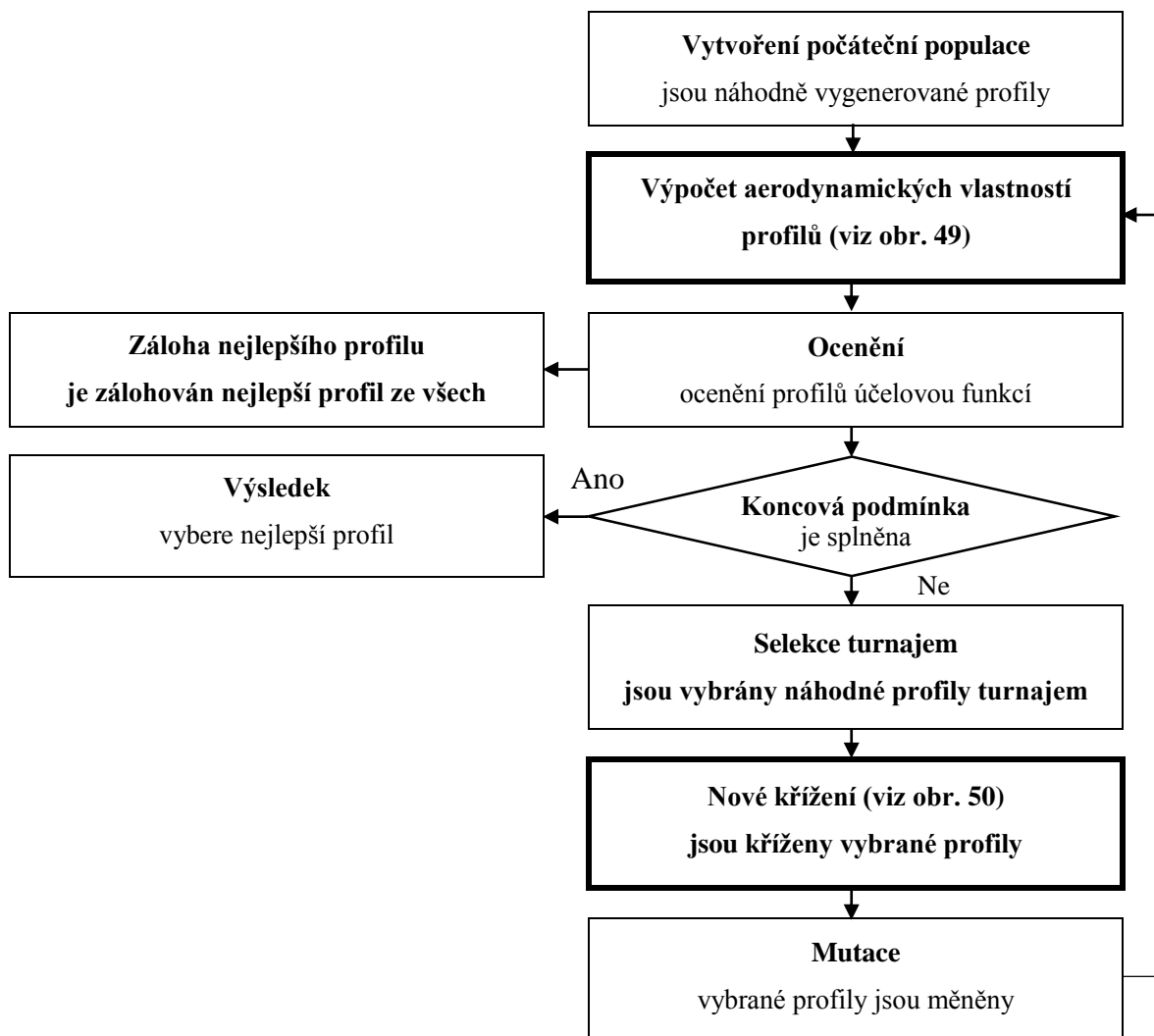
Při použití Bezier-PARSEC 3434 parametrizace je tvar profilu charakterizován 15 parametry. Pokud jsou vynechány konstantní parametry, pak ze zbývajících 13 parametrů lze vytvořit křivku tloušťky a střední křivku. Křivka tloušťky je definovaná 6 parametry a střední křivka 7 parametry, kdy při sloučení parametrů lze dostat tvar profilu.

Při novém křížení s Bezier-PARSEC 3434 parametrizací jsou profily rozděleny na křivky tloušťky a střední křivky. Po rozdělení se mezi sebou se nejkvalitnější křivky tloušťky a střední křivky sloučí a vytvoří nový tvar profilu (viz obr. 50).



Obr. 50: Nové křížení profilů kombinací parametrů Bezier-PARSEC 3434 parametrizace.

Pro potenciální obohacení populace může být zálohován nejlepší profil, který je později vkládán. Inicializace je provedena na základě náhodně vygenerovaných profilů reprezentovaných náhodnými parametry z Bezier-PARSEC modelu, přičemž náhodné parametry střední křivky a tloušťky jsou limitovány, tak aby výsledný profil byl realizovatelný.



Obr. 51: Obecné schéma aEA se selekcí turnajem a novým křížením.

11.2 Varianta optimalizace aPSO

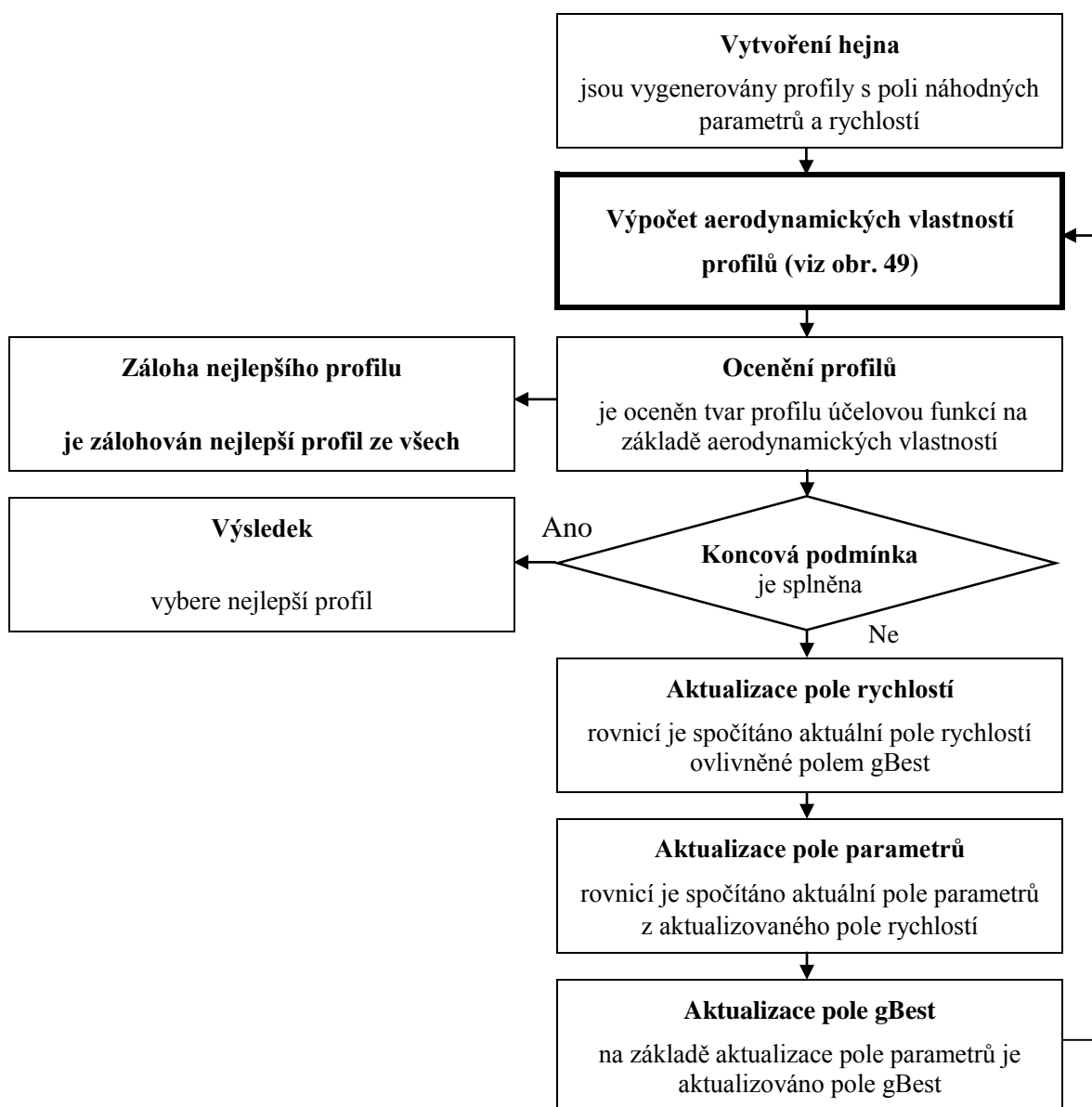
Optimalizace profilu hejnem částic je upravenou variantou optimalizace hejnem částic Many Optimizing Liaisons (MOL) pro snadnější implementaci parametrizace a větší rychlost výpočtu.

Parametrizace a účelová funkce jsou stejné jako u aEA (Bezier-PARSEC 3434 a maximální aerodynamická jemnost).

Optimalizace

Byla rozšířena pozice a rychlost částice na pole pozic a pole rychlostí tak, aby bylo možno použít Bezier-PARSEC parametrizaci. Z hlediska aplikace pak namísto částic byly jmenovány profily, namísto pozic byly jmenovány parametry.

Na počátku jsou vytvořeny náhodné parametry pro vytvoření hejna. Při každé iteraci jsou změněny parametry, z kterých jsou vypočítány nové tvary profilů s novými aerodynamickými vlastnostmi.



Obr. 52: Obecné schéma optimalizace hejnem částic s inicializací Bezier-PARSEC parametry.

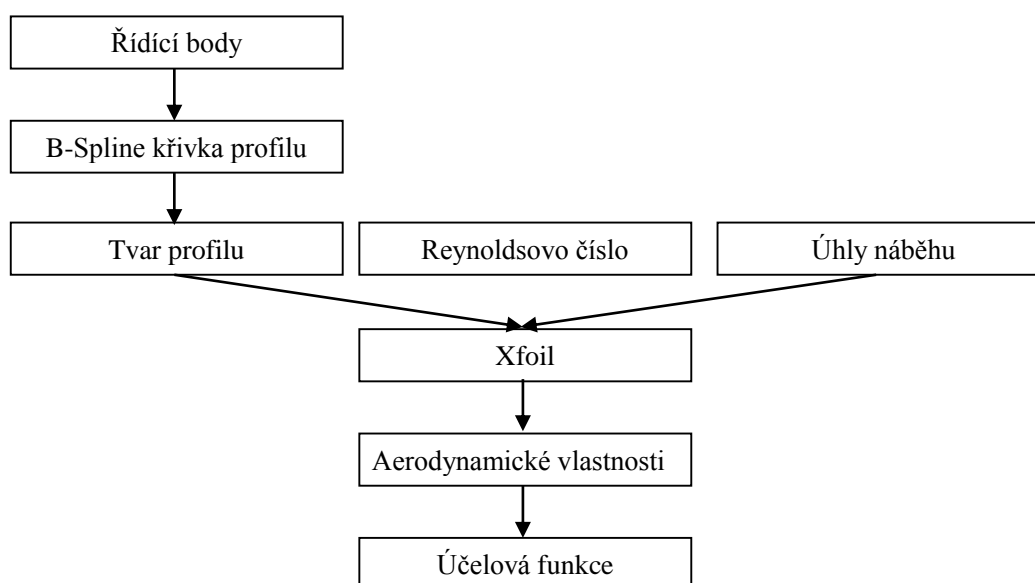
11.3 Varianta optimalizace aPSO s výchozím profilem

Optimalizace profilu hejnem částic je znovu upravenou variantou optimalizace hejnem částic Many Optimizing Liaisons (MOL).

Parametrizace

Pro optimalizaci výchozího profilu je možné použít výchozí profil pro počáteční tvar. Na počátku jsou řídicí body B-spline křivky odpovídají tvaru optimalizovaného profilu. Při každé iteraci jsou pak pozměněny řídicí body, z nichž vychází nový tvar (viz obr. 53).

Programové vybavení Xfoil může z tvaru profilu, Reynoldsova čísla a úhlů náběhu vypočítat aerodynamické vlastnosti tvaru profilu, tj. odpor, vztlak a moment.



Obr. 53: Výpočet aerodynamických vlastností na základě řídicích bodů.

Účelová funkce

Je definován rozsah úhlů náběhu, v kterém má být optimalizován výchozí profil. Výsledkem je vybrání profilu s nejlepším průměrem aerodynamické jemnosti v definovaném rozsahu.

$$\sum_{j=1}^N (C_L/C_D)_j / N \quad (47)$$

N = všechny optimalizované úhly náběhu

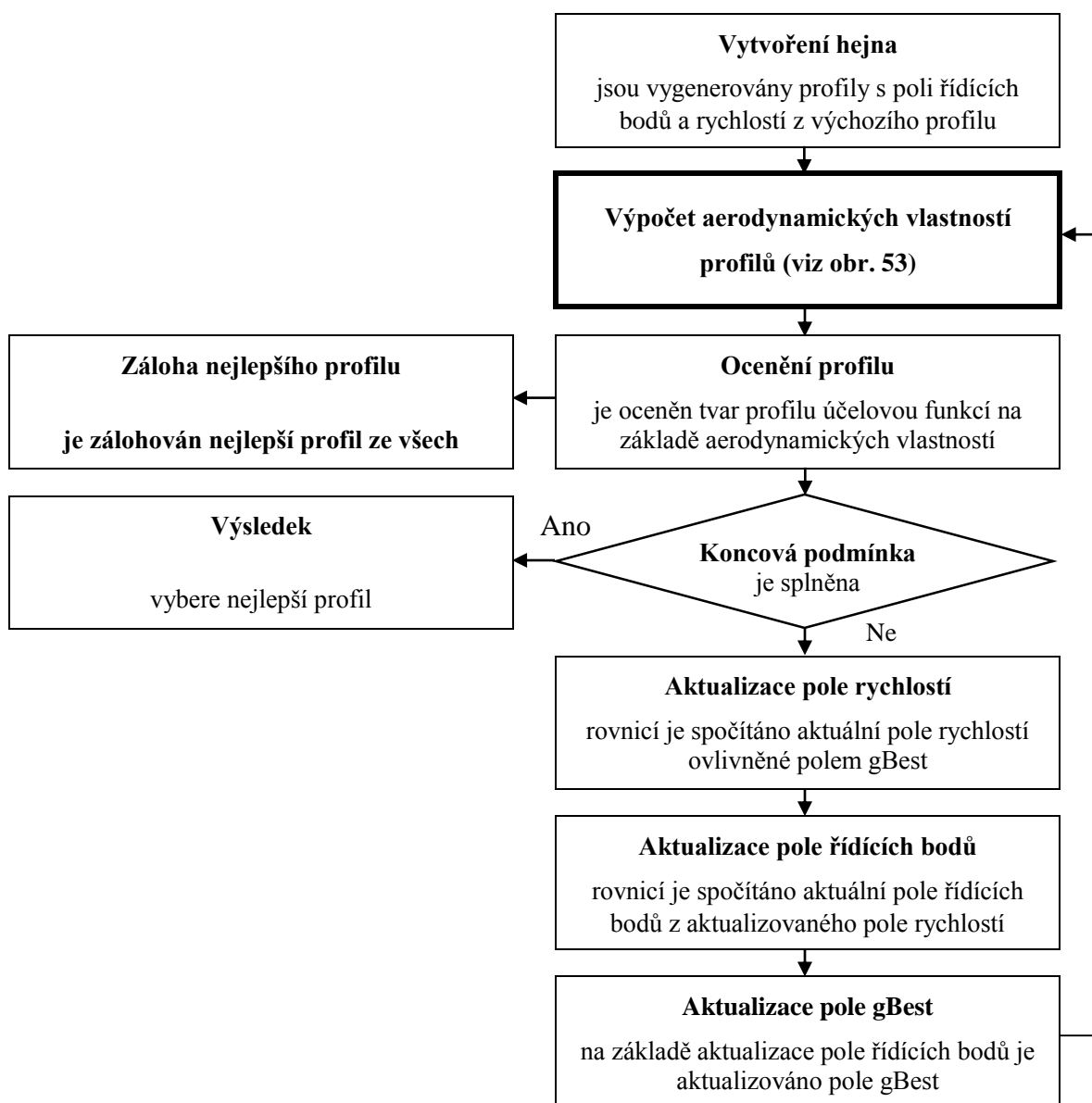
C_L = součinitel vztlaku

C_D = součinitel odporu

Optimalizace

Optimalizace profilu hejnem částic s výchozím profilem vychází z optimalizace Many Optimizing Liaisons. Stejně je také, že místo jedné pozice je pole pozic a místo jedné rychlosti pole rychlostí. Při aplikování jsou jmenovány profily namísto částic, namísto pozic jsou jmenované řídicí body.

Pro vytvoření hejna bylo použito řídicích bodů křivky B-spline tvaru zvoleného profilu. V každé iteraci jsou omezeně změněny vertikální souřadnice řídicích bodů, přičemž horizontální zůstávají. Každou změnou je definován nový B-spline tvar s novými aerodynamickými vlastnostmi.



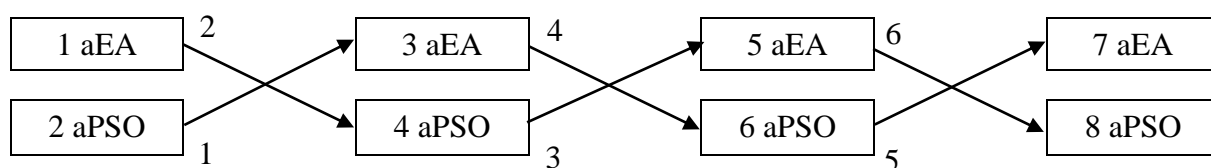
Obr. 54: Obecné schéma optimalizace hejnem částic s použitím řídicích tvarů profilu.

12 Vytvoření Paralelně hybridní optimalizace PHO

Na základě zkušeností s optimalizacemi popsány v předchozích kapitolách byla navržena a vytvořena zcela nová a původní paralelní hybridní optimalizace.

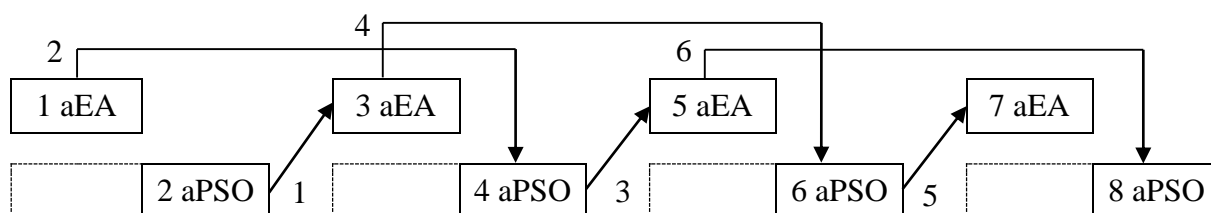
Pro hybridizaci byla vybrána optimalizace profilu evolučními algoritmy (aEA) a dále je zvolena optimalizace profilu hejnem částic (aPSO) založená na variantě optimalizace Many Optimizing Liaisons (MOL). Díky použití obou optimalizací dochází také ke spojení jejich výhod a díky paralelizmu je zaručena diverzita nového typu optimalizace.

Je zvolena n -tice první optimalizace a zároveň n -tice druhé optimalizace tvořící hybridní optimalizaci. Každá ze zvolených optimalizací by měla být opakovaně spuštěna paralelně, díky čemuž byl konečný výpočet urychlen ve srovnání se sekvenčním výpočtem. Po každém kroku, definovaném časovým intervalem, by si měly optimalizace vyměnit optimalizované výsledné data.

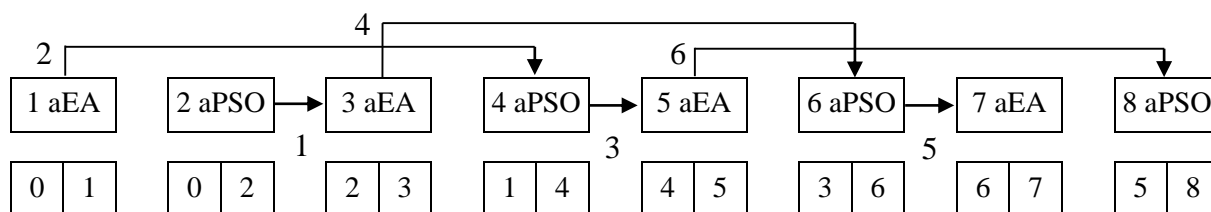


Obr. 55: Paralelní hybridní optimalizace PHO při střídání aEA a aPSO.

Kvůli komplexitě paralelizace je konečná optimalizace převedena na sekvenční postup výpočtu. Výhodou této transformace je možnost definovat pro aEA a aPSO různý krok, tedy různý časový interval pro různou optimalizaci. Nové varianty optimalizací aEA a aPSO se spouštějí za sebou a vyměňují si mezi sebou populaci, hejno. Na obrázku 57 je vidět po sobě jdoucí aEA a aPSO, kdy v spodním řádku je vstupní populace, hejno a výstupní populace, hejno.



Obr. 56: Mezi-stav paralelní hybridní optimalizace PHO při střídání aEA a aPSO.



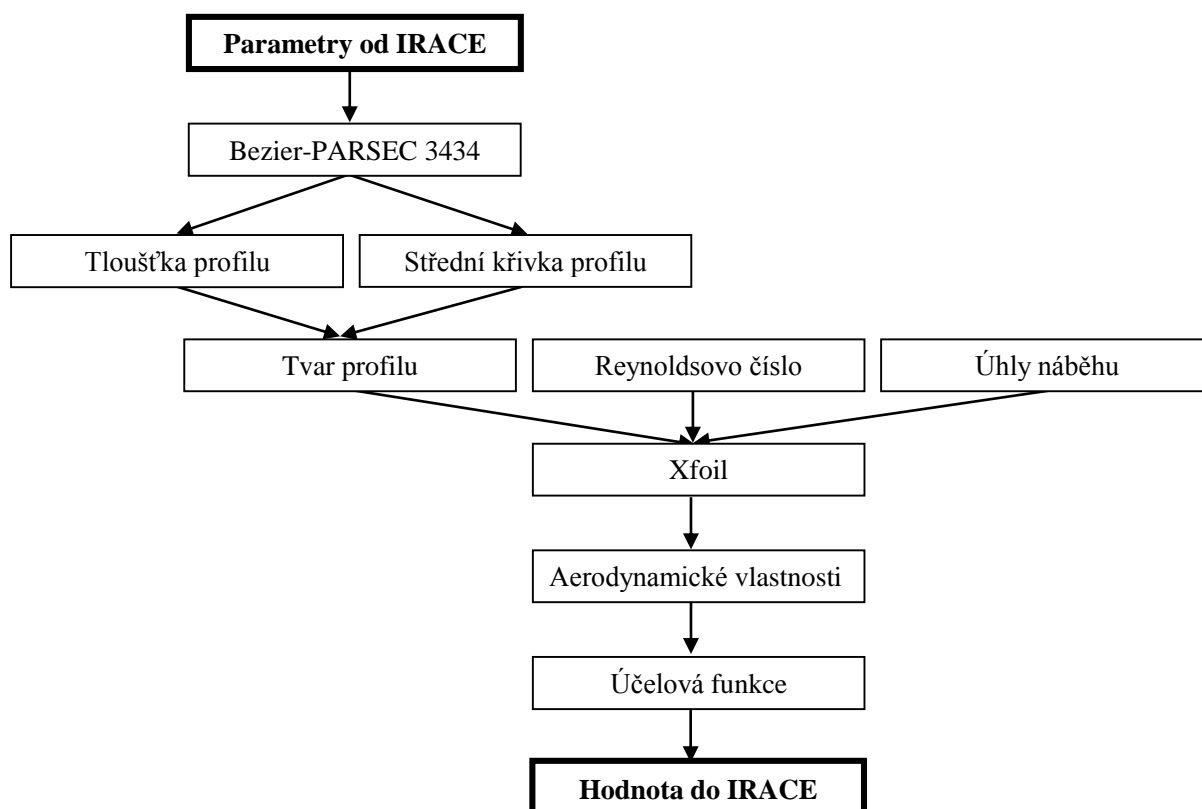
Obr. 57: Paralelní hybridní optimalizace PHO při střídání aEA a aPSO převedena na sekvenční.

13 Použití autokonfiguratoru IRACE

Autokonfigurator IRACE lze použít dvěma způsoby. Díky aplikaci optimalizací na aerodynamiku se vyskytuje parametrizace profilu, tedy se nemusí přímo konfigurovat optimalizace, stačí pouze použít IRACE na parametrizaci profilu.

13.1 Autokonfigurace parametrizací profilu

13.1.1 IRACE použitý na parametrizaci Bezier-PARSEC 3434



Obr. 58: Aplikace IRACE autokonfiguratoru na Bezier-PARSEC 3434 parametrizaci.

Účelová funkce

Hledá se úhel náběhu s největší aerodynamickou jemností.

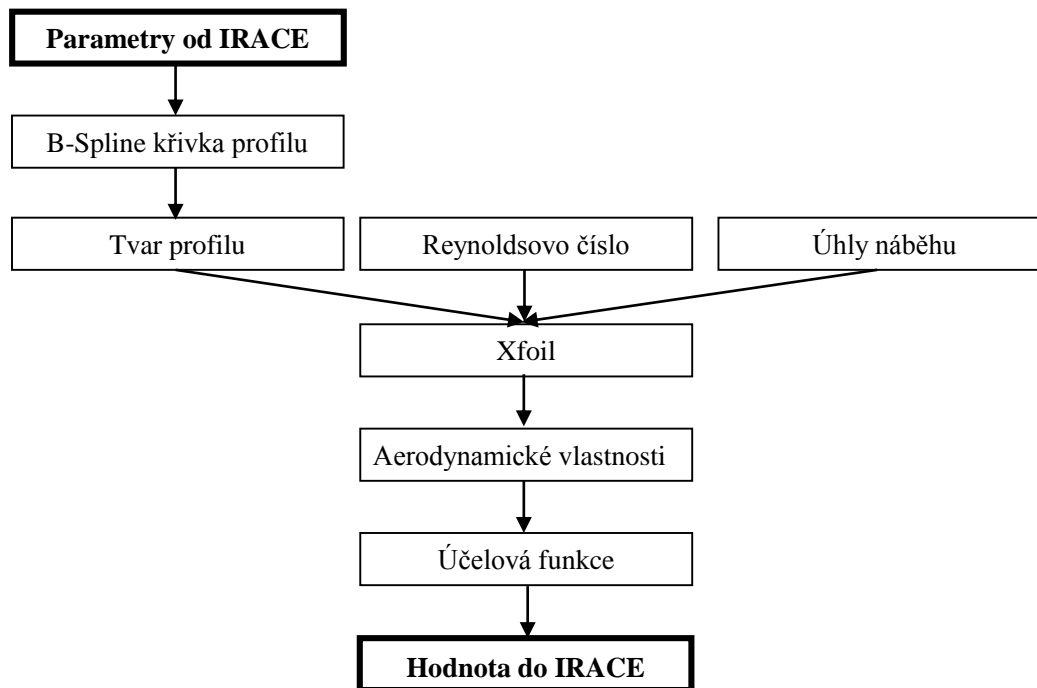
$$\max_{1 \leq j \leq N} (C_L/C_D)_j \quad (48)$$

N = všechny optimalizované úhly náběhu

C_L = součinitel vztlaku

C_D = součinitel odporu

13.1.2 IRACE použitý na parametrizaci B-spline



Obr. 59: Aplikace autokonfiguratoru IRACE na parametrizaci B-spline.

Účelová funkce

Je definován rozsah úhlů náběhu, na kterém se hledá největší průměr aerodynamické jemnosti.

$$\sum_{j=1}^N (C_L/C_D)_j / N \quad (49)$$

N = všechny optimalizované úhly náběhu

C_L = součinitel vztlaku

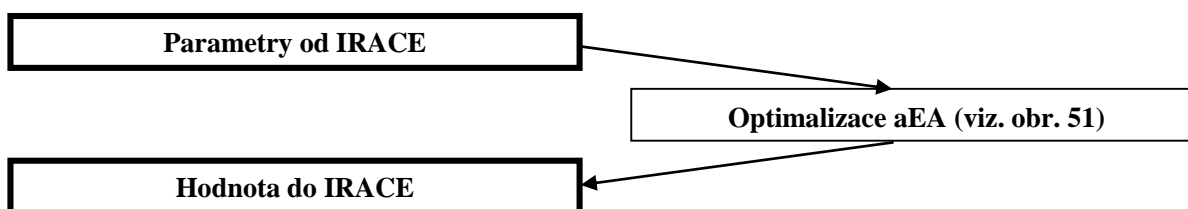
C_D = součinitel odporu

13.2 Autokonfigurace optimalizací

Není použito IRACE přímo na parametrizaci profilu, ale IRACE je použito na autokonfiguraci parametrů optimalizace. Hlavní nevýhodou autokonfigurace parametrů optimalizací je výpočetní náročnost, tedy je nutné použít superpočítač.

13.2.1 IRACE použitý na optimalizaci aEA

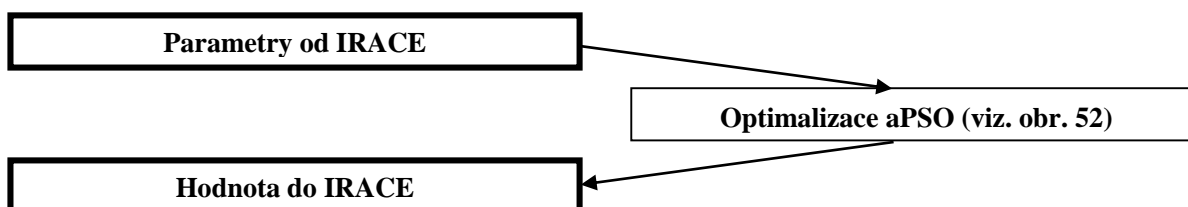
Konfigurací pro aEA jsou parametry jako velikost populace, pravděpodobnost křížení, velikost turnaje.



Obr. 60: Aplikace IRACE na optimalizaci aEA.

13.2.2 IRACE použitý na optimalizaci aPSO

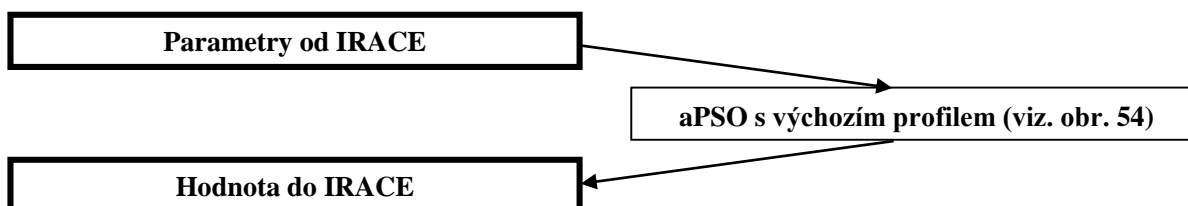
Pro konfiguraci aPSO jsou optimalizovány parametry jako velikost hejna, parametr váhy setrvačnosti, akcelerační koeficient.



Obr. 61: Aplikace IRACE na optimalizaci aPSO.

13.2.3 IRACE použitý na optimalizaci aPSO s výchozím profilem

Pro konfiguraci aPSO jsou optimalizovány parametry jako velikost hejna, parametr váhy setrvačnosti, akcelerační koeficient.



Obr. 62: Aplikace IRACE na optimalizaci aPSO s výchozím profilem.

14 Výsledky optimalizačních algoritmů

Vytvořené optimalizační algoritmy jsou novými variantami původních optimalizačních algoritmů, tedy kromě paralelně hybridní optimalizace a optimalizace s výchozím profilem. Od původních optimalizačních algoritmů se liší implementací parametrizace Bezier-PARSEC 3434 nebo použitím optimalizace výchozího profilu.

Pro parametrizace Bezier-PARSEC nebo parametrizační model B-spline je velikost

Za pomoci vytvořených optimalizačních algoritmů, jak z hlediska nové struktury, tak i nastavením nových konstant, jsou vytvářeny nové algoritmy.

Pro kvalitativní srovnání jednotlivých změn v struktuře a konstant je vhodné použít stejná počáteční data tak, aby došlo k co největší eliminaci náhodných vlivů. Díky náhodným vlivům jsou výsledky stejného typu optimalizace odlišné a pro srovnání algoritmů musí dojít k několikanásobnému spuštění optimalizace. Díky možnosti srovnání různých výsledků optimalizace lze jednotlivé algoritmy mezi sebou porovnávat.

Parametrizace

Pro parametrizace Bezier-PARSEC nebo parametrizační model B-spline je velikost populace, resp. hejna zvolena na 25 jedinců, částic, což by mělo být ověřeno autokonfiguračním nástrojem IRACE.

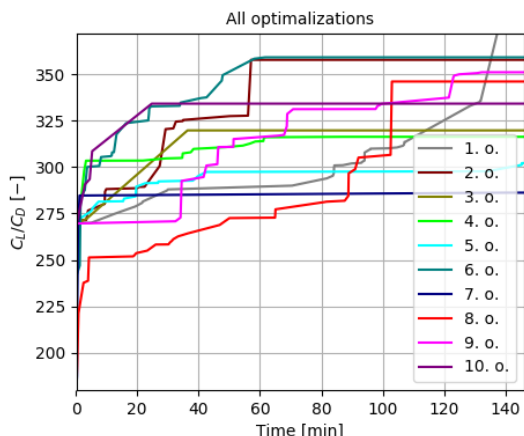
Popis grafu optimalizace

Pro ukončení jedné optimalizace je použita koncová podmínka, založená na nastavení doby trvání jedné optimalizace cca 150 minut (osa x), kdy po uplynutí 150 minut již nedocházelo k zlepšení výsledků optimalizací. Pro optimalizaci bylo použito aerodynamické jemnosti jako ocenění účelové funkce C_L/C_D (osa y).

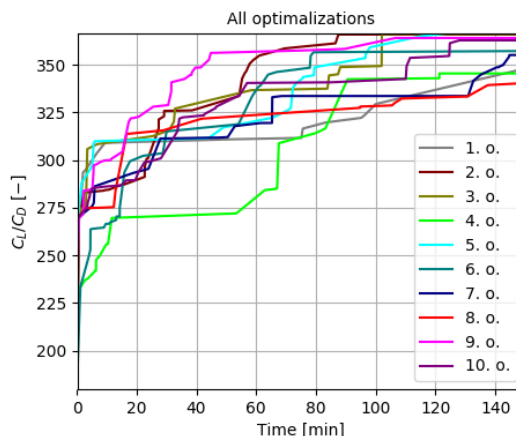
14.1 Výsledky optimalizace aEA

Pro hledání optimálního extrému je výhodné použití optimalizace profilu evolučními algoritmy spolu s parametrizací Bezier-PARSEC 3434. Tyto algoritmy mají novým inovativním způsobem řešeno křížení jedinců (viz obr. 50), tak, aby bylo využito výhod parametrizace. Pro kooperaci nebo pro srovnání s ostatními typy optimalizací je použita stejná počáteční populace.

14.1.1 Velikost turnaje 3 a 6 jedinců



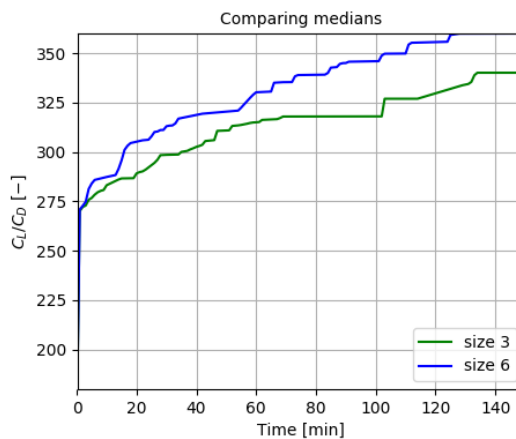
Obr. 63: Všechny aEA velikost turnaje 3.



Obr. 64: Všechny aEA velikost turnaje 6.

a)	371.92053	b)	347.94007
	357.76765		365.94810
	319.75052		366.17391
	316.28959		345.48223
	302.14854		366.47541
	359.11894		357.21030
	286.20690		355.15991
	346.07792		345.43520
	351.07486		364.08624
	334.16507		362.84861

Tab. 12: Pro Mann-Whitneyův test ocenění aEA velikost turnaje a) 3, b) 6.

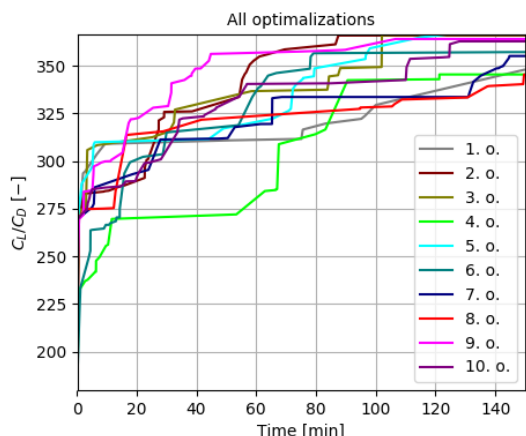


Obr. 65: Medián aEA velikost turnaje 3 a 6.

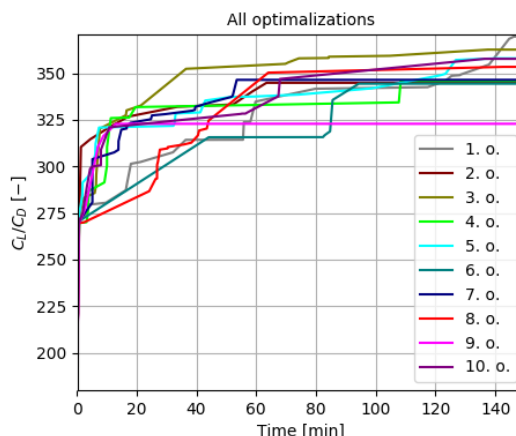
Optimalizace	Mann-Whitneyův test	Maximální medián
aEA s turnajem o 3 jedincích	p-hodnota 0.064022 > 0.05 výsledky nejsou signifikantně rozdílné	340.12149
aEA s turnajem o 6 jedincích		360.02945

Tab. 13: Výsledek srovnání aEA s velikostí turnaje 3 a 6 jedinců.

14.1.2 Velikost turnaje 6 a 9 jedinců



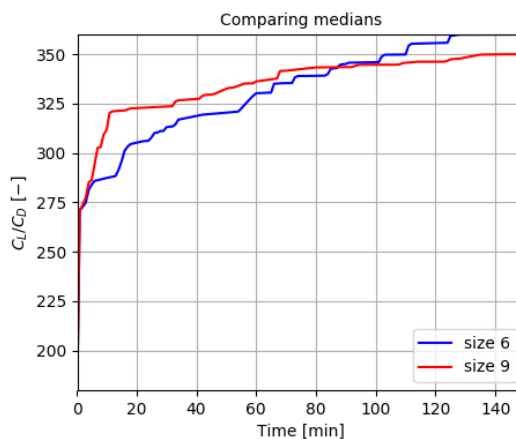
Obr. 66: Všechny aEA velikost turnaje 6.



Obr. 67: Všechny aEA velikost turnaje 9.

a)	347.94007	b)	370.89015
	365.94810		344.91863
	366.17391		362.73885
	345.48223		345.93381
	366.47541		358.06324
	357.21030		344.43359
	355.15991		346.49895
	345.43520		353.46856
	364.08624		322.90135
	362.84861		357.79468

Tab. 14: Pro Mann-Whitneyův test ocenění aEA velikost turnaje a) 6, b) 9

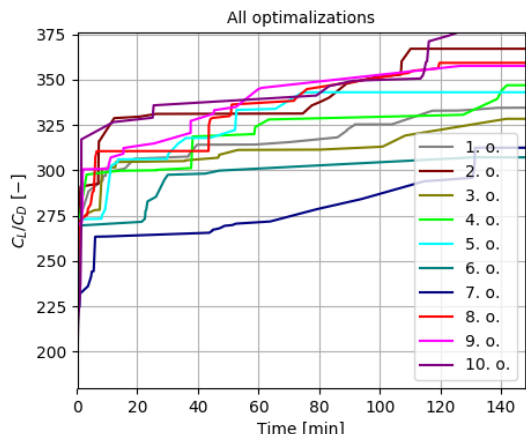


Obr. 68: Medián aEA velikost turnaje 6 a 9.

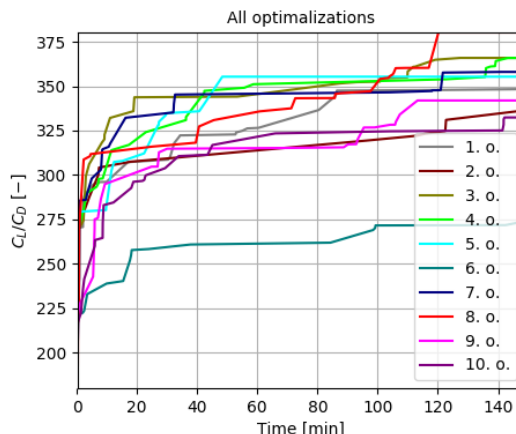
Optimalizace	Mann-Whitneyův test	Maximální medián
aEA s turnajem o 6 jedincích	p-hodnota 0.18588 > 0.05 výsledky nejsou signifikantně rozdílné	360.02945
aEA s turnajem o 9 jedincích		349.98376

Tab. 15: Výsledek srovnání aEA s velikostí turnaje 3 a 6 jedinců.

14.1.3 Znovu vkládání nejlepších jedinců po 10 a 20 minutách



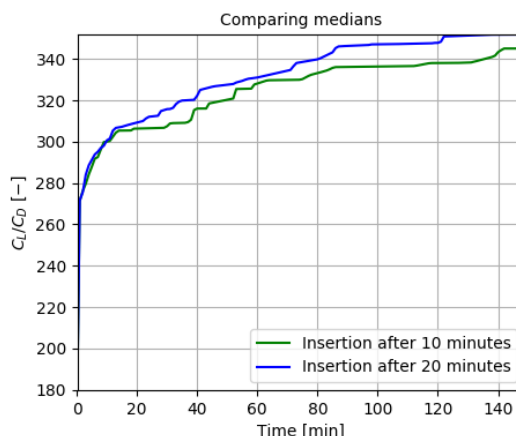
Obr. 69: Všechny aEA vkládání po 10 min.



Obr. 70: Všechny aEA vkládání po 20 min.

a)	334.58244	b)	348.35590
	367.08835		336.25219
	328.43548		365.93617
	346.98152		365.89147
	343.10696		355.34749
	307.19770		278.75350
	312.50646		358.04255
	359.31590		380.28398
	357.62419		341.94946
	376.09658		332.38095

Tab. 16: Pro Mann-Whitneyův test ocenění aEA vkládání po a) 10 min., b) 20 min.

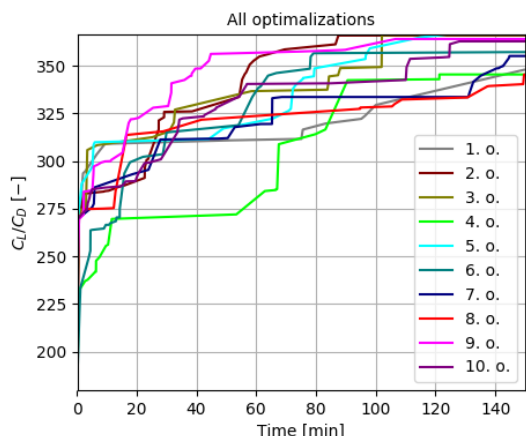


Obr. 71: Medián aEA vkládání po 10 min. a 20 min.

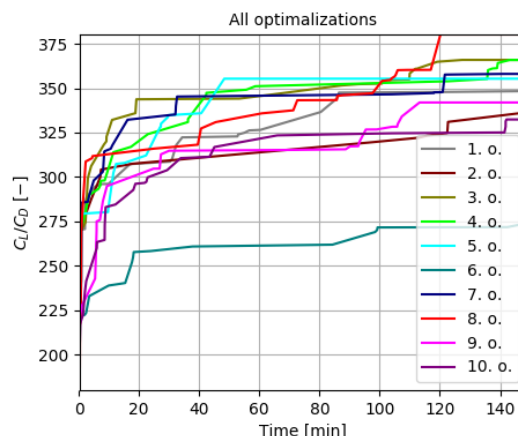
Optimlizace	Mann-Whitneyův test	Maximální medián
aEA vkládání po 10 min.	p-hodnota 0.67758 > 0.05 výsledky nejsou signifikantně rozdílné	345.04424
aEA vkládání po 20 min.		351.85169

Tab. 17: Výsledek srovnání aEA se znovu vkládání po 10 min. a 20 min.

14.1.4 Velikost turnaje 6 a vkládání nejlepších jedinců po 20 minutách



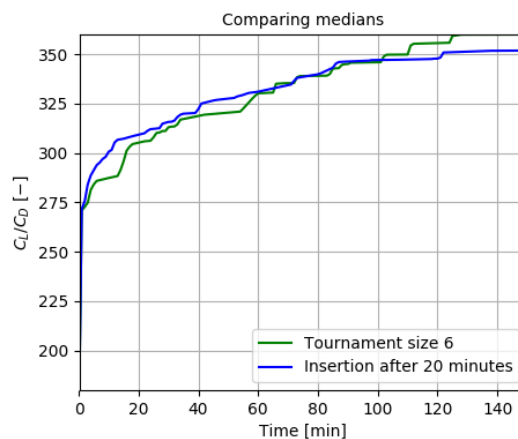
Obr. 72: Všechny aEA velikost turnaje 6.



Obr. 73: Všechny aEA vkládání po 20 min.

a)	347.94007	b)	348.35590
	365.94810		336.25219
	366.17391		365.93617
	345.48223		365.89147
	366.47541		355.34749
	357.21030		278.75350
	355.15991		358.04255
	345.43520		380.28398
	364.08624		341.94946
	362.84861		332.38095

Tab. 18: Pro Mann-Whitneyův test ocenění aEA velikost turnaje a) 6, b) vkládání 20 min.



Obr. 74: Medián aEA velikost turnaje 6 a vkládání 20 min.

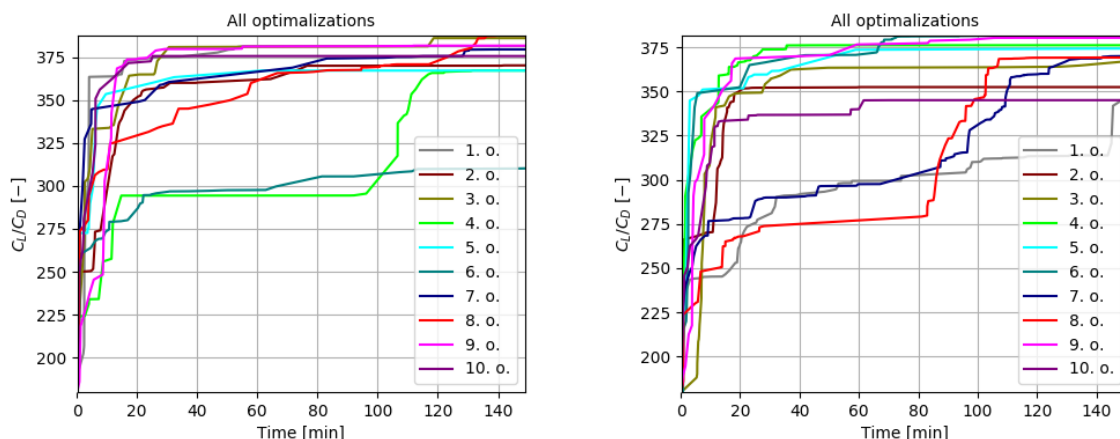
Optimalizace	Mann-Whitneyův test	Maximální medián
aEA s turnajem o 6 jedincích	p-hodnota 0.30749 > 0.05 výsledky nejsou signifikantně rozdílné	360.02945
aEA vkládání po 20 min.		351.85169

Tab. 19: Výsledek srovnání aEA s turnajem o 6 jedincích a znovu vkládání po 20 min.

14.2 Výsledky optimalizace aPSO

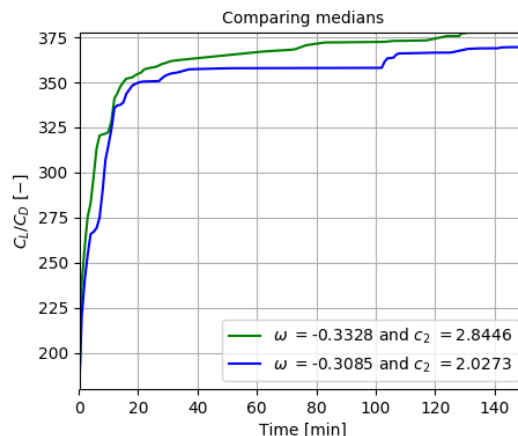
Byla navržena optimalizace profilu hejnem částic s Bezier-PARSEC 3434 parametrizací, která je použita pro hledání extrému. Původní algoritmy byly změněny za účelem optimalizace a spolu s novými konstantami ovlivňují kvalitu výsledků optimalizace. Ke srovnání je použito takových počátečních parametrů hejna, aby mohla optimalizace profilu hejnem částic dále kooperovat nebo se srovnávat s jinými typy optimalizace.

14.2.1 Omega -0.3328 a -0.3085, c2 2.8446 a 2.0273



Obr. 75: Všechny aPSO ω -0.3328 a c_2 2.8446. Obr. 76: Všechny aPSO ω -0.3085 a c_2 2.0273.

a)	381.51394	b)	345.05643
	370.27290		352.46552
	386.16883		366.73285
	367.29572		376.16307
	367.24528		374.27699
	310.23810		381.63793
	379.58167		370.00000
	387.66292		369.19101
	381.78261		380.38610
	375.71992		345.07273

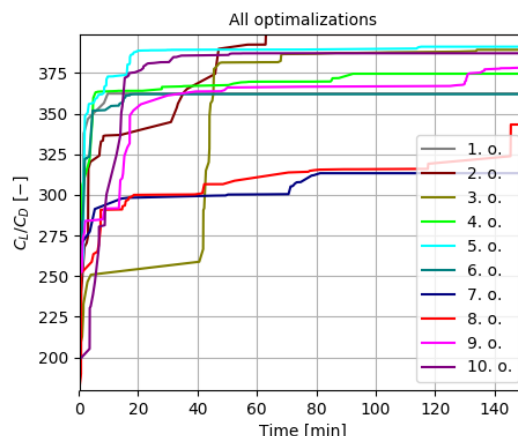
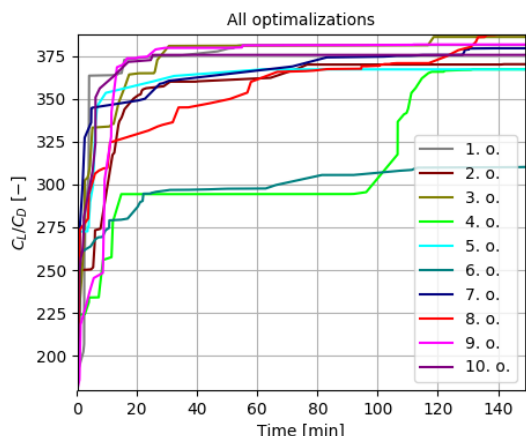


Tab.20: Pro Mann-Whitneyův test ocenění a) ω -0.3328 a c_2 2.8446, b) ω -0.3085 a c_2 2.0273. Obr. 77: Medián ω -0.3328, c_2 2.8446 a ω -0.3085, c_2 2.0273.

Optimalizace	Mann-Whitneyův test	Maximální medián
aPSO ω -0.3328 a c_2 2.8446	p-hodnota 0.18588 > 0.05 výsledky nejsou signifikantně rozdílné	377.65080
aPSO ω -0.3085 a c_2 2.0273		369.59551

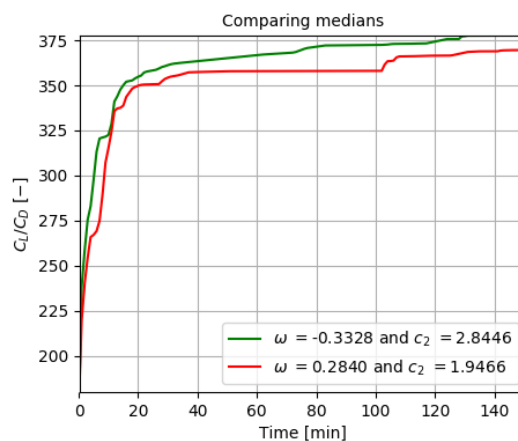
Tab. 21: Výsledek srovnání aPSO ω -0.3328, -0.3085 a c_2 2.8446, 2.0273.

14.2.2 Omega -0.3328 a 0.2840, c2 2.8446 a 1.9466



Obr. 78: Všechny aPSO ω -0.3328 a c_2 2.8446. Obr. 79: Všechny aPSO ω -0.2840 a c_2 1.9466.

a)	381.51394	b)	362.41758
	370.27290		398.80899
	386.16883		389.36957
	367.29572		374.59615
	367.24528		391.26338
	310.23810		362.05128
	379.58167		313.40062
	387.66292		344.17582
	381.78261		378.35073
	375.71992		387.21854

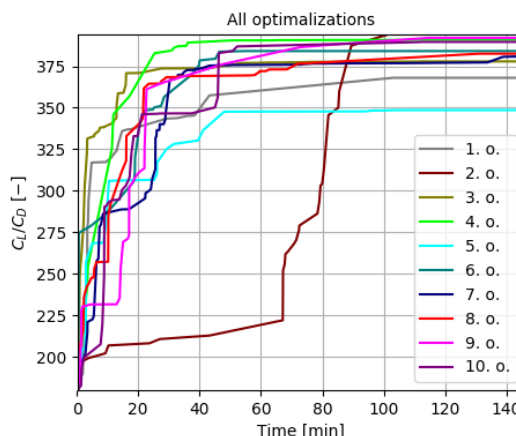
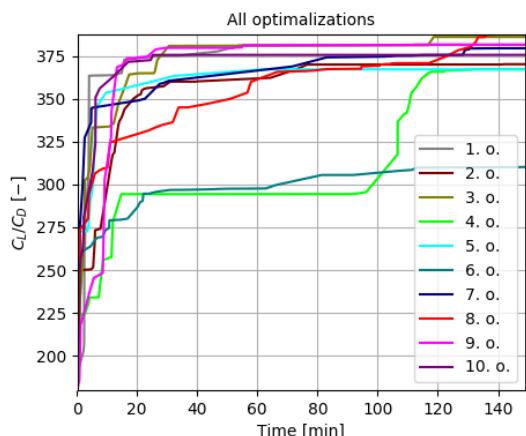


Tab.22: Pro Mann-Whitneyův test ocenění a) ω -0.3328 a c_2 2.8446, b) ω 0.2840 a c_2 1.9466. Obr. 80: Medián ω -0.3328, c_2 2.8446 a ω 0.2840, c_2 1.9466.

Optimalizace	Mann-Whitneyův test	Maximální medián
aPSO ω -0.3328 a c_2 2.8446	p-hodnota 0.90972 > 0.05 výsledky nejsou signifikantně rozdílné	377.65080
aPSO ω -0.2840 a c_2 1.9466		369.59551

Tab. 23: Výsledek srovnání aPSO ω -0.3328, 0.2840 a c_2 2.8446, 1.9466.

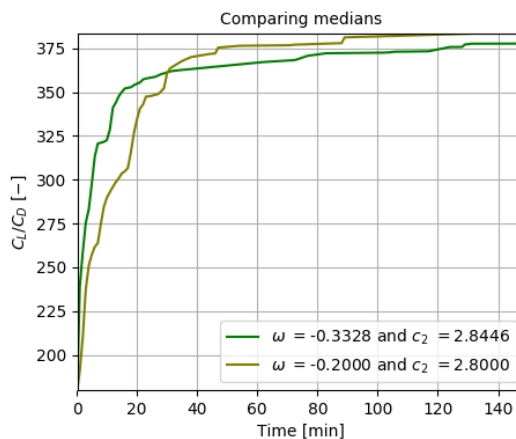
14.2.3 Omega -0.3328 a -0.2, c2 2.8446 a 2.8



Obr. 81: Všechny aPSO ω -0.3328 a c_2 2.8446.

Obr. 82: Všechny aPSO ω -0.2 a c_2 2.8.

a)	381.51394	b)	368.05169
	370.27290		394.18491
	386.16883		377.93419
	367.29572		391.08696
	367.24528		348.60952
	310.23810		384.17308
	379.58167		381.63968
	387.66292		382.58799
	381.78261		392.08333
	375.71992		389.70149



Tab. 24: Pro Mann-Whitneyův test ocenění a) Obr. 83: Medián ω -0.3328, c_2 2.8 a ω -0.2, c_2 ω -0.3328 a c_2 2.8446, b) ω 0.2000 a c_2 2.8.

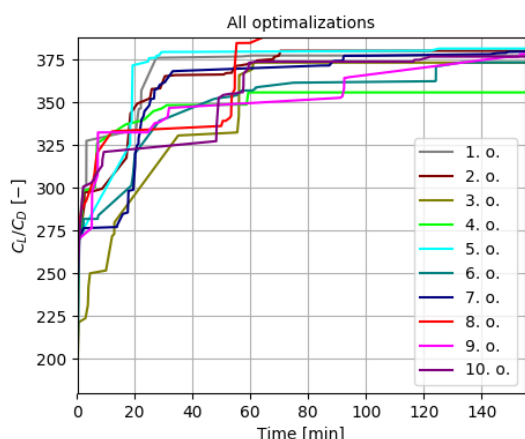
Optimalizace	Mann-Whitneyův test	Maximální medián
aPSO ω -0.3328 a c_2 2.8446	p-hodnota 0.10410 > 0.05 výsledky nejsou signifikantně rozdílné	377.65080
aPSO ω -0.2 a c_2 2.8		381.38053

Tab. 25: Výsledek srovnání aPSO ω -0.3328, -0.2 a c_2 2.8446, 2.8.

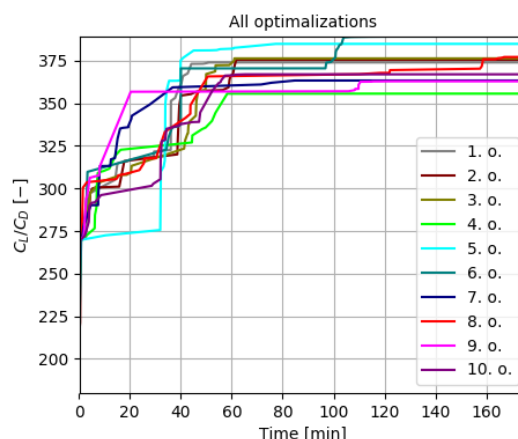
14.3 Výsledky optimalizace PHO

Byl vytvořen zcela nový a původní typ paralelní hybridní optimalizace, založený na nové optimalizaci profilu evolučními algoritmy aEA a nové optimalizaci profilu hejnem částic aPSO. Díky spojení obou optimalizací dochází také ke spojení jejich výhod a díky paralelizmu je zaručena diverzita nového typu optimalizace.

14.3.1 Střídání aEA a aPSO po 15 a 30 minutách



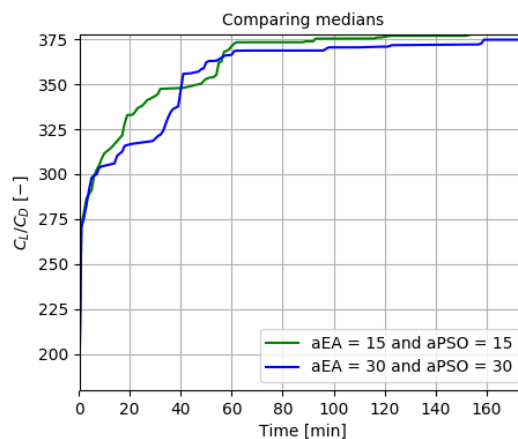
Obr. 84: Všechny PHO střídání po 15 min.



Obr. 85: Všechny PHO střídání po 30 min.

a)	377.28421	b)	374.01656
	380.05792		375.57522
	373.13152		376.38298
	355.72202		355.67460
	381.35827		384.90486
	373.27345		389.20354
	379.50397		363.38596
	387.86571		377.57261
	378.18381		362.92627
	376.66008		367.02752

Tab. 26: Pro Mann-Whitneyův test ocenění PHO střídání po a) 10, b) 30 min.

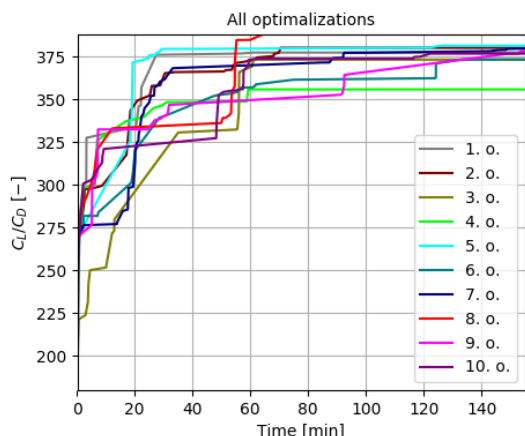


Obr. 86: Medián PHO střídání po 15, 30 min..

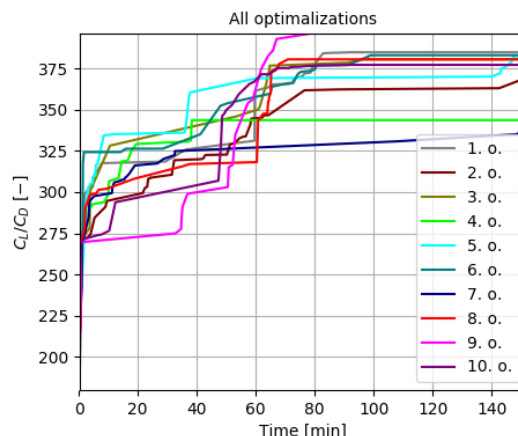
Optimalizace	Mann-Whitneyův test	Maximální medián
PHO střídání po 15 min.	p-hodnota 0.30750 > 0.05 výsledky nejsou signifikantně rozdílné	377.73401
PHO střídání po 30 min.		374.79589

Tab. 27: Výsledek srovnání PHO při střídání aEA a aPSO po 15 a 30 minutách.

14.3.2 Střídání aEA a aPSO po 15 a 45 minutách

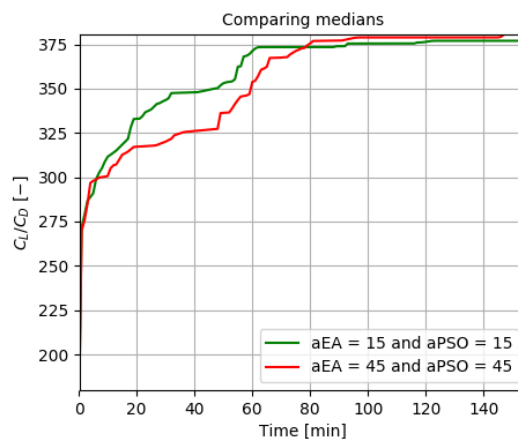


Obr. 87: Všechny PHO střídání po 15 min.



Obr. 88: Všechny PHO střídání po 45 min.

a)	377.28421	b)	384.83516
	380.05792		369.18447
	373.13152		380.57026
	355.72202		343.64522
	381.35827		382.53112
	373.27345		383.05882
	379.50397		339.01716
	387.86571		380.55344
	378.18381		396.20915
	376.66008		377.20879

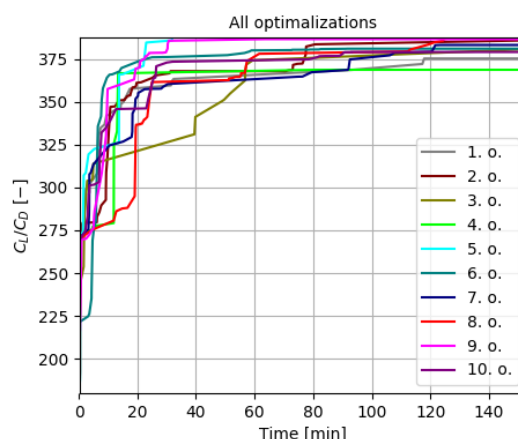
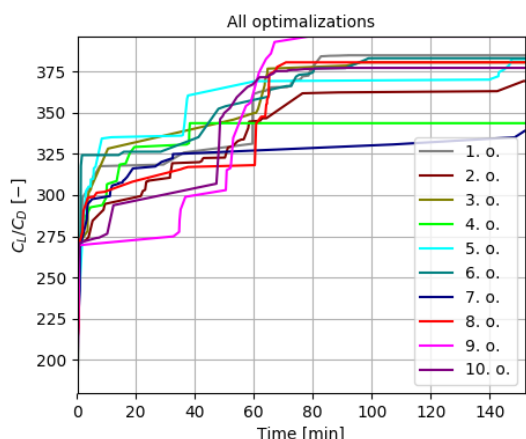


Tab. 28: Pro Mann-Whitneyův test ocenění PHO střídání po a) 10, b) 45 min. Obr. 89: Medián PHO střídání po 15, 45 min..

Optimalizace	Mann-Whitneyův test	Maximální medián
PHO střídání po 15 min.	p-hodnota 0.57075 > 0.05 výsledky nejsou signifikantně rozdílné	377.73401
PHO střídání po 45 min.		380.56185

Tab. 29: Výsledek srovnání PHO při střídání aEA a aPSO po 15 a 30 minutách.

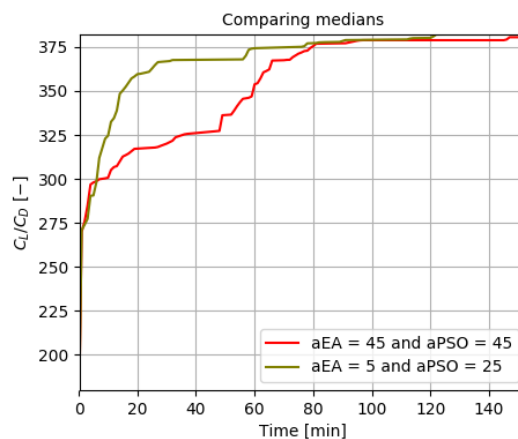
14.3.3 Střídání aEA a aPSO po 45 minutách a aEA s 5 min. a aPSO s 25 min.



Obr. 90: Všechny PHO střídání po 45 min.

Obr. 91: Všechny PHO střídání po 5 a 25 min.

a)	384.83516	b)	375.32164
	369.18447		385.89744
	380.57026		380.73123
	343.64522		368.72659
	382.53112		387.63043
	383.05882		381.06509
	339.01716		383.22034
	380.55344		386.99561
	396.20915		386.63327
	377.20879		379.13934



Tab. 30: Pro Mann-Whitneyův test ocenění PHO střídání po a) 10, b) 45 min.

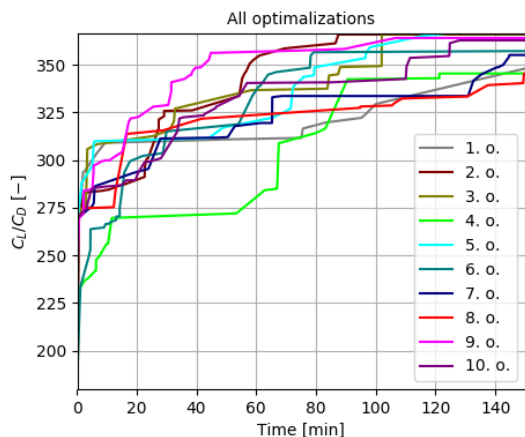
Obr. 92: Medián PHO střídání po 15, 45 min..

Optimalizace	Mann-Whitneyův test	Maximální medián
PHO střídání po 45 min.	p-hodnota 0.27303 > 0.05 výsledky nejsou signifikantně rozdílné	380.56185
PHO střídání po 5 a 25 min.		382.14271

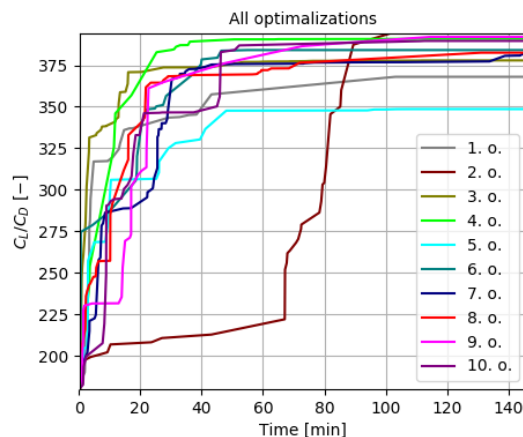
Tab. 31: Výsledek střídání aEA a aPSO po 45 a aEA po 5 a aPSO po 25 minutách.

14.4 Srovnání výsledků optimalizačních algoritmů

14.4.1 Srovnání výsledků aEA a aPSO



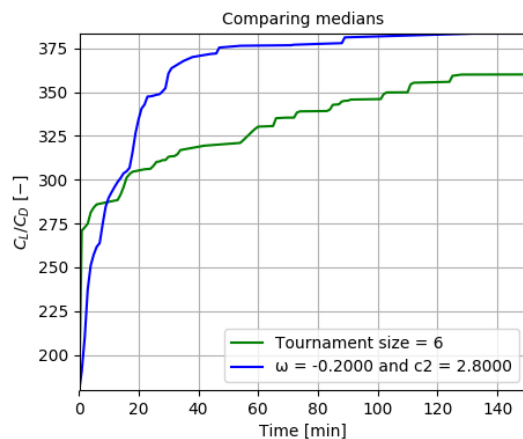
Obr. 93: Všechny aEA velikost turnaje 6.



Obr. 94: Všechny aPSO ω -0.2 a c_2 2.8.

a)	347.94007	b)	368.05169
	365.94810		394.18491
	366.17391		377.93419
	345.48223		391.08696
	366.47541		348.60952
	357.21030		384.17308
	355.15991		381.63968
	345.43520		382.58799
	364.08624		392.08333
	362.84861		389.70149

Tab. 32: Pro Mann-Whitneyův test ocenění a) aEA velikost turnaje 6, b) aPSO ω -0.2 a c_2 2.8

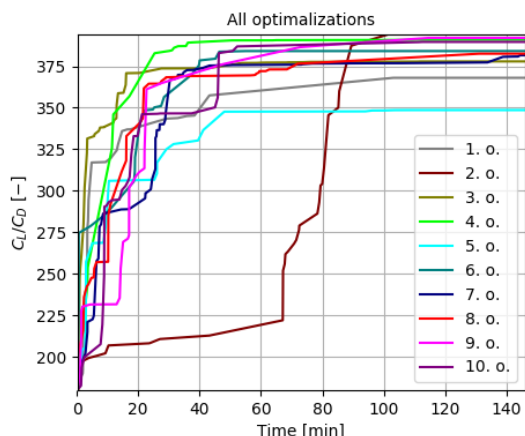


Obr. 95: Medián aEA velikost turnaje 6 a aPSO ω -0.2 a c_2 2.8.

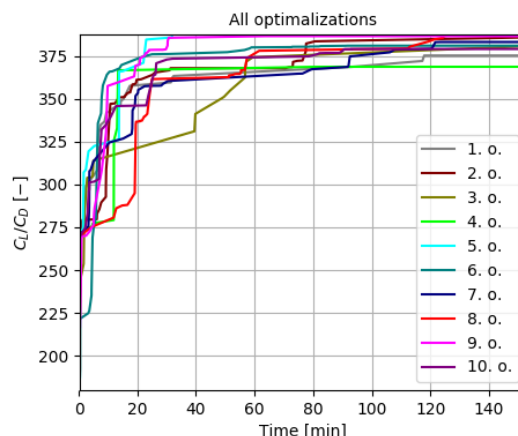
Optimalizace	Mann-Whitneyův test	Maximální medián
aEA s turnajem o 6 jedincích	p-hodnota 0.00132 < 0.05 výsledky jsou signifikantně rozdílné	360.02945
aPSO ω -0.2 a c_2 2.8		381.38053

Tab. 33: Výsledek srovnání aEA s velikostí turnaje 6 jedinců a aPSO ω -0.2, c_2 2.8.

14.4.2 Srovnání výsledků aPSO a PHO



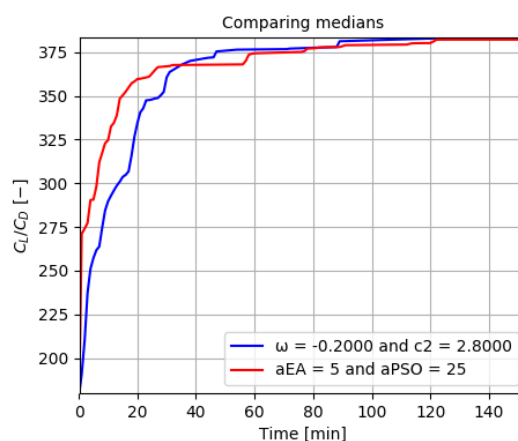
Obr. 96: Všechny aPSO ω -0.2 a c_2 2.8.



Obr. 97: Všechny PHO střídání po 5 a 25 min.

a)	368.05169	b)	375.32164
	394.18491		385.89744
	377.93419		380.73123
	391.08696		368.72659
	348.60952		387.63043
	384.17308		381.06509
	381.63968		383.22034
	382.58799		386.99561
	392.08333		386.63327
	389.70149		379.13934

Tab. 34: Pro Mann-Whitneyův test ocenění a) aPSO ω -0.2 a c_2 2.8, b) PHO střídání po 5 a 25 min.



Obr. 98: Medián aPSO ω -0.2 a c_2 2.8, PHO střídání po 5 a 25 min.

Optimalizace	Mann-Whitneyův test	Maximální medián
aPSO ω -0.2 a c_2 2.8	p-hodnota 0.57075 > 0.05 výsledky nejsou signifikantně rozdílné	381.38053
PHO střídání po 5 a 25 min.		382.14271

Tab. 35: Výsledek srovnání aPSO ω -0.2, c_2 2.8 a PHO, aEA po 5 a aPSO po 25 min.

14.4.3 Shrnutí srovnání výsledků optimalizací

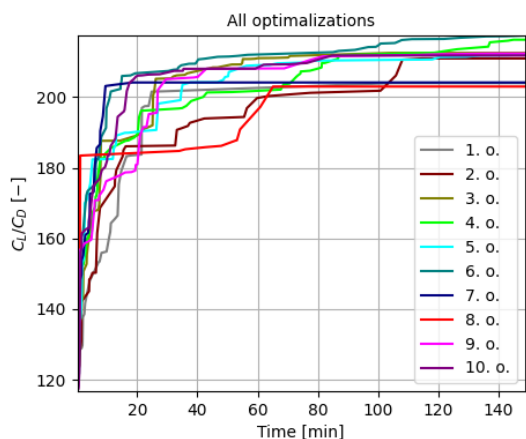
Ze srovnání výsledků aEA a aPSO jednoznačně vyplývá, že jsou signifikantně rozdílné, diametrálně lepších výsledků dosahuje aPSO. Optimalizace aPSO a PHO dává téměř stejné výsledky, je tudíž na zvážení, kterou ze zmíněných optimalizací použít. Byla nakonec vybrána optimalizace PHO pro rychlejší průběh počátku optimalizace.

15 Výsledky autokonfiguratůru IRACE

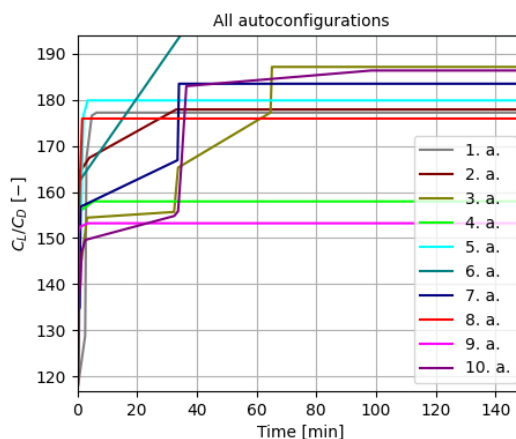
15.1 Výsledky autokonfigurace parametrizací profilu

Díky použití parametrizace u optimalizace profilu lze použít autokonfiguratůru namísto optimalizace. Lze použít autokonfiguratůru IRACE jak s parametrizací B-spline tak i Bezier-PARSEC 3434. Při absenci parametrizací by nebylo možné použít autokonfiguratůru místo optimalizací.

15.1.1 Srovnání výsledků aPSO a IRACE použitého na parametrizaci B-spline



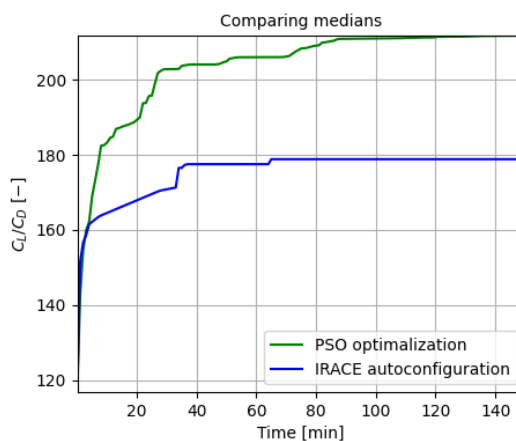
Obr. 99: Všechny aPSO optimalizace.



Obr. 100: Všechny IRACE autokonfigurace.

a)	203.92752	b)	177.21374
	210.90575		177.90609
	212.47880		187.13717
	216.21261		157.95649
	211.96095		179.86773
	217.41962		193.96253
	204.11270		183.48438
	202.95546		175.92931
	212.28416		153.21399
	211.75154		186.34922

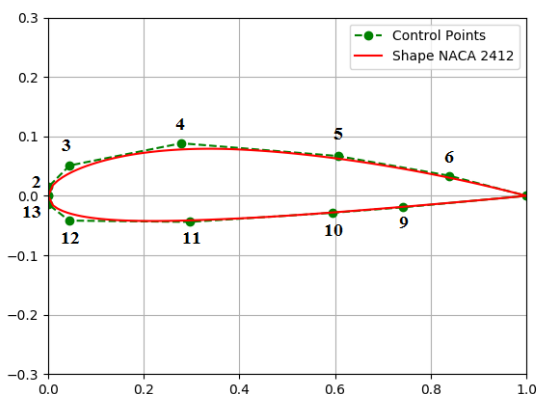
Tab. 36: Pro Mann-Whitneyův test ocenění
a) aPSO a b) IRACE.



Obr. 101: Medián aPSO a IRACE.

Srovnání	Mann-Whitneyův test	Maximální medián
aPSO optimalizace	p-hodnota 0.00018 < 0.05 výsledky jsou signifikantně rozdílné	211.85624
IRACE autokonfigurace		178.88691

Tab. 37: Srovnání výsledků aPSO optimalizace a IRACE autokonfiguratůru.



```

p2 "--p2 " r (0.00743, 0.0263)
p3 "--p3 " r (0.02551, 0.0965)
p4 "--p4 " r (0.06192, 0.2415)
p5 "--p5 " r (0.04685, 0.1070)
p6 "--p6 " r (0.02704, 0.05556)
p9 "--p9 " r (-0.0188, -0.0120)
p10 "--p10 " r (-0.12421, -0.01019)
p11 "--p11 " r (-0.12707, -0.01373)
p12 "--p12 " r (-0.0429, -0.0103)
p13 "--p13 " r (-0.0291, -0.0090)

```

dolní meze
horní meze

Obr. 102: Optimalizované řídicí body profilu. Obr. 103: IRACE parametry B-spline profilu.

Faktory ovlivňující vypočítané výsledky optimalizace aPSO a autokonfiguratoru IRACE:

- IRACE byl atypicky použit namísto optimalizace pro parametrizaci, čehož lze využít pouze při prezenci parametrizace, tedy například u řešení aerodynamických vlastností profilu,
- IRACE je nástroj pro řešení obecných problémů přes parametry, zatímco optimalizace je speciálně používána pro řešení konkrétního problému, tudíž by optimalizace měla dávat lepší výsledky,
- nevýhodou použití IRACE je, že meze parametrů, tedy řídicích bodů, jsou pevně zadány, zatímco při optimalizaci profilu hejnem částic aPSO se při každé iteraci mění vertikální pozice řídicích bodů tak i meze těchto bodů,
- důležité je zadání mezí parametrů, tedy řídicích bodů, u IRACE, což ovlivňuje negativně výsledky při nesprávném zadání a při správném pozitivně. Dále se podílí na rychlosti konvergence k optimu.

15.1.2 Srovnání výsledků IRACE použitého na parametrizaci Bezier-PARSEC

Nelze použít autokonfiguratoru IRACE na parametrizaci Bezier-PARSEC 3434 z důvodu nemožnosti použít v zadání parametrů odkazy na předešlé parametry (dependent domains) a nemožnost použití matematických funkcí (viz obr. 104). Zmíněné možnosti by měly být implementovány [143] v další verzi nástroje IRACE.

```
p1 "--p1 " r (0.2, 0.5)
p2 "--p2 " r (0.1, 0.2)
p3 "--p3 " r (-0.1, "-p2")
p4 "--p4 " r (0.001, "min(p2, (-2*p3*p1/3)^0.5)")
p5 "--p5 " r ("p1", 0.7)
p6 "--p6 " r ("min(atan2(p2, 1.0 - p5), 0.0872665)", "atan2(p2, 1.0 - p5)")
p7 "--p7 " r (0.2, 0.5)
p8 "--p8 " r (0.0, 0.2)
p9 "--p9 " r (0.001, "p7")
p10 "--p10 " r (0.001, "p9")
p11 "--p11 " r ("atan2(p8, p9)", "atan2(p8, p10)")
p12 "--p12 " r ("p7", 0.95)
p13 "--p13 " r (0.261799, "atan2(p8, 1.0 - p12)")
```

dependent domains
math functions

Obr. 104: Parametry Bezier-PARSEC 3434 pro IRACE 3.5.

15.2 Výsledky autokonfigurace optimalizací

Pokud nejsou k dispozici parametrizace tak je použito pro řešení optimalizací a autokonfigurator je využit v typičtější roli jako je meta-optimalizace. Výhodou je, že může optimalizovat několik problémů zároveň. Typické je, že meta-optimalizace optimalizací řeší pouze jeden problém.

Pro příklad, pokud je hledána neoptimálnější velikost turnaje mezi třemi velikostmi turnajů, tak je třikrát spuštěna sada optimalizací pokaždé s jinou velikostí turnaje. Přičemž zadání velikostí turnajů je odhadem. Výsledky se porovnají a vybere se nejlepší velikost turnaje. Tento postup řeší na každý problém jedno spuštění sady optimalizací.

Pokud je hledána optimální velikost turnaje přes autokonfigurator, tak mohou být zadány najednou všechny testované velikosti turnajů. To znamená, že na jedno spuštění autokonfiguratoru může být řešeno více problémů. Stejně tak není nutné explicitně uvést velikosti turnajů, je možné uvést první a třetí velikost turnaje a autokonfigurator vybere nejlepší velikost turnaje v rozmezí první a třetí velikosti.

16 Výpočet optimálních aerodynamických vlastností

16.1 Výpočet aerodynamických vlastností

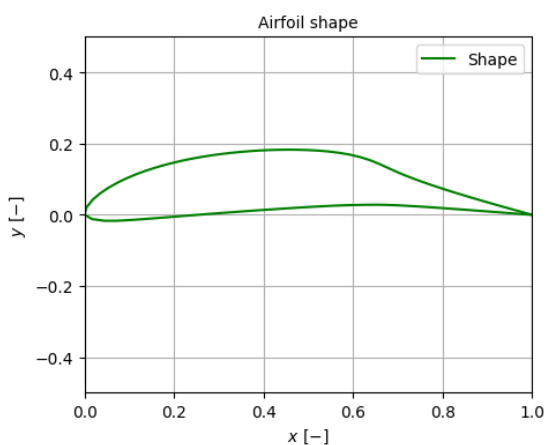
Pro spočítání aerodynamických vlastností profilu byl použit výpočetní program Xfoil [144].

16.1.1 Výpočet extrému aerodynamické jemnosti

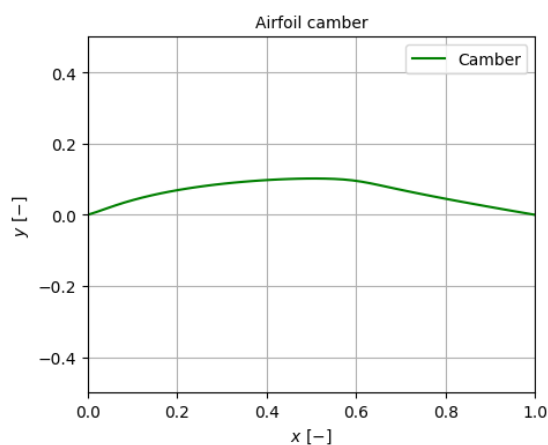
Na základě srovnání výsledků optimalizace je pro výpočty aerodynamických vlastností vybrána paralelní hybridní optimalizace PHO. Je použito parametrizace Bezier-PARSEC 3434 s náhodnými parametry při inicializaci. Ocenění účelové funkce je aerodynamická jemnost profilu. Hledá se maximální hodnota aerodynamické jemnosti, tj. extrém.

Mach: 0.000	Re: 5.0e+06	α_{\min} : -15.00°	α_{\max} : 15.00°	α_{step} : 0.25°
α : 7.00°	C_L : 1.9056	C_D : 0.00481	C_L/C_D : 396.1746	C_M : -0.2404

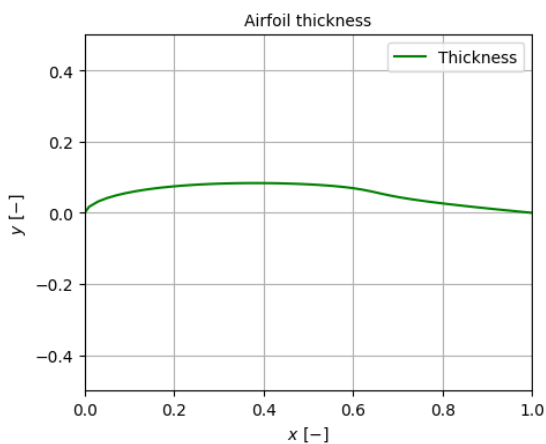
Tab. 38: Vypočítané aerodynamické vlastnosti pro maximální aerodynamickou jemnost.



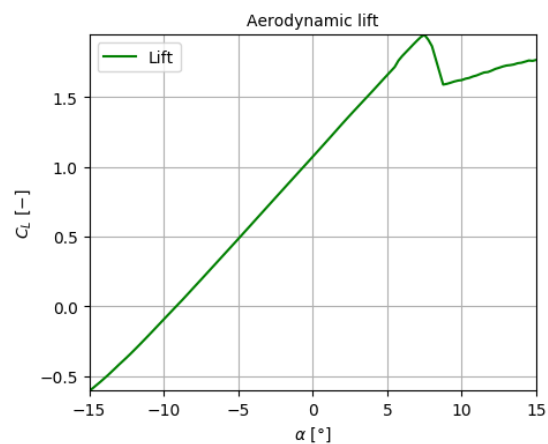
Obr. 105: Tvar profilu.



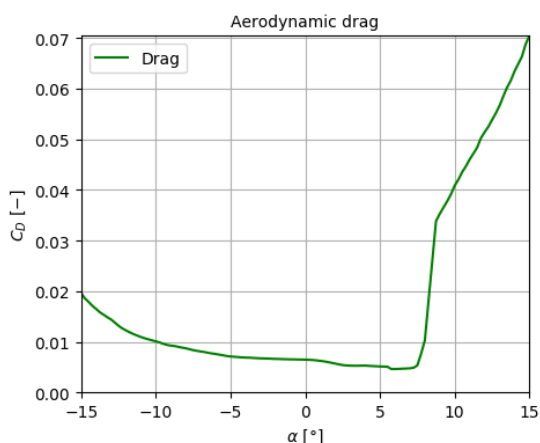
Obr. 106: Střední křivka profilu.



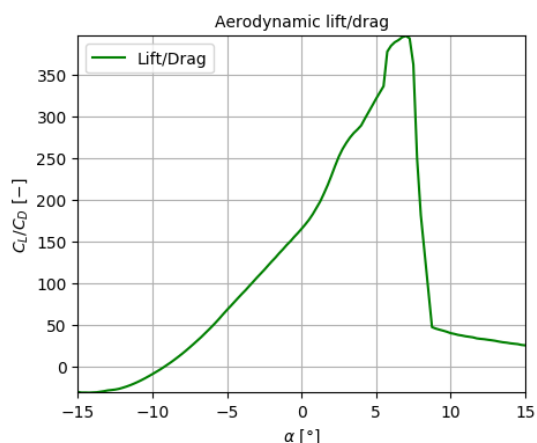
Obr. 107: Tloušťka profilu.



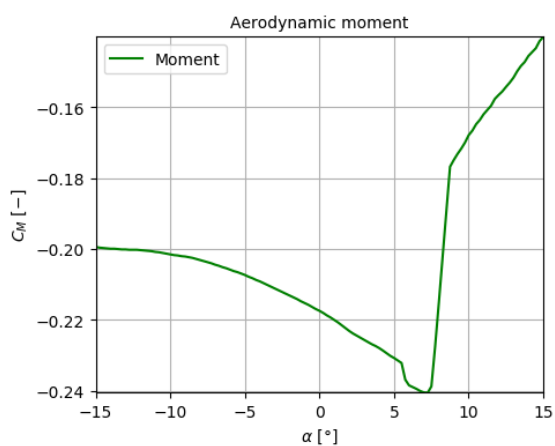
Obr. 108: Vztlaková křivka profilu.



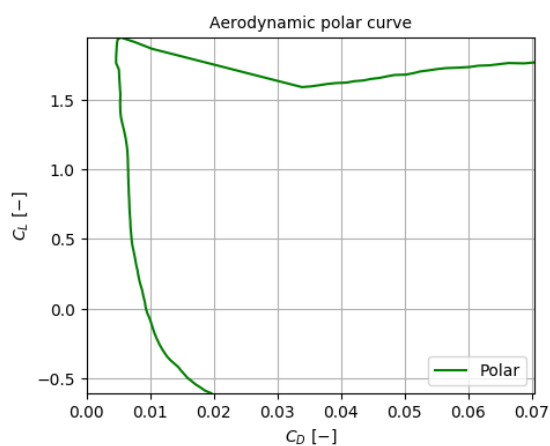
Obr. 109: Odporová křivka profilu.



Obr. 110: Aerodynamická jemnost profilu.



Obr. 111: Momentová křivka profilu.



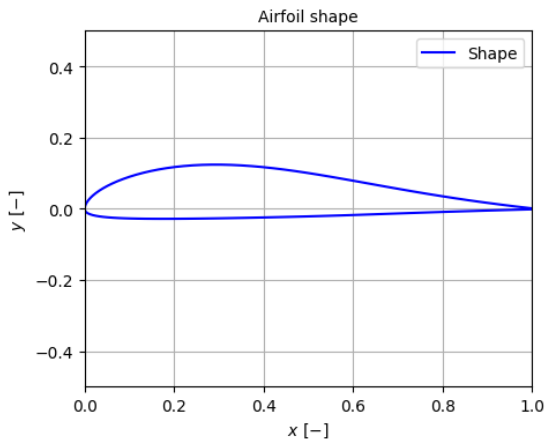
Obr. 112: Polára profilu.

16.1.2 Výpočet průměru aerodynamické jemnosti při ustáleném letu

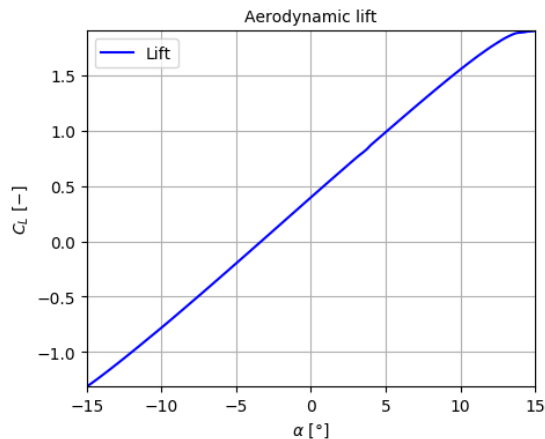
Širší ustálený let je definován rozsahem úhlů náběhu, v tomto případě $3^\circ - 9^\circ$. Pro výpočty aerodynamických vlastností je použita optimalizace profilu hejnem částic s výchozím profilem, v tomto případě profilem NACA 2412. Základem je optimalizace řídicích bodů počáteční B-spline křivky profilu. Účelová funkce hledá maximální průměr aerodynamické jemnosti profilu v rozsahu širšího ustáleného letu.

Mach: 0.000	Re: 5.0e+06	α_{\min} : -15.00°	α_{\max} : 15.00°	α_{step} : 0.25°
α : 3.00° - 9.00°	C_L : 1.0997	C_D : 0.00600	C_L/C_D : 183.2631	C_M : -0.0556

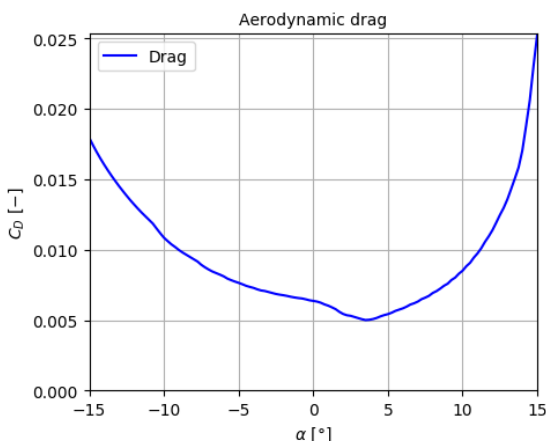
Tab. 39: Vypočítané aerodynamické vlastnosti pro průměrnou aerodynamickou jemnost.



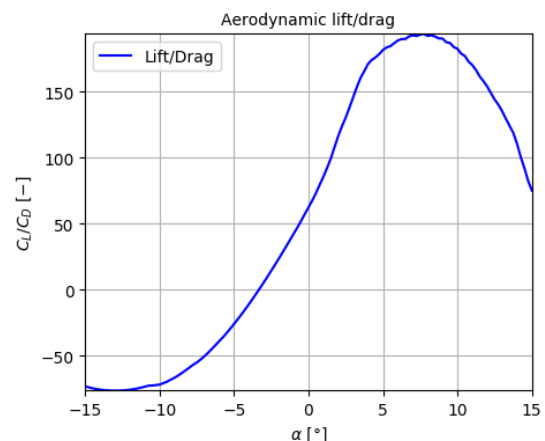
Obr. 113: Tvar profilu.



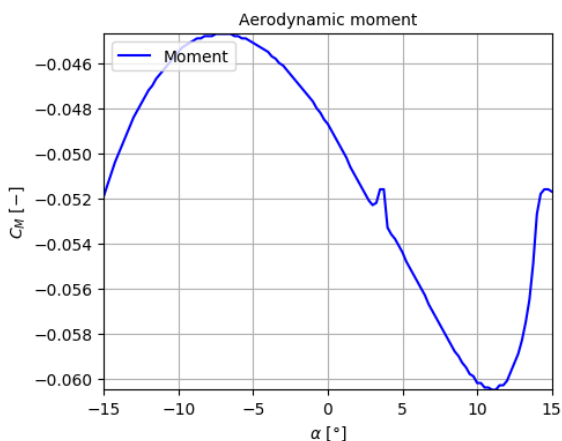
Obr. 114: Vztlaková křivka profilu.



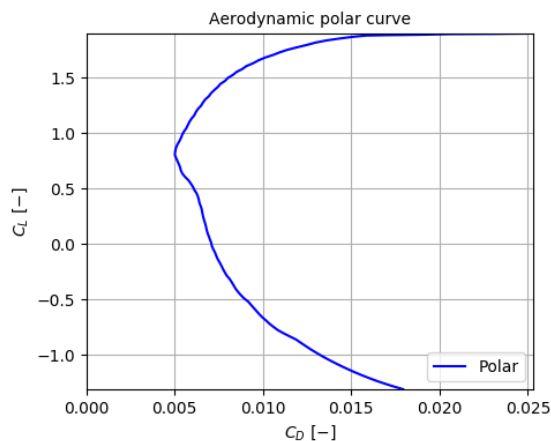
Obr. 115: Odporová křivka profilu.



Obr. 116: Aerodynamická jemnost profilu.



Obr. 117: Momentová křivka profilu.



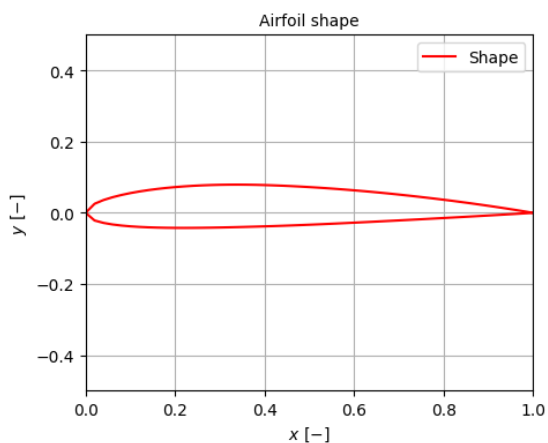
Obr. 118: Polára profilu.

16.1.3 Profil NACA 2412 a jeho aerodynamické vlastnosti

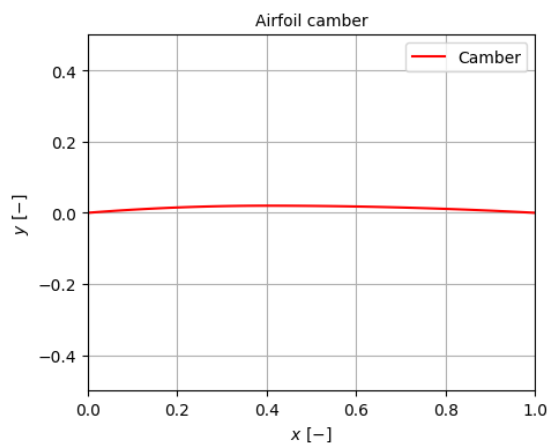
Profil NACA 2412 slouží ke srovnání vypočítaného extrému nové paralelní hybridní optimalizace PHO, stejně tak slouží jako počáteční profil pro optimalizaci hejnem částic s výchozím profilem.

Mach: 0.000	Re: 5.0e+06	α_{\min} : -15.00°	α_{\max} : 15.00°	α_{step} : 0.25°
-------------	-------------	---------------------------	--------------------------	--------------------------------

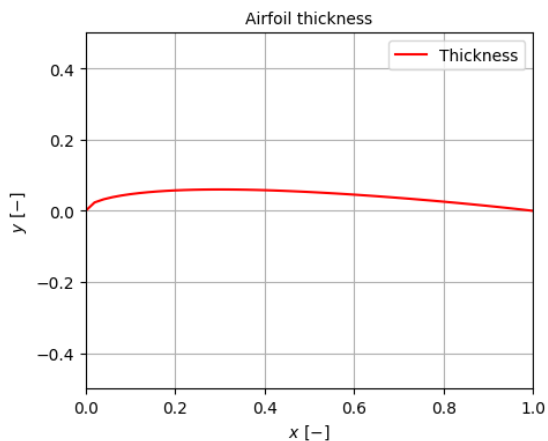
Tab. 40: Profil NACA 2412 sloužící ke srovnání optimalizačních výsledků.



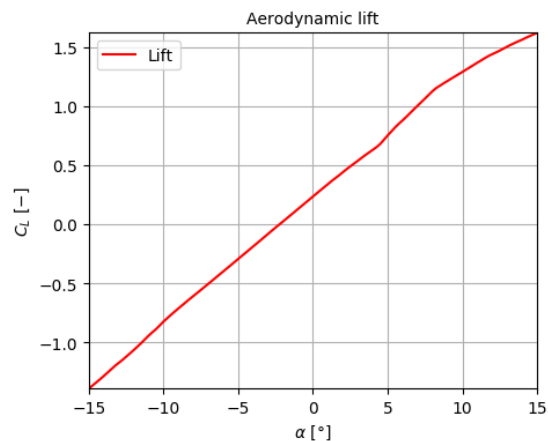
Obr. 119: Tvar profilu.



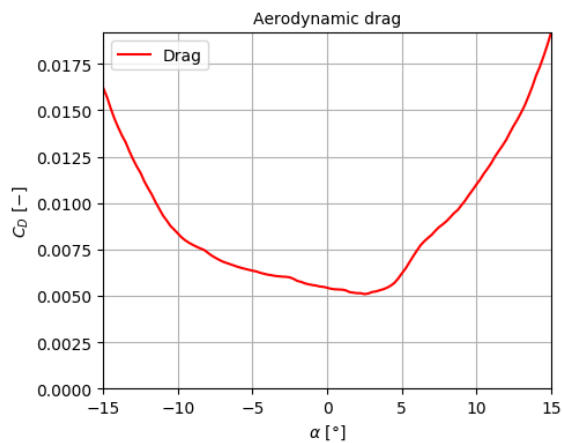
Obr. 120: Střední křivka profilu.



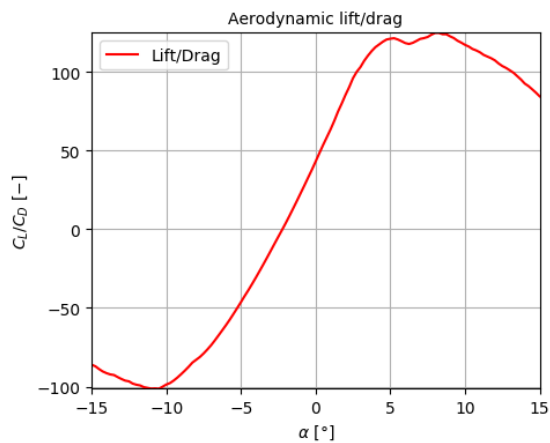
Obr. 121: Tloušťka profilu.



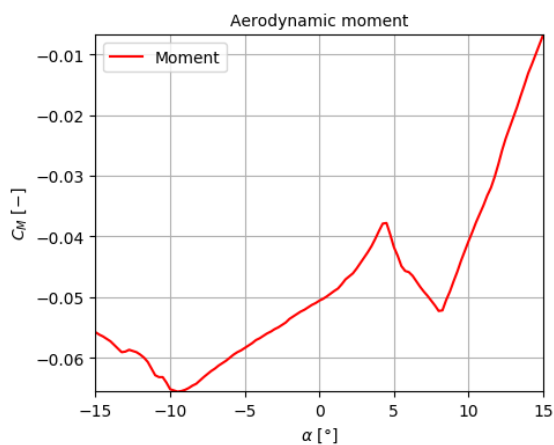
Obr. 122: Vztlaková křivka profilu.



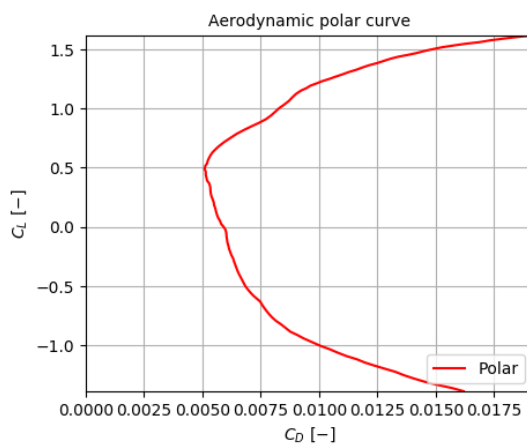
Obr. 123: Odporová křivka profilu.



Obr. 124: Aerodynamická jemnost profilu.



Obr. 125: Momentová křivka profilu.



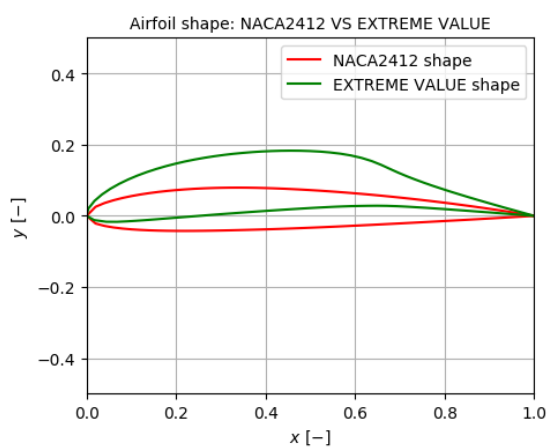
Obr. 126: Polára profilu.

16.2 Srovnání aerodynamických vlastností

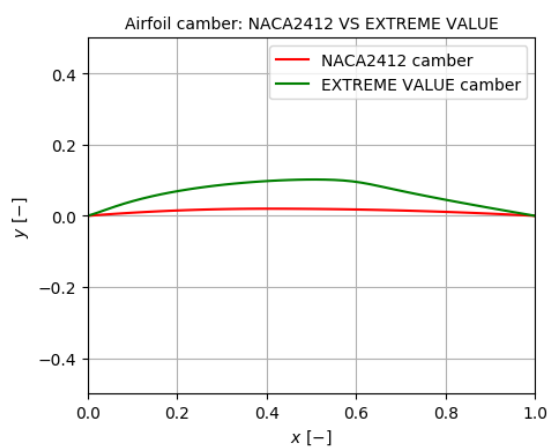
16.2.1 Srovnání extrém aerodynamické jemnosti s NACA 2412

Mach: 0.000	Re: 5.0e+06	α_{\min} : -15.00°	α_{\max} : 15.00°	α_{step} : 0.25°
α : 8.25°	C_L : 1.1542	C_D : 0.00924	C_L/C_D : 124.9134	C_M : -0.0522
α : 7.00°	C_L : 1.9056	C_D : 0.00481	C_L/C_D : 396.1746	C_M : -0.2404

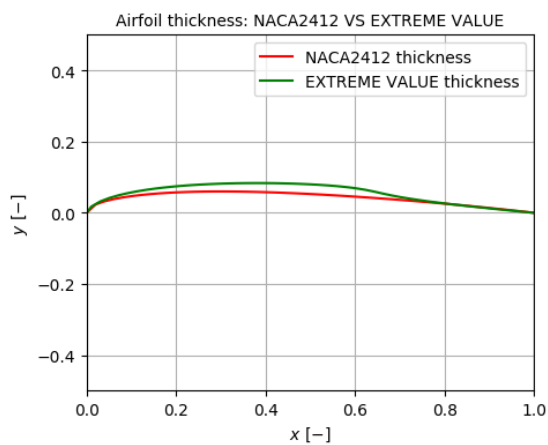
Tab. 41: Srovnání aerodynamických vlastností pro extrém aerodynamické jemnosti.



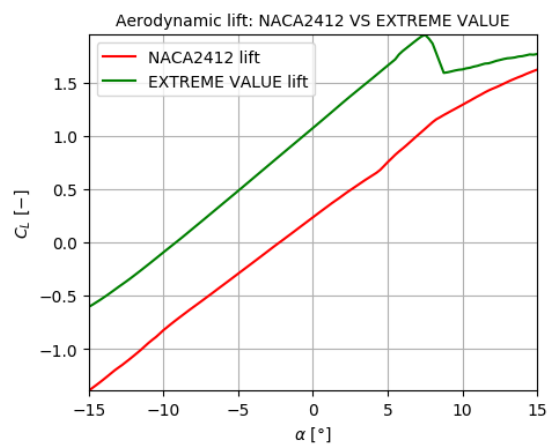
Obr. 127: Tvar profilu.



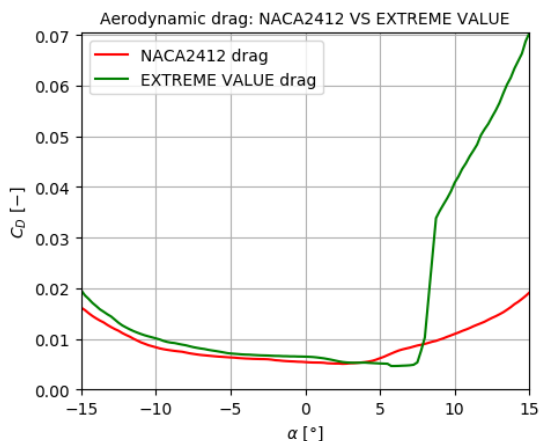
Obr. 128: Střední křivka profilu.



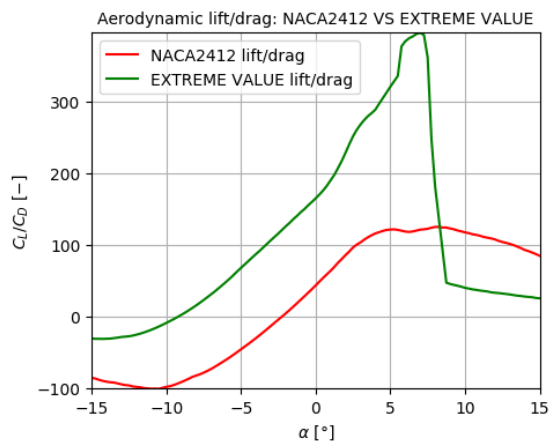
Obr. 129: Tloušťka profilu.



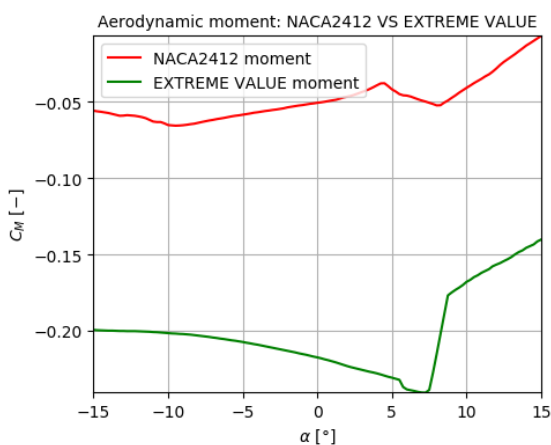
Obr. 130: Vztlaková křivka profilu.



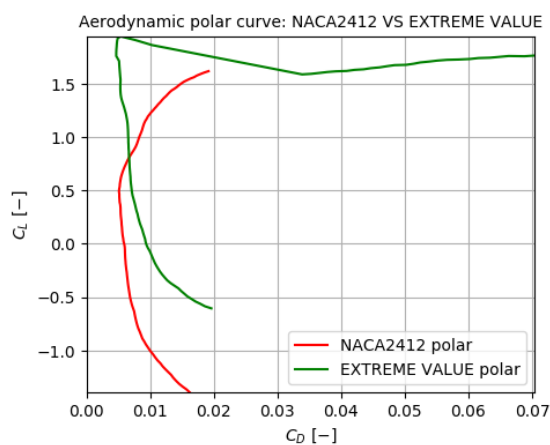
Obr. 131: Odporová křivka profilu.



Obr. 132: Aerodynamická jemnost profilu.



Obr. 133: Momentová křivka profilu.



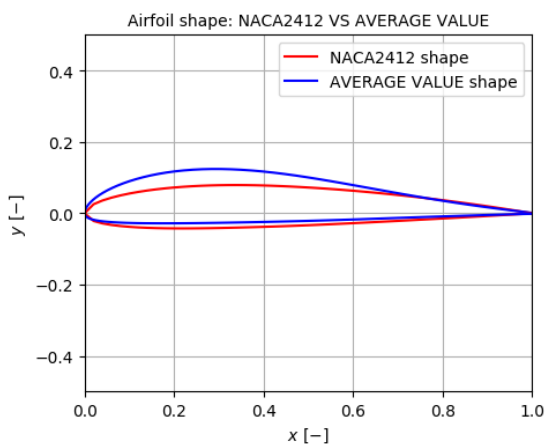
Obr. 134: Polára profilu.

16.2.2 Srovnání průměru aerodynamické jemnosti při ustáleném letu s NACA 2412

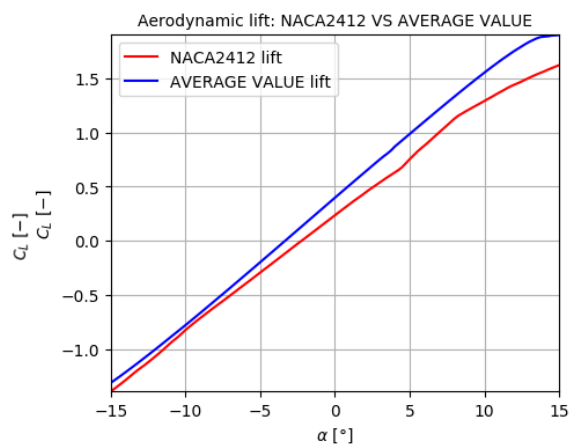
Širší ustálený let je definován rozsahem úhlů náběhu, v tomto případě 3° - 9°.

Mach: 0.000	Re: 5.0e+06	α_{\min} : -15.00°	α_{\max} : 15.00°	α_{step} : 0.25°
α : 3.00° - 9.00°	C_L : 0.8779	C_D : 0.00732	C_L/C_D : 119.8291	C_M : -0.0456
α : 3.00° - 9.00°	C_L : 1.0997	C_D : 0.00600	C_L/C_D : 183.2631	C_M : -0.0556

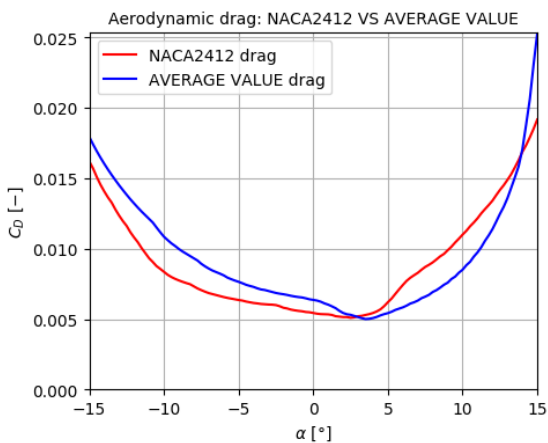
Tab. 42: Srovnání aerodynamických vlastností pro průměr aerodynamické jemnosti.



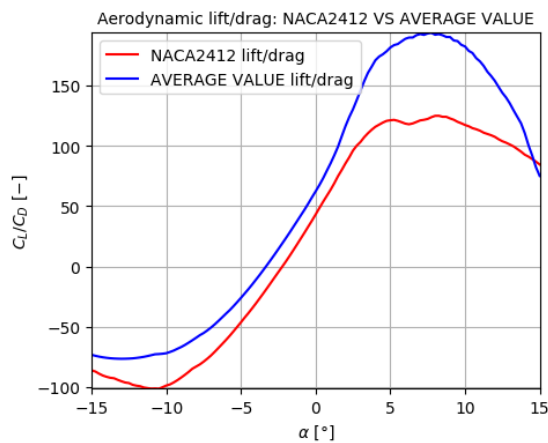
Obr. 135: Tvar profilu.



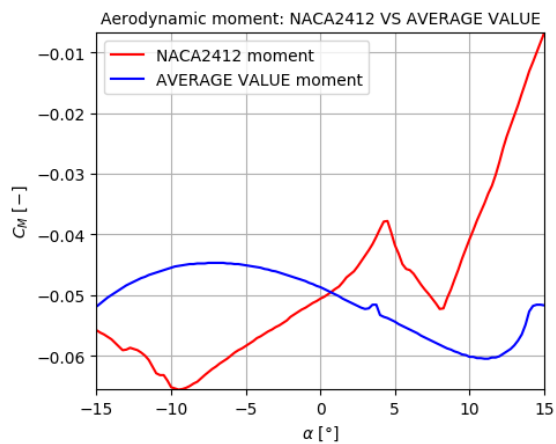
Obr. 136: Vztlaková křivka profilu.



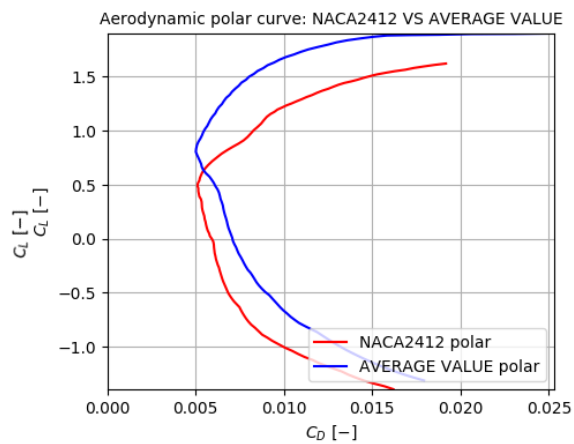
Obr. 137: Odporová křivka profilu.



Obr. 138: Aerodynamická jemnost profilu.



Obr. 139: Momentová křivka profilu.



Obr. 140: Polára profilu.

17 Výsledky práce

17.1 Srovnání výsledků optimalizačních algoritmů

Pro optimalizaci profilu evolučními algoritmy při velikosti populace 25 vychází nejlépe selekce turnajem o velikosti 6. Křížení jedinců je pozměněno pro použití s parametrizací Bezier-PARSEC 3434. Nicméně výsledky této optimalizace vůči ostatním optimalizačním algoritmům nevyhází nejlépe.

Optimalizace	Mann-Whitneyův test	Maximální medián
aEA s turnajem o 6 jedincích	p-hodnota 0.00132 < 0.05 výsledky jsou signifikantně rozdílné	360.02945
aPSO ω -0.2 a c_2 2.8		381.38053

Tab. 43: Výsledek srovnání aEA s velikostí turnaje 6 jedinců a aPSO ω -0.2, c_2 2.8.

Lepších výsledků než při optimalizaci profilu evolučními algoritmy dosahují optimalizace profilu hejnem částic (viz tab. 43), kdy pro velikost hejna 25 dochází k nejlepším výsledkům při váze setrvačnosti ω -0.2 a akceleračního koeficientu c_2 2.8. Optimalizace jsou signifikantně rozdílné. Optimalizace hejnem je založena na variantě optimalizace Many Optimizing Liaisons, která byla také pozměněna pro použití parametrizace Bezier-PARSEC 3434.

Nová paralelní hybridní optimalizace PHO využívá optimalizací aEA a aPSO, kdy nejlepší výsledky dává při střídání obou optimalizací. Nejlepšího výsledku bylo dosaženo střídáním slabší optimalizace profilu evolučními algoritmy po 5 minutách a střídáním lepší optimalizaci profilu hejnem částic po 25 minutách.

Optimalizace	Mann-Whitneyův test	Maximální medián
aPSO ω -0.2 a c_2 2.8	p-hodnota 0.57075 > 0.05 výsledky nejsou signifikantně rozdílné	381.38053
PHO střídání po 5 a 25 min.		382.14271

Tab. 44: Výsledek srovnání aPSO ω -0.2, c_2 2.8 a PHO, aEA po 5 a aPSO po 25 min.

Nová paralelní hybridní optimalizace PHO dosahuje podobných výsledků než samostatná optimalizace profilu hejnem částic aPSO, s rychlejší konvergencí na počátku a díky paralelizmu je zaručena diverzita této nové optimalizace. Nakonec byla vybrána optimalizace PHO pro rychlejší průběh počátku optimalizace.

17.2 Srovnání výsledků autokonfigurátoru

U optimalizací řešících aerodynamické vlastnosti profilu byly použity parametrizace profilu. Díky tomu tedy šlo použít autokonfigurátoru IRACE přímo na parametrizaci B-spline.

Srovnání	Mann-Whitneyův test	Maximální medián
aPSO optimalizace	p-hodnota 0.00018 < 0.05 výsledky jsou signifikantně rozdílné	211.85624
IRACE autokonfigurace		178.88691

Tab. 45: Srovnání výsledků aPSO optimalizace a IRACE autokonfigurátoru.

Výsledky (viz tab. 45) byly ovlivněny hlavně hrubým odhadnutím mezí řídicích bodů B-spline parametrizace, dále pak dobou rezervovanou pro běh autokonfiguračního nástroje a zřejmě netypickým použitím autokonfiguratoru namísto optimalizace. Dále jsou zaručeny lepší výsledky, tj. maximální aerodynamická jemnost, u optimalizací, které jsou vhodné pro daný problém, zatímco aplikace autokonfigurátoru je obecnější.

```
# With dependent domains
p1 "" r (0, 100)
p2 "" r ("p1", "p1 + 10")
```

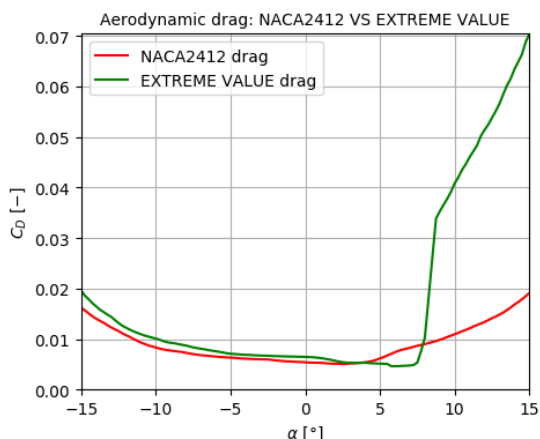
Obr. 141: Odkazy na předchozí parametry (tj. dependent domains) [116].

V budoucnu by bylo možné použít autokonfigurátoru IRACE také na Bezier-PARSEC 3434 parametrizaci, kdy by mělo být možné používat nejdůležitější matematické funkce a také odkazy na předchozí parametry (dependent domains). Oboje je plánováno autorem IRACE pro další verzi.

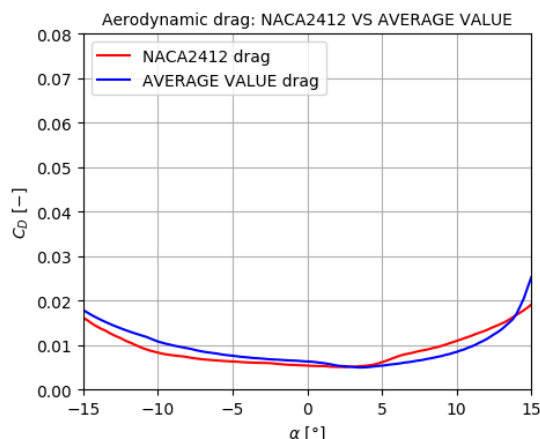
Výhodou by bylo, že na rozdíl od řídicích bodů, které u parametrizace B-spline mění pozici a tedy i meze při každé iteraci optimalizace, tak u parametrizace Bezier-PARSEC se ohraničení parametrů nemění.

Další použití autokonfiguratoru, tentokrát typickým, je namísto meta-optimalizace. Jde tedy o automatickou konfiguraci optimalizací, například aEA, aPSO nebo PHO. Hlavní nevýhodou autokonfigurace parametrů optimalizací je výpočetní náročnost, tedy je nutné použít superpočítač.

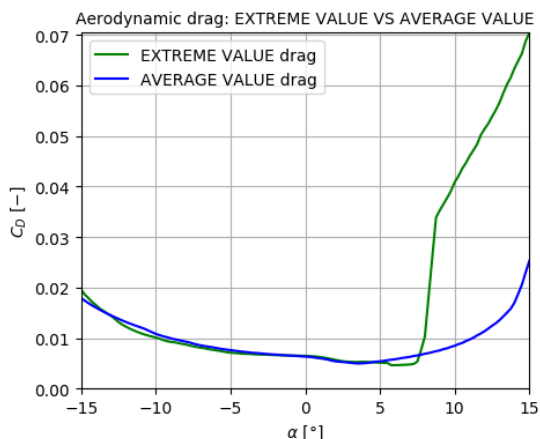
17.3 Srovnání výsledků aerodynamického odporu



Obr. 142: Odporová křivka maximální aerodynamické jemnosti (extrému) PHO a profilu NACA 2412.



Obr. 143: Odporová křivka maximální průměrné aerodynamické jemnosti (průměru) aPSO a profilu NACA 2412.



Obr. 144: Odporová křivka maximální aerodynamické jemnosti (extrému) PHO a maximálního průměru aerodynamické jemnosti (průměru) aPSO.

Nejdůležitějším parametrem pro ustálený let je aerodynamický odpor, ovlivňující spotřebu paliva. Snížením spotřeby paliva dochází jak ke značným ekonomickým úsporám, tak i ke snížení ekologické zátěže. Pro snížení aerodynamického odporu profilu je použito dvou typů optimalizací.

Pro nalezení nejlepšího extrému, tedy úhlu náběhu s nejlepšími aerodynamickými vlastnostmi je použito paralelní hybridní optimalizace PHO. V ní se střídají po určitých časových intervalech optimalizace profilu evolučními algoritmy aEA a optimalizace profilu hejnem částic aPSO. Ocenění účelovou funkcí je aerodynamická jemnost profilu hledající maximální hodnotu aerodynamické jemnosti, tj. extrém.

aEA je uzpůsobena pro použití s parametrizací Bezier-PARSEC 3434 a zároveň používá selekci turnajem s nejlepším výsledkem, tedy o velikosti turnaje 6 jedinců při velikosti populace 25 jedinců. Stejně tak aPSO musí být přizpůsobeno pro spolupráci s parametrizací Bezier-PARSEC 3434 a jsou hledány vždy nejlepší koeficienty aPSO, tj. aPSO používá poměrně nové varianty optimalizace MOL (Many Optimizing Liaisons). Nejlepší výsledek MOL je při parametru váhy setrvačnosti 0.2 a akceleračním koeficientu 2.8.

Na obrázku 142 je zřetelné, že při účelové funkci hledající maximální jemnost (extrém) dosahuje lepších výsledků profil z PHO než profil NACA 2412.

Pro udržení nejlepšího úhlu náběhu pro extrém, během ustáleného letu letounu, je nutno kontrolovat a korigovat úhel náběhu počítačem. Počítač nejenže drží nejlepší úhel náběhu během ustáleného letu, ale také slouží ke kontrole chování letounu při nebezpečných úhlech náběhu. Výhodou této optimalizace je, že při udržení nejlepšího úhlu náběhu dochází ke snížení aerodynamického odporu, tedy i k značnému snížení spotřeby paliva ve srovnání s profilem NACA 2412. Nevýhodou je výskyt potenciálně nebezpečných úhlů náběhu, které jsou však kontrolovány počítačem.

Na některých typech letounů, např. Cessna 172, nelze použít profil s nejlepším extrémem tj. s nejlepšími aerodynamickými vlastnostmi, pokud nejsou osazeny počítačem, kontrolujícím nebezpečné úhly náběhu.

Pro optimalizaci širšího ustáleného letu, který je definován rozsahem 3° - 9° úhlů náběhu (meze jsou vybrány pro demonstraci funkčnosti algoritmů), je použito aPSO s optimalizací počátečního profilu, v tomto případě NACA 2412. Účelová funkce hledá maximální průměr aerodynamické jemnosti profilu v rozsahu širšího ustáleného letu. To se používá k hledání optimalizovaných aerodynamických vlastností již existujícího profilu. Optimalizace definuje tvar profilu na základě řídicích bodů tvaru počátečního profilu. Ty jsou pak měněny za účelem dosažení lepší geometrie profilu s odpovídajícími lepšími aerodynamickými vlastnostmi. Také zde jsou hledány nejlepší koeficienty optimalizace. Stejně tak jako u Paralelní hybridní optimalizace, využívá tato optimalizace nové varianty optimalizace MOL.

Výhodou využití optimalizace širšího ustáleného letu ve výpočtech definovaného rozsahem 3° - 9° úhlů náběhu je bezpečnější profil (pokud se vychází z relativně bezpečného profilu), bez nebezpečných úhlů náběhu v širším ustáleném letu. Nemusí tedy být použito počítače ke kontrole nebezpečných úhlů náběhu. Nevýhodou je, že výsledný průměr snížení aerodynamického odporu je snížen pouze o 18.10% (viz tab. 47) vůči počátečnímu profilu NACA2412. Cena za bezpečnější chování letounu je menší redukce průměrného aerodynamického odporu za ustálený let včetně ekonomických a ekologických dopadů na provoz letounu. Lze použít i pro Cessnu 172.

Z obrázku 143 lze vyčíst, že při širším ustáleném letu dosahuje lepších výsledků profil z aPSO ve srovnání s profilem NACA 2412.

Finální rozdíl mezi výsledky snížení aerodynamického odporu při variantě optimalizace s nejlepším extrémem (viz tab. 46) a při variantě optimalizace širšího ustáleného letu (viz tab. 47) je 23.74 % ve prospěch nejlepšího extrému.

17.4 Srovnání optimálních aerodynamických vlastností

Ocenění účelovou funkcí je aerodynamická jemnost profilu. Hledá se maximální hodnota aerodynamické jemnosti, tedy extrém u optimalizovaného profilu ve srovnání s extrémem u NACA 2412. Pro srovnání aerodynamických vlastností obou profilů bylo nastaveno Reynoldsovo číslo $5.0e+06$ a úhel náběhu pro extrém NACA 2412 je 8.25° a pro optimalizovaný profil je extrém 7.00° .

	C_L	C_D	C_L/C_D	C_M
NACA 2412	1.1542	0.00924	124.9134	- 0.0522
Optimalizovaný profil	1.9056	0.00481	396.1746	- 0.2404
Zlepšení	+ 65.10 %	- 47.94 %	+ 217.16 %	+ 360.54 %

Tab. 46: Srovnání aerodynamických vlastností NACA 2412 a optimalizovaného profilu.

Z tabulky 46 je patrné, že došlo k nárůstu ocenění profilu o 217.16 %. Nejdůležitějším výsledkem je radikální snížení aerodynamického odporu vůči profilu NACA 2412, což znamená z hlediska ekonomiky letu značně menší spotřebu paliva a také z ekologického hlediska značné snížení emisí.

Širší ustálený let je definován rozsahem úhlů náběhu, v tomto případě $3^\circ - 9^\circ$. Účelová funkce hledá maximální průměr aerodynamické jemnosti profilu v rozsahu širšího ustáleného letu. Pro výpočet aerodynamických vlastností obou srovnávaných profilů bylo nastaveno Reynoldsovo číslo $5.0e+06$ a úhel náběhu $3.00^\circ - 9.00^\circ$.

	C_L	C_D	C_L/C_D	C_M
NACA 2412	0.8779	0.00732	119.8291	- 0.0456
Optimalizovaný profil	1.0997	0.00600	183.2631	- 0.0556
Zlepšení	+ 25.26 %	- 18.10 %	+ 52.94 %	+ 22.11 %

Tab. 47: Srovnání aerodynamických vlastností NACA 2412 a optimalizovaného profilu.

Z výsledků je zřejmý rozdíl mezi optimalizací nejlepších extrémů a optimalizací pro širší ustálený let. Byly nalezeny jiné výsledky u optimalizace s počátečním profilem NACA 2412. Pozitivem je nalezení bezpečnějšího profilu, bez nebezpečných extrémů. V tomto případě dochází ke zvýšení ocenění profilu pouze o 52.94 % a stejně tak aerodynamický odpor je snížen pouze o 18.10 %.

18 Přínosy práce

Zásadní význam a přínos této práce spočívá ve vytvoření nového, dosud nepublikovaného algoritmu pro optimalizaci geometrie profilu. Nová paralelní hybridní optimalizace je založena na optimalizaci profilu evolučními algoritmy (aEA) a na optimalizaci profilu hejnem částic (aPSO)

Došlo k vytvoření tří druhů nových variant optimalizací, jako jsou optimalizace profilu evolučních algoritmů (aEA), algoritmy optimalizace profilu hejnem částic (aPSO), optimalizace profilu hejnem částic (aPSO) s výchozím profilem a zcela nový paralelní hybridní algoritmus (PHO).

V současnosti je často využíván parametrický model profilu Bezier-PARSEC 3434, který umožňuje vytvořit tvar zcela nového profilu, zatímco algoritmus, založený na zvoleném profilu, používá model profilu, založený na řídicích bodech B-splíny parametrizace. Pomocí účelové funkce se následně ohodnotí aerodynamické parametry vypočítané programem Xfoil.

Po zamítnutí použití statistických testů, hlavně pak Wilcoxonova párového testu vyžadujícího závislé výběry (související páry) byl shledán nejvhodnější Mann-Whitneyův test, který pracuje se zcela nezávislými výběry.

Díky parametrizacím při optimalizaci geometrie profilu byl použit autokonfigurátor IRACE, který byl aplikován namísto aPSO s výchozím profilem.

Těmito novými typy vytvořených optimalizací tvaru profilu bylo dosaženo:

- počáteční podmínky nejsou vázány na předlohu, tzn., jsou zcela obecné a výsledkem je profil s největší aerodynamickou jemností profilu,
- počáteční podmínky jsou vázány na předlohu, jedná se o optimalizaci již daného profilu, kdy předloha hledá nejlepší průměr aerodynamické jemnosti profilu,
- rychlá konvergence k výsledku, ve srovnání se současnými používanými výpočty (dle nastavené účelové funkce),
- výpočty optimalizovaných profilů jsou rychlé, ve srovnání se současnými používanými výpočty,
- největší aerodynamickou jemnost profilu se promítla do značných úspor paliva a snížení zátěže životního prostředí v řádech desítek procent,
- pro širší ustálený let bylo dosaženo úspor paliva a snížení zátěže životního prostředí v řádech desítek procent.

19 Pokračování práce

Výsledky, popsané v předchozích kapitolách, předurčují tuto práci k dalšímu rozvoji jak na úrovni vlastní optimalizace, použití autokonfiguratoru na parametrizaci i na optimalizaci, tak na úrovni experimentálního ověření.

V oblasti optimalizace by měly být prozkoumány především:

- alternativní možnosti optimalizace s vyšší rychlostí zpracování optimalizace, např. zavedením SOMA optimalizace, kernel optimalizační metody, popř. převedením do jazyka „C“ nebo využití vícero jader procesoru,
- optimalizace speciálních profilů, jako jsou např. autostabilní profily,
- dále by měla být podrobněji prozkoumána možnost odstranění extrémů v průběhu součinitele vzlaku, odporu a momentu profilu. Tato část se úzce pojí k nalezení takových účelových funkcí, včetně omezujících podmínek, které by eliminovaly extrémy a to i za cenu zhoršení aerodynamických vlastností profilu,
- prozkoumat a popř. využít ANN (artificial neural networks) hlubokého učení s využitím hejnových algoritmů ke generování nových, excelentních profilů.

Experimentální část by měla zahrnovat:

- tvorbu funkčních vzorků křidel s optimalizovanými profily,
- měření funkčních vzorků křidel v aerodynamickém tunelu,
- získání experimentálních dat z měření v aerodynamického tunelu,
- analýzu rozdílů mezi výpočtem a skutečností.

Následně by měl být podán EU a USA patent na upravené konstrukční řešení.

20 Závěr

Práce je zaměřena na optimalizaci geometrie profilu nosné plochy letounu. Optimalizace geometrie profilu nosných ploch letounu má klíčový význam pro aerodynamiku a ekonomii provozu letounu.

Byly vytvořeny nové varianty optimalizací přizpůsobené na aplikaci aerodynamiky. Vznikly tak evoluční algoritmy profilu (aEA), optimalizace profilu hejnem částic (aPSO) a nakonec optimalizace profilu hejnem částic (aPSO) s výchozím profilem.

Zásadní význam a přínos této práce spočívá ve vytvoření nového algoritmu pro optimalizaci geometrie profilu. Nová paralelní hybridní optimalizace (PHO) je založena na optimalizaci profilu evolučními algoritmy a na optimalizaci profilu hejnem částic. Tato nová optimalizace dosahuje lepších výsledků, než je dosaženo v samostatné optimalizaci profilu evolučními algoritmy nebo optimalizací profilu hejnem částic, což se odráží v rychlejší konvergenci a díky paralelizmu i ve větší diverzitě řešení.

Pro reprezentaci geometrie profilů bylo použito parametrizace, tedy definování geometrie profilu na základě parametrů Bezier-PARSEC 3434, kdy parametry jsou hlídány tak, aby vycházely reálné tvary profilů. Byla vytvořena nová verze parametrizace.

Další použitou parametrizací byla B-spline parametrizace, která byla využita pro optimalizace s výchozím profilem, kdy změnou vertikální pozice řídicích bodů jsou generovány nové tvary profilů a tedy i nové aerodynamické vlastnosti.

Byl aplikován autokonfigurační nástroj IRACE na parametrizaci B-spline s výhledem na použití s Bezier-PARSEC parametrizací a aplikaci na optimalizace.

Pro srovnání výsledků jednotlivých optimalizací jsou používány statistické testy, kdy se jako nejvhodnější test, speciálně v případě této práce, osvědčil Mann-Whitneyův test.

Pro srovnání výsledků výpočtu aerodynamických vlastností i pro výchozí profil, byl vybrán jeden z nejpoužívanějších profilů NACA 2412.

Byly vytvořeny účelové funkce vytvářející nejkvalitnější profily z hlediska největší aerodynamické jemnosti, tak i největšího průměru aerodynamické jemnosti.

Pro optimalizaci profilu evolučními algoritmy, modifikované z původních evolučních algoritmů, vychází nejlépe optimalizace při selekci turnajem o velikosti 6 a při velikosti populace 25. Křížení jedinců je pozměněno pro použití s parametrizací Bezier-PARSEC.

Optimalizace profilu hejnem částic při velikost hejna 25 dosahují nejlepších výsledů s váhou setrvačnosti ω -0.2 a s akceleračním koeficientem c_2 . 2.8. Optimalizace je založena na variantě optimalizace Many Optimizing Liaisons, která byla také pozměněna pro použití parametrizace Bezier-PARSEC.

Lepších výsledků ohodnocení bylo dosaženo u nové paralelní hybridní optimalizace po 5 minutách běhu evolučního algoritmu profilu a po 25 minutách běhu optimalizace profilu hejnem částic.

Z výsledků je zřejmé, že nová paralelní hybridní optimalizace dává lepší výsledky ohodnocení v závislosti na čase oproti novým evolučním optimalizacím.

Vytvořeným programovým vybavením (20 750 řádků v Python) proběhlo rozsáhlé testování optimalizací, které potvrdilo správnost návrhu a řešení nových optimalizačních metod.

Přínosy práce z hlediska aerodynamiky spočívají v nalezení minimálního aerodynamického odporu v jednom extrému až o desítky procent vůči profilu NACA 2412 a pro širší ustálený let v rozmezí 3° - 9° úhlu náběhu je snížení aerodynamického odporu o 18.10% vůči profilu NACA 2412. Současným trendem u dopravních letounů je optimalizace s jedním extrémem, kdy počítač udržuje letoun v nejlepším úhlu náběhu při nejmenším aerodynamickému odporu, který se adekvátně promítá jak do spotřeby paliva, tedy i do snížení nákladů na let.

Na tuto optimalizaci profilu přímo navazuje řízení jeho geometrie dle mého vynálezu (příloha B). Idea vynálezu vychází z měnících se nároků na aerodynamiku letounu. Jedná se o možnost měnit tvar profilu (morphing) za letu tak, aby chování letounu bylo bezpečné a současně i ekonomické.

Možnosti úspor ve spotřebě paliva naznačené v dizertační práci lze pokládat za unikátní.

Srovnáním dosažených výsledků z optimalizací profilu (jen s morphingem horní strany profilu) letounu ATR-42-4 je platnost výsledků zcela evidentní.

Stanovené cíle dizertační práce považuji za beze zbytku splněné a doufám, že přispějí k rozvoji oboru.

Použitá literatura

- [1] DOFEK, Ivan. *PARAMETRICKÁ TVAROVÁ OPTIMALIZACE LETOUNU Z AERODYNAMICKÉHO HLEDISKA*. VUT v Brně. Brno, ČR, 2014.
- [2] MÜLLER, Jan a Rudolf MÜLLER. *Zařízení pro kontinuální a definovanou změnu geometrie tvaru a zakřivení profilu křidel*. Česká republika. Patent 300 728. Uděleno 29.7.2009.
- [3] RAYMER, Daniel. *Aircraft Design: A Conceptual Approach, Sixth Edition*. Amazon. USA: American Institute of Aeronautics and Astronautics, 2018. ISBN 978-1-60086-911-2.
- [4] ZELINKA, Ivan a Jouni LAMPINEN. *Mechanical engineering problem optimization by SOMA*. New Optimization Techniques in Engineering. Berlin, Heidelberg, 2004.
- [5] DANTZIG, George. *The Nature of Mathematical Programming*. Mathematical Programming Glossary, INFORMS Computing Society. USA, 2014.
- [6] YANG, Xin-She. *Introduction to Mathematical Optimization: From Linear Programming to Metaheuristics*. NA. UK: Cambridge International Science Publishing, 2008. ISBN 978-1-904602-82-8.
- [7] *Optimalizační problém* [online]. USA: wikipedie, 2017 [cit. 2022-04-20]. Dostupné z: https://cs.wikipedia.org/wiki/Optimalizační_přoblém
- [8] MITSOS, Alexander, Jaromi NAJMAN a Ioannis KEVREKIDIS. *Optimal Deterministic Algorithm Generation*. Journal of Global Optimization. Netherlands, 2016.
- [9] YAZICI, Adnan. *The Theory of NP-Completeness*. Middle East Technical University. Ankara, Turkey, 2007.
- [10] ITSYKSON, Dmitry a Dmitry SOKOLOV. *On fast heuristic non-deterministic algorithms and short heuristic proofs*. Fundamenta Informaticae Journal. Poland, 2014.
- [11] YANG, Xin-She. *Nature-Inspired Metaheuristic Algorithms*. Amazon. Cambridge, UK: Luniver Press, 2010. ISBN 9781905986286.
- [12] CHUROVÁ, VENDULA. *Optimalizační algoritmy inspirované živou přírodou*. MASARYKOVA UNIVERZITA. Brno, ČR, 2015.
- [13] *Metaheuristic Picture* [online]. USA: Wikipedia, 2021 [cit. 2021-06-24]. Dostupné z: <https://en.wikipedia.org/wiki/Metaheuristic>
- [14] YU, Xinjie a Mitsuo GEN. *Introduction to Evolutionary Algorithms*. Amazon. London, UK: Springer, 2010. ISBN 978-1-84996-129-5.
- [15] CARR, Jenna. *An Introduction to Genetic Algorithms*. USA, 2014.
- [16] KITA, Eisuke. *EVOLUTIONARY ALGORITHMS*. InTech. Rijeka, Croatia, 2011.
- [17] BODENHOFER, Ulrich. *Genetic Algorithms: Theory and Applications*. Johannes Kepler University Linz. Austria, 2004.

- [18] WALKER, Matthew. *Introduction to Genetic Programming*. Canada, 2001.
- [19] KRAWIEC, Krzysztof a Tomasz PAWLAK. *Genetic Programming*. Poznan University of Technology. Poznan, Poland, 2001.
- [20] KOZA, John. *Genetic Programming: On the Programming of Computers by Means of Natural Selection*. A Bradford Book. USA, 1992.
- [21] VITÁNYI, Paul. *A Discipline of Evolutionary Programming*. Proc. 7th Int’nl Workshop on Algorithmic Learning Theory, Lecture Notes in Artificial Intelligence. Netherlands, 1996.
- [22] BÄCK, Thomas, Rudolph GÜNTER a Hans-Paul SCHWEFEL. *Evolutionary Programming and Evolution Strategies: Similarities and Differences*. University of Dortmund. Germany, 1997.
- [23] YAO, Xin, Yong LIU a Guangming LIN. *Evolutionary Programming Made Faster*. IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION Journal. USA, 1999.
- [24] BHATTACHARJYA, Rajib. *Introduction to Differential Evolution*. Indian Institute of Technology Guwahtai. India, 2004.
- [25] PIYASATIAN, Napapan. *Differential Evolution A Simple Evolution Strategy for Fast Optimization*. King Mongkut’s Institute of Technology Ladkrabang. Bangkok, Thailand, 1999.
- [26] RAHNAMAYAN, Shahryar, Hamid TIZHOOSH a Magdy SALAMA. *Opposition-Based Differential Evolution*. IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION Journal. USA, 2008.
- [27] HANSEN, Nikolaus, Dirk ARNOLD a Anne AUGER. *Evolution Strategies*. Handbook of Computational Intelligence. Germany, 2015.
- [28] BÄCK, Thomas. *Evolution Strategies: An Alternative Evolutionary Algorithm*. Artificial Evolution. Brest, France, 1995.
- [29] ORTEGA, Alfonso, Marina CRUZ a Manuel ALFONSECA. *Christiansen Grammar Evolution: Grammatical Evolution With Semantics*. IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION Journal. USA, 2007.
- [30] ROHL, J. *A note on Backus Naur form*. The Computer Journal. Oxford, UK, 1968.
- [31] YANG, Shengxiang a Xin YAO. *Evolutionary Computation for Dynamic Optimization Problems*. Studies in Computational Intelligence. EU, 2013.
- [32] LANE, Fergal, R. AZAD a Conor RYAN. *Principled Evolutionary Algorithm Design and the Kernel Trick*. Genetic and Evolutionary Computation Conference (GECCO). USA, 2016.
- [33] VÁCHA, Ladislav. *Řešení optimální cesty svozu odpadů pomocí rojové inteligence*. Jihočeská univerzita v Českých Budějovicích. České Budějovice, ČR, 2015.

- [34] CHAKRABORTY, Amrita a Arpan KAR. *Swarm Intelligence: A Review of Algorithms. Nature-Inspired Computing and Optimization. Modeling and Optimization in Science and Technologies*. UK: Springer, 2017. ISBN 978-3319509198.
- [35] AHMED, Hazem a Janice GLASGOW. *Swarm Intelligence: Concepts, Models and Applications*. Queen's University, School of Computing Technical Reports. Kingston, Canada, 2012.
- [36] NAYYAR, Anand, Dac-Nhuong LE a Nhu NGUYEN. *Advances in Swarm Intelligence for Optimizing Problems in Computer Science*. Amazon. USA: CRC Press/Taylor & Francis Group, 2019. ISBN 9781138482517.
- [37] KENNEDY, James a Russell EBERHART. *Particle swarm optimization*. Proceedings of ICNN'95 - International Conference on Neural Networks. Australia, 1995.
- [38] BHATTACHARJYA, Rajib. *Introduction to Particle Swarm Optimization*. Indian Institute of Technology Guwahati. India, 2004.
- [39] SCHUTTE, Jaco. *The Particle Swarm Optimization Algorithm*. Vilniaus universitetas. Lithuania, 2005.
- [40] EL-SHORBAGY, M.A. a Aboul HASSANIEN. *Particle Swarm Optimization from Theory to Applications*. International Journal of Rough Sets and Data Analysis. Egypt, 2018.
- [41] VOLNÁ, EVA. *EVOLUČNÍ ALGORITMY A NEURONOVÉ SÍTĚ*. Ostravská univerzita v Ostravě. 2012, Ostrava, ČR.
- [42] ZELINKA, Ivan. *Evoluční výpočetní techniky - Principy a aplikace*. BEN. Praha, ČR, 2009.
- [43] KODĚROVÁ, Lucie. *Heuristiky*. UNIVERZITA PALACKÉHO V OLOMOUCI PŘÍRODOVĚDECKÁ FAKULTA. Olomouc, ČR, 2008.
- [44] STEPHEN, Adubi a Sanjay MISRA. *A Comprehensive Study on the Ant Colony Optimization Algorithms*. 11th International Conference on Electronics Computer and Computation · ICECCO. Abuja, Nigeria, 2014.
- [45] SALAMI, Nada. *Ant Colony Optimization Algorithm*. UbiCC Journal. New York, USA, 2009.
- [46] BHOKRAY, Ketan. *Artificial Bee Colony Optimization*. Indian Institute of Technology. Bombay, India, 2013.
- [47] KARABOGA, D. a B. BASTURK. *On the performance of artificial bee colony (ABC) algorithm*. Applied Soft Computing Journal. Netherlands, 2007.
- [48] KARABOGA, Dervis a Bahriye BASTURK. *Artificial Bee Colony (ABC) Optimization Algorithm for Solving Constrained Optimization Problems*. 12th International Fuzzy Systems Association World Congress. Cancun, Mexico, 2007.
- [49] LIU, JENN- a CHUNG-CHIH LI. *An Improved Artificial Bee Colony Algorithm Applied to Engineering Optimization Problems*. JOURNAL OF INFORMATION SCIENCE AND ENGINEERING. Taiwan, 2016.

- [50] JOHARI, Nur, Azlan ZAIN, Noorfa MUSTAFFA a Amirmudin UDIN. *Firefly Algorithm for Optimization Problem*. Applied Mechanics and Materials Journal. Switzerland, 2013.
- [51] YANG, Xin-She a Xing-shi HE. *Why the Firefly Algorithm Works?*. Nature-Inspired Algorithms and Applied Optimization. USA: Amazon, 2018. ISBN 978-3319676685.
- [52] FATEEN, Seif-Eddeen a Adrián BONILLA-PETRICIOLET. *Intelligent Firefly Algorithm for Global Optimization*. Cuckoo Search and Firefly Algorithm. Switzerland: Springer, 2014. ISBN 978-3-319-02141-6.
- [53] STAFLEU, Frans. *Lamarck: The Birth of Biology*. Taxon. Austria, 1971.
- [54] DARWIN, Charles. *On the Origin of Species*. Amazon. UK: Signet, 2003. ISBN 978-0451529060.
- [55] STARK, Alan. *Stages in the evolution of the Hardy-Weinberg law*. Genetics and Molecular Biology Journal. Brazil, 2006.
- [56] EDWARDS, A. *Perspectives*. Genetics Society of America. USA, 2008.
- [57] KOMÍNEK, JAN. *HEURISTICKÉ ALGORITMY PRO OPTIMALIZACI*. VUT Brno. Brno, ČR, 2012.
- [58] STUDNIČKA, VLADIMÍR. *GENETICKÉ ALGORITMY – MULTI-CORE CPU IMPLEMENTACE*. VUT Brno. Brno, ČR, 2010.
- [59] *Intelligence roje (Swarm intelligence)* [online]. Praha, ČR: Wilmington, 2015 [cit. 2021-04-04]. Dostupné z: <https://managementmania.com/cs/rojova-intelligence-swarm-intelligence>
- [60] LI, Changhe. *Particle Swarm Optimization in Stationary and Dynamic Environments*. University of Leicester, Department of Computer Science. Leicester, UK, 2010.
- [61] HEPNER, Frank a Ulf GRENANDER. *A Stochastic Nonlinear Model for Coordinated Bird Flocks*. American Association for the Advancement of Science. USA, 1990.
- [62] PÁNEK, Ondřej. *Algoritmus optimalizace hejnem částic: vývoj a jeho aplikace*. ČVUT v Praze. Praha, ČR, 2018.
- [63] TALUKDER, Satyobroto. *Mathematical Modelling and Applications of Particle Swarm Optimization*. School of Engineering at Blekinge Institute of Technology. Karlskrona, Sweden, 2011.
- [64] NĚMEČEK, PATRIK. *OPTIMALIZAČNÍ ÚLOHY NA BÁZI ČÁSTICOVÝCH HEJN (PSO)*. VUT Brno. Brno, ČR, 2014.
- [65] ALMEIDA, Bruno a Victor LEITE. *Particle Swarm Optimization: A Powerful Technique for Solving Engineering Problems*. Swarm Intelligence. London, UK: IntechOpen, 2019. ISBN 978-1-78984-537-2.
- [66] PEDERSEN, Magnus. *Good Parameters for Particle Swarm Optimization*. Hvass Laboratories. EU, 2010.

- [67] KENNEDY, James. *The particle swarm: Social adaptation of knowledge*. IEEE International Conference on Evolutionary Computation. USA, 1997.
- [68] HICKS, Raymond a Preston HENNE. *Wing Design by Numerical Optimisation*. Journal of Aircraft. USA, 1978.
- [69] MASTERS, D., N. TAYLOR, T. RENDALL, C. ALLEN a D. POOLE. *Review of Aerofoil Parameterisation Methods for Aerodynamic Shape Optimisation*. AIAA Journal. Reston, USA, 2015.
- [70] MASTERS, D., N. TAYLOR, T. RENDALLZ, C. ALLENX a D. POOLE. *A Geometric Comparison of Aerofoil Shape Parameterisation Methods*. AIAA Journal. Reston, Virginia, USA, 2006.
- [71] *Radial basis function* [online]. USA: wikipedie, 2023 [cit. 2023-08-01]. Dostupné z: https://en.wikipedia.org/wiki/Radial_basis_function
- [72] ZAPLETAL, Jiří. *Aplikace Radiálních bázových funkcí*. Západočeská univerzita v Plzni, Katedra informatiky a výpočetní techniky. Plzeň, ČR, 2007.
- [73] NOVÁK, Jiří. *METODY FFD*. VUT Brno. Brno, ČR, 2017.
- [74] COQUILLART, Sabine a Pierre JANCÉNE. *Animated free-form deformation*. Le Chesnay, France, 1991.
- [75] LAMOUSIN, Henry a Warren WAGGENSPACK. *NURBS-based free-form deformations*. IEEE Computer Graphics and Applications. Louisiana, USA, 1994.
- [76] SAMAREH, Jamshid. *Aerodynamic Shape Optimization Based on Free-Form Deformation*. 10th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference. Hampton, USA, 2004.
- [77] SAMEC, Václav. *Modelování pomocí lokálních deformací*. Masarykova univerzita, Fakulta informatiky. Brno, ČR, 2005.
- [78] REID, Janeita. *Free-Form Deformation (FFD)*. Jamaica, 2021.
- [79] NEGAHBAN, Mir, Musavir BASHIR a Ruxandra BOTEZ. *Free-Form Deformation Parameterization on the Aerodynamic Optimization of Morphing Trailing Edge*. École de Technologie Supérieure, Université du Québec. Canada, 2023.
- [80] RONZHEIMER, Arno. *SHAPE PARAMETERISATION BASED ON FREEFORM DEFORMATION IN AERODYNAMIC DESIGN OPTIMISATION*. German Aerospace Center DLR, Institute of Aerodynamics and Flow Technology. Germany, 2004.
- [81] RONZHEIMER, Arno. *Prospects of Geometry Parameterization based on Freeform Deformation in MDO*. Institute of Aerodynamics and Flow Technology German Aerospace Center. Braunschweig, Germany, 2006.
- [82] SALUNKE, Nilesh, Juned AHAMAD a S.A. CHANNIWALA. *Airfoil Parameterization Techniques: A Review*. American Journal of Mechanical Engineering. USA, 2014.

- [83] SRIPAWADKUL, Vis, Mattia PADULO a Marin GUENOV. *A Comparison of Airfoil Shape Parameterization Techniques for Early Design Optimization*. Cranfield University. UK, 2010.
- [84] FALCO, I., A. CIOPPA, R. BALIO a E. TARANTINO. *Breeder Genetic Algorithms for Airfoil Design Optimisation*. National Research Council of Italy. Naples, Italy, 2011.
- [85] STRAATHOF, Michiel. *Shape Parameterization in Aircraft Design: A Novel Method, Based on B-Splines*. Delft University of Technology. Netherlands, 2012.
- [86] NETO, Rafael, Luciano NOLETO, Josiane SOUZA a Antonio BRASIL. *PARSEC PARAMETERIZATION METHODOLOGY FOR ENHANCING AIRFOILS GEOMETRY USING PSO ALGORITHM*. XXXVII IBERIAN LATIN AMERICAN CONGRESS ON COMPUTATIONAL METHODS IN ENGINEERING. Brazil, 2016.
- [87] KHARAL, Athar a Ayman SALEEM. *Neural networks based airfoil generation for a given C_p using Bezier-PARSEC parameterization*. Aerospace Science and Technology. Amsterdam, Netherlands, 2011.
- [88] KIM, Jae. *Optimal Level of Significance for Regression and Other Statistical Tests*. 2022.
- [89] *Testování statistických hypotéz* [online]. 2023 [cit. 2023-10-21]. Dostupné z: https://www.wikiskripta.eu/w/Testování_statistických_hypotéz
- [90] VERHAGEN, Arianne, Raymond OSTELO a Arno RADEMAKER. *Is the p value really so significant?**. Australian Journal of Physiotherapy. 2004.
- [91] *P-hodnota* [online]. 2023 [cit. 2023-10-21]. Dostupné z: <https://www.wikiskripta.eu/w/P-hodnota>
- [92] *P-hodnota*. [online]. 2018 [cit. 2023-10-21]. Dostupné z: <https://cs.wikipedia.org/wiki/P-hodnota>
- [93] GLEN, Stephanie. *P-Value Explained in One Picture* [online]. 2019 [cit. 2023-10-21]. Dostupné z: <https://www.datasciencecentral.com/p-value-explained-in-one-picture/>
- [94] KIM, Jae a In CHOI. *Choosing the Level of Significance: A Decision-theoretic Approach*. Abacus. 2019.
- [95] TRAVERS, Jason, Bryan COOK a Lysandra COOK. *Null Hypothesis Significance Testing and p Values*. Learning Disabilities Research & Practice. 2017.
- [96] PERNET, Cyril. *Null hypothesis significance testing: a short tutorial*. F1000 Research Ltd. 2015.
- [97] *Gaussova funkce* [online]. 2010 [cit. 2023-11-12]. Dostupné z: https://cs.wikipedia.org/wiki/Gaussova_funkce
- [98] *ZÁKLADNÍ TYPY ROZDĚLENÍ PRAVDĚPODOBNOSTI SPOJITÉ NÁHODNÉ VELIČINY* [online]. 2019 [cit. 2023-11-12]. Dostupné z: <https://homel.vsb.cz/~oti73/cdpast1/KAP05/PRAV5.HTM>
- [99] BOŘIL, Tomáš. *Testy a p-hodnota* [online]. 2015 [cit. 2023-10-21]. Dostupné z: https://fu.ff.cuni.cz/STAT/13_testy_p_hodnota.html

- [100] LEVINE, Timothy, René WEBER, Hee PARK a Craig HULLETT. *A Communication Researchers' Guide to Null Hypothesis Significance Testing and Alternatives*. Human Communication Research. 2008.
- [101] *Testování hypotéz ve statistice* [online]. 2017 [cit. 2023-11-12]. Dostupné z: <https://cit.vfu.cz/statpotr/POTR/Teorie/Predn3/hypotezy.htm>
- [102] PAMPLONA, Fabricio. *Co jsou statistické parametrické testy a kde je použít?* [online]. 2023 [cit. 2023-11-12]. Dostupné z: <https://mindthegraph.com/blog/cs/statistics-parametric-tests/>
- [103] *Testování statistických hypotéz*. [online]. 2020 [cit. 2023-10-21]. Dostupné z: https://www.fd.cvut.cz/departament/k611/pedagog/K611THO_soubory/0_testovani_hypotez.pdf
- [104] LAJDOVÁ, DAGMAR. *Neparametrické metody v systému STATISTICA*. MASARYKOVA UNIVERZITA, PŘÍRODOVĚDECKÁ FAKULTA. Brno, ČR, 2009.
- [105] CORDER, Gregory a Dale FOREMAN. *Nonparametric Statistics: A Step-by-Step Approach*. Second Edition. U.S.: Wiley, 2014. ISBN 978-1118840313.
- [106] BEZDĚK, Václav. *Možnosti a meze využití fuzzy logiky pro řešení problémů ekonomiky a managementu*. Univerzita Tomáše Bati Zlíně, Fakulta managementu a ekonomiky. Zlín, ČR, 2008.
- [107] TURČIČOVÁ, Marie. *Dvouvýběrové testy*. Univerzita Karlova. Praha, ČR, 2021.
- [108] *Toggle the table of contents Wilcoxon signed-rank test* [online]. 2020 [cit. 2023-10-30]. Dostupné z: https://en.wikipedia.org/wiki/Wilcoxon_signed-rank_test
- [109] *Paired Wilcoxon Signed Rank Test* [online]. [cit. 2023-10-30]. Dostupné z: <https://openpress.usask.ca/introtoappliedstatsforpsych/chapter/16-5-paired-wilcoxon-signed-rank-test/>
- [110] VARGICOVÁ, Katarína. *MATEMATICKÉ METODY V EKONOMII*. VUT Brno. Brno, ČR, 2018.
- [111] DIVINE, George, H.James NORTON, Anna BARÓN a Elizabeth JUAREZ-COLUNGA. *The Wilcoxon–Mann–Whitney Procedure Fails as a Test of Medians*. The American Statistician. 2018.
- [112] KRUSKAL, William a W. WALLIS. *Use of Ranks in One-Criterion Variance Analysis*. Journal of the American Statistical Association. 1952.
- [113] FRIEDMAN, Milton. *The use of ranks to avoid the assumption of normality implicit in the analysis of variance*. Journal of the American Statistical Association. 1937.
- [114] BOŘIL, Tomáš. *Testy středních hodnot* [online]. 2015 [cit. 2023-11-09]. Dostupné z: https://fu.ff.cuni.cz/STAT/16_testy_strednich_hodnot.html
- [115] OPENAI, . *Kritéria k výběru statistického testu*. In: *GPT-3.5* [[chatbot]. [online], 2023-11-09 [cit. 2023-11-09]. Dostupné z: <https://chat.openai.com/chat>

- [116] LÓPEZ-IBÁÑEZ, Manuel, Leslie PÉREZ CÁCERES, Jérémie DUBOIS-LACOSTE, Thomas STÜTZLE a Mauro BIRATTARI. *The irace Package: User Guide*. Bruxelles, Belgium, 2016.
- [117] PÉREZ CÁCERES, Leslie, Manuel LÓPEZ-IBÁÑEZ a Thomas STÜTZLE. *An Analysis of Parameters of irace*. IRIDIA, CoDE, Université libre de Bruxelles. Bruxelles, Belgium, 2014.
- [118] PÉREZ CÁCERES, Leslie, Manuel LÓPEZ-IBÁÑEZ, Holger HOOS a Thomas STÜTZLE. *An Experimental Study of Adaptive Capping in irace*. Springer International Publishing. EU, 2017.
- [119] LÓPEZ-IBÁÑEZ, Manuel, Jérémie DUBOIS-LACOSTE, Leslie PÉREZ CÁCERES, Mauro BIRATTARI a Thomas STÜTZLE. *The irace package: Iterated racing for automatic algorithm configuration*. Operations Research Perspectives. 2016.
- [120] MEJÍA-DE-DIOS, Jesús-Adolfo, Efrén MEZURA-MONTES a Marcela QUIROZ. *Automated parameter tuning as a bilevel optimization problem solved by a surrogate-assisted population-based approach*. Applied Intelligence. 2021.
- [121] GABOR, Oliviu, Andreea KOREANSCHI a Ruxandra BOTEZ. *Low-speed aerodynamic characteristics improvement of ATR 42 airfoil using a morphing wing approach*. Department of Automated Production Engineering École de Technologie Supérieure. Montréal, Canada, 2012.
- [122] PANKONIEN, Alexander. *Smart Material Wing Morphing for Unmanned Aerial Vehicles*. University of Michigan. Michigan, USA, 2015.
- [123] SKOŘEPA, Tomáš. *Prvky pasivního řízení mezní vrstvy na křídle výkonového kluzáku*. ČVUT v Praze. Praha, ČR, 2021.
- [124] DOJODIHARDJO, Harijono. *Numerical Modelling of Flapping Wing Kinematics and Aerodynamics*. LAP LAMBERT Academic Publishing. Republic of Moldova, 2017.
- [125] *Cessna 172* [online]. 2023 [cit. 2023-09-20]. Dostupné z: https://en.wikipedia.org/wiki/Cessna_172
- [126] *The Reynolds Number About Rowing and Flying* [online]. EU: AeroDrag Publishing, 1992 [cit. 2023-07-25]. Dostupné z: <http://www.aerodrag.com/Articles/ReynoldsNumber.htm>
- [127] *Cessna 172S Skyhawk - FAQ's* [online]. USA: h-aviation, 2020 [cit. 2023-07-25]. Dostupné z: <https://www.h-aviation.com/cessna-172s---skyhawk.html>
- [128] MCIVER, John. *Cessna Skyhawk II/100 (172) Performance Assessment*. USA, 2003.
- [129] *TRANSFER Výzkum a vývoj pro letecký průmysl*. Výzkumný a zkušební letecký ústav. Praha, ČR, 2006.
- [130] ROSKAM, Jan. *Airplane Design*. DARcorporation. USA, 2003.
- [131] FELDOVÁ, Petra. *OPTIMALIZACE ZADNÍHO KŘÍDLA ZÁVODNÍHO VOZU*. VUT Brno. Brno, ČR, 2018.

- [132] KŘÍŽÁK, Michal. *INTEGRACE VÝPOČTOVÉHO PROGRAMU XFOIL V PROSTŘEDNÍ MATLAB*. VUT v Brně. Brno, ČR, 2010.
- [133] LANE, Kevin a David MARSHALL. *Inverse Airfoil Design Utilizing CST Parameterization*. 48th AIAA Aerospace Sciences Meeting Including the New Horizons Forum and Aerospace Exposition. Orlando, Florida, USA, 2010.
- [134] ZÁCHA, Pavel. *FLUENT přednáška Workbench Design Modeler*. ČVUT v Praze. Praha, ČR, 2010.
- [135] BLEJCHAŘ, Tomáš. *TURBULENCE MODELOVÁNÍ PROUDĚNÍ - CFX*. Vysoká škola báňská – Technická univerzita Ostrava. Ostrava, ČR, 2012.
- [136] BEDNÁŘ, Eduard. *ANALÝZA VLIVU PROUDĚNÍ PLYNU V OBLASTI UMÍSTĚNÍ VZORKU V KOMOŘE ENVIROMENTÁLNÍHO RASTROVACÍHO ELEKTRONOVÉHO MIKROSKOPU*. VUT v Brně. Brno, ČR, 2016.
- [137] VRBA, Pavel. *Řešení proudění v pokusném kanálu pomocí CFD*. Západočeská univerzita v Plzni. Plzeň, ČR, 2016.
- [138] JOCH, Lukáš. *MODELOVÁNÍ PROUDĚNÍ V ROTAČNÍCH STROJÍCH*. VUT v Brně. Brno, ČR, 2010.
- [139] *ANSYS Release Notes*. ANSYS, Inc. Canonsburg, USA, 2017.
- [140] PUREVSUREN, Enkhdavaa a De TIAN. *Aerodynamic CFD study of Wortmann FX66-S-196 Airfoil*. International Journal of Engineering Research & Technology. Gandhinagar, GUJARAT, INDIA, 2018.
- [141] ŠIDLOF, Petr. *Metody a postupy ve výpočetní mechanice tekutin (CFD)*. Technická univerzita v Liberci. Liberec, ČR, 2015.
- [142] WALTER, Jiří. *POROVNÁNÍ EXPERIMENTÁLNÍHO A NUMERICKÉHO ŘEŠENÍ VÝVOJE MEZNÍ VRSTVY NA LETECKÉM PROFILU*. ČVUT v Praze. Praha, ČR, 2023.
- [143] LÓPEZ-IBÁÑEZ, Manuel. *TODO* [online]. 2023 [cit. 2023-11-30]. Dostupné z: <https://github.com/MLopez-Ibanez/irace/blob/master/TODO>
- [144] DRELA, Mark a Harold YOUNGREN. *XFOIL 6.9 User Primer*. 2001.

Seznam zkratk a symbolů

aEA	Optimalizace profilu evolučními algoritmy (aerofoil Evolutionary Algorithms)
ALGOL	Počítačový programovací jazyk (ALGOrithmic Language)
ANN	Umělá neuronová síť (Artificial Neural Networks)
aPSO	Optimalizace profilu hejnem částic (aerofoil Particle Swarm Optimization)
ATR	Dopravní letoun (Aerei di Trasporto Regionale)
BFO	Optimalizace chování bakterií (Bacterial foraging optimization)
BP	Bezier-PARSEC (Bezier-PARSEC)
CAD	Počítačem podporované kreslení (computer-aided drafting)
CFD	Počítačová dynamika tekutin (Computational Fluid Dynamics)
CST	Parametrizace CST (Class Shape Function)
DE	Diferenciální evoluce (Differential Evolution)
DNA	Deoxyribonukleová kyselina (DeoxyriboNucleic Acid)
DNS	Metoda přímé simulace (Direct Numerical Simulation)
EA	Evoluční algoritmy (Evolutionary Algorithms)
FFD	Volná tvarová deformace (Free-Form Deformation)
GA	Genetické algoritmy (Genetic Algorithms)
IDE	Integrované vývojové prostředí (Integrated Development Environment)
IRACE	Autokonfigurační nástroj (Iterated Racing for Automatic Algorithm Configuration)
LES	Velká vířivá simulace (Large Eddy Simulation)
MKO	Metoda konečných objemů (Finite Volume Method)
MKP	Metoda konečných prvků (Finite Element Method)
MOL	Varianta optimalizace hejnem částic (Many Optimizing Liaisons)
NACA	Řada profilů (National Advisory Committee for Aeronautics)
NASA	Národní úřad pro letectví a vesmír (National Aeronautics and Space Administration)
PARSEC	Druh parametrizačního modelu (Parametric section)
PHO	Paralelní hybridní optimalizace (Parallel Hybrid Optimization)
PSO	Optimalizace hejnem částic (Particle Swarm Optimization)

RANS	Reynoldsem zprůměrované Navier-Stokesovy rovnice (Reynolds Averaged Navier-Stokes Equations)
RBF	Radiální bázové funkce (Radial Basis Functions)
SOMA	Samo Organizující se Migrační Algoritmus (Self-Organizing Migrating Algorithm)
SPSO	Standardní optimalizace hejnem částic (Standard Particle Swarm Optimization)

$actIter$	aktuální iterace
α_i	návrhové parametry parametrizace
$\alpha_i^{UP}, \alpha_i^{LO}$	proměnná pro horní a dolní povrch profilu
α	úhel náběhu, hladina významnosti, chyba 1. druhu
α_{max}	maximální úhel náběhu
α_{min}	minimální úhel náběhu
α_{step}	krok pro úhel náběhu
α_{te}	úhel střední křivky odtokové hrany
α_{TE}	úhel odtokové hrany
B_i^n	Bernsteinův polynom
β	chyba 2. druhu
β_{te}, β_{TE}	sklon odtokové hrany
c	rychlost zvuku, akcelerační koeficient, střed bázové funkce
c_1, c_2	akcelerační koeficienty (učící faktory)
C_D	součinitel odporu
C_L	součinitel vztlaku
C_M	klopivý moment
$C_{N2}^{N1}(\Psi)$	tvarová třída parametrizace
$b_0, b_2, b_8, b_{15}, b_{17}$	Bezier parametry
dz_{te}	polovina tloušťky odtokové hrany
$\Delta\alpha$	řídí modifikaci střední křivky k hornímu a spodnímu povrchu
$\Delta Z, \Delta Z_1, \Delta Z_2$	obecný, horní a dolní přírůstek tloušťky přidané k povrchu profilu
ΔZ_{TE}	tloušťka
e_i^{UP}, e_i^{LO}	vypočítaná konstanta pro horní a dolní povrch profilu
F, F_i, F_{ij}	spojité distribuční funkce
Φ	radiální bázová funkce
$gBest$	nejlepší nalezená pozice hejnem
G	spojitá distribuční funkce
γ	vířivé rozložení lineárně se měnící síly u panelové metody
γ_{le}	úhel směru náběžné hrany
H_0	nulová hypotéza
H_1	alternativní hypotéza
χ	omezující faktor

i	označení jednotlivé částic, konstanty pro Bernsteinův polynom
I	instance u autokonfiguratoru IRACE
j	číslo panelu u panelové metody
κ_c	zakřivení střední křivky
κ_t	zakřivení tloušťky
ξ_1, ξ_2	x souřadnice hloubky profilu
l	hloubka křídla
L_1, L_2	hloubky upraveného profilu
$Mach$	Machovo číslo
μ	dynamická viskozita, hodnota parametrizace, střední hodnota
n	stupeň Bezierovy křivky, hodnota parametrizace, počet hodnot
N	všechny optimalizované úhly náběhu, rozsah u mediánova testu
N_i, N_j, N_k	uzlové vektory parametrizace
$numIter$	celkový počet iterací
$\omega, \omega_{min}, \omega_{max}$	obecný, minimální a maximální parametr váhy setrvačnosti
$pBest_i$	nejlepší nalezená pozice částice i
$P(t)$	Bezierova křivka
Q_F	testová statistika Friedmanova testu
Q_K	testová statistika Kriskal – Wallisova testu
Q_M	testová statistika Mediánova testu
$Q_{i,j,k}$	řídící body parametrizace
$Q_{i,j,k}^{new}$	nové řídící body parametrizace
R_i, R_j, R_{ij}	pořadí u statistického testu
r, r_1, r_2	náhodné číslo v intervalu 0 až 1
Re	Reynoldsovo číslo
$r_{le}, r_{LEUP}, r_{LELO}$	obecné, horní a dolní poloměry křivosti náběžné hrany
ρ	hustota tekutiny
S_1, S_2	součet pořadí hodnot
S^+, S^-	součty kladných a záporných pořadí
$S(\psi)$	tvarová funkce parametrizace
σ	osměrodatná odchylka Gaussovy křivky
t	označení jednotlivé iterace
t_1	poloha maxima bázové funkce
t_2	tloušťku bázové funkce

T	kritická hodnota pro statistiku
U	kritická hodnota pro statistiku
θ	konfigurací parametrů u autokonfiguratoru IRACE
ν	rychlost pohybu tělesa, tekutiny, kinematická viskozita tekutiny
ν_i	rychlost částice i
ν_{min}, ν_{max}	minimální a maximální rychlost částic
x	bod v prostoru
x_c	vzdálenost maximálního prohnutí od náběžné hrany
x_i	pozice částice i
x_{min}, x_{max}	minimální a maximální pozice částic
x_t	vzdálenost maximální tloušťky od náběžné hrany
$X_{UP}, X_{LO}, Z_{UP}, Z_{LO}$	maxima obou povrchů
U_1, U_2	testovací statistiky pro Mann – Whitneův test
ν	kinematická viskozita tekutiny
y_c	maximální prohnutí
y_t	maximální tloušťka
Z^{UP}, Z^{LO}	horní a dolní povrch profilu
Z_{te}	svislé posunutí odtokové hrany
Z_{TE}	zetová souřadnice
Z_{XXUP}, Z_{XXLO}	reciproční křivosti (druhá derivace) maxima obou povrchů

Příloha A - Seznam publikací

A1 - Seznam vlastních publikací

MULLER, Jan. *Improving Initial Aerofoil Geometry Using Aerofoil Particle Swarm Optimisation*. MENDEL. 28, 1 (Jun. 2022), 63-67. DOI:<https://doi.org/10.13164/mendel.2022.1.063>.

MULLER, Jan. *Radical Reduction of Aircraft Fuel Consumption by Optimizing Aerofoil by New Evolutionary Algorithms*. Preprints, 2021, 2021060219 (doi: 10.20944/preprints202106.0219.v1).

A2 - Publikace v přípravě (under revision 3rd round)

MULLER, Jan. *Radical Reduction of Aerofoil Drag by the Optimization Technique*. Scientific Reports. Springer Nature. 2022. London, UK.

A3 - Tutoriály na mezinárodní konferenci Mendel

MULLER, Jan. *Improving wing geometry, Part I.: Mechanical design*. International Conference on Soft Computing MENDEL 2018. Brno, Czech Republic, 2018.

MULLER, Jan a Pavel OSMERA. *IMPROVING AEROFOIL GEOMETRY PART II.: OPTIMALIZATION*. International Conference on Soft Computing MENDEL 2019. Brno, Czech Republic, 2019.

Příloha B - Seznam patentů

MÜLLER, Jan a Rudolf MÜLLER. *Zařízení pro kontinuální a definovanou změnu geometrie tvaru a zakřivení profilu křidel*. ČR. Patent 300 728. Uděleno 29.07.2009.

MÜLLER, Jan. *Systém zabezpečení přemístitelných předmětů fixací jejich polohy*. ČR. Patent 297 511. Uděleno 03.01.2007.

MÜLLER, Jan a Rudolf MÜLLER. *Vícekomorová dvouplášťová kubická nádrž*. ČR. Patent 296 976. Uděleno 16.08.2006.

MÜLLER, Jan a Rudolf MÜLLER. *Vícestupňový absorpční separátor*. ČR. Patent 299 890. Uděleno 29.12.2008.

Příloha C - Výběr částí zdrojových kódů

Funkce evoluční optimalizace

```
1.def fctEvolutionaryAlgorithmsOptimization():
2. fctTSRandomize()
3.
4. for varIterationIndex in range(TSVariables.varTSIterationLength):
5.     varPopulation = fctObtainPopulationFirstHalf()
6.
7.     varOptimizationLength = TSVariables.varTSCrossoverLength
8.     for varIndex in range(varOptimizationLength):
9.         fctTSCrossover(TSVariables.varTSPopulationHalfLength + varIndex, 6, 7,varPopulation)
10.
11.     varOptimizationStart = varOptimizationLength
12.     varOptimizationLength = varOptimizationStart + TSVariables.varTSMutationLength
13.     for varIndex in range(varOptimizationStart, varOptimizationLength):
14.         fctTSMutation(TSVariables.varTSPopulationHalfLength + varIndex)
15.
16.     varOptimizationStart = varOptimizationLength
17.     varOptimizationLength = varOptimizationStart + TSVariables.varTSRandomLength
18.     for varIndex in range(varOptimizationStart, varOptimizationLength):
19.         fctTSRandom(TSVariables.varTSPopulationHalfLength + varIndex)
20.
21.     fctRewritePopulationFirstHalf()
```

Účelová funkce evoluční optimalizace

```
1.def fctFitnessLiftDragExtreme(parLogFileName, parAlphaMin, parAlphaMax):
2. try:
3.     varXfoilPath = os.path.abspath(r'..\XFoil')
4.     os.chdir(varXfoilPath)
5.
6.     varLogFile = open(parLogFileName + '.log', 'r')
7.     varLines = varLogFile.readlines()
8.     varLogFile.close
9.
10.    varLinesLength = len(varLines) - 12
11.    if(varLinesLength > 12):
12.        varWords = str.split(varLines[12])
13.        varStartAlpha = float(varWords[0])
14.
15.        varWords = str.split(varLines[len(varLines) - 1])
16.        varEndAlpha = float(varWords[0])
17.
18.        if(varStartAlpha == parAlphaMin and varEndAlpha == parAlphaMax):
19.            varFileAlphas, varFileLifts, varFileDrags, varFileMoments = \
20.                fctFromLogFileToArray(parLogFileName)
21.
22.            varExtremeLiftDrag = 0
23.
24.            for varFileIndex in range(len(varFileAlphas)):
25.                varFileLift = varFileLifts[varFileIndex]
26.                varFileDrag = varFileDrags[varFileIndex]
27.                varFileLiftDrag = varFileLift/varFileDrag
28.
29.                if(varFileLiftDrag > varExtremeLiftDrag):
30.                    varExtremeLiftDrag = varFileLiftDrag
31.
32.            return varExtremeLiftDrag
33.
34.    return None
35.
36.except:
36.
37.return None
```

Funkce optimalizace s výchozím profilem

```
1. def fctInitailAerofoilOptimization():
2.   varOmega = -0.2
3.   varC2 = 2.8
4.
5.   varBestFitness = None
6.   gBest = None
7.
8.   varParticles = [varParticle() for varIndex in range(0, IAVariables.varIASwarmLength)]
9.
10.  for varT in range(IAVariables.varIAIterationLength):
11.
12.    for varParticle in enumerate(varParticles):
13.      varFitness = None
14.
15.      while not varFitness:
16.        if varBestFitness:
17.          varParticle.fctUpdate(varOmega, varC2, gBest)
18.
19.          fctCreateLogFile(varParticle)
20.          varFitness = fctFitnessDragLiftAverage()
21.
22.          if not varFitness or not varBestFitness:
23.            varParticle.fctRandomize()
24.
25.      if varFitness < varBestFitness or not varBestFitness:
26.        varBestFitness = varFitness
27.        gBest = copy(varParticle.varControlPoints)
```

Účelová funkce optimalizace s výchozím profilem

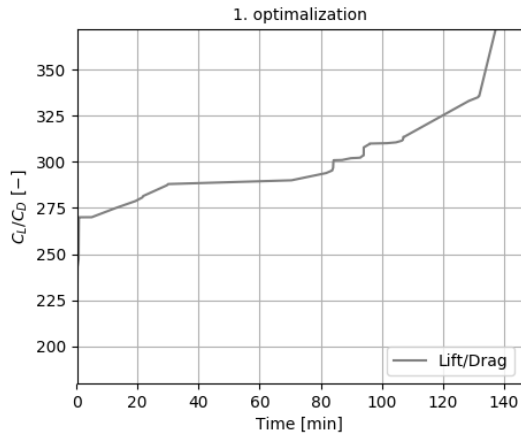
```
1. def fctFitnessDragLiftAverage():
2. try:
3.     varXfoilPath = os.path.abspath(r'..\XFoil')
4.     os.chdir(varXfoilPath)
5.
6.     varLogFile = open('Xfoil.log', 'r')
7.     varLines = varLogFile.readlines()
8.     varLogFile.close
9.
10.    varLinesLength = len(varLines)
11.    if(varLinesLength > 12):
12.        varWords = str.split(varLines[12])
13.        varStartAlpha = float(varWords[0])
14.        varStartDrag = float(varWords[2])
15.
16.        varWords = str.split(varLines[len(varLines) - 1])
17.        varEndAlpha = float(varWords[0])
18.        varEndDrag = float(varWords[2])
19.
20.        varStartDragLimit = 0.026
21.        varEndDragLimit = 0.026
22.
23.        if((varStartAlpha == IAVariables.varIAAlphaMin and varEndAlpha ==
24.        IAVariables.varIAAlphaMax) and (varStartDrag < varStartDragLimit and
25.        varEndDrag < varEndDragLimit)):
26.            varFileAlphas, varFileLifts, varFileDrags, varFileMoments = \
27.                fctFromLogFileToArray(varLines)
28.
29.            varFirstAlpha = 3
30.            varSecondAlpha = 9
31.            varStepAlpha = 1
32.            varLengthAlpha = int(((varSecondAlpha - varFirstAlpha)/varStepAlpha) + 1)
33.
34.            varAverageAlphas = numpy.linspace(varFirstAlpha, varSecondAlpha, num=varLengthAlpha)
35.            varAverageLifts = fctOuterXPointsToInnerYPoints(varAverageAlphas, varFileAlphas,
36.                varFileLifts)
37.            varAverageDrags = fctOuterXPointsToInnerYPoints(varAverageAlphas, varFileAlphas,
38.                varFileDrags)
39.
40.            varSumLift = 0.0
41.            varSumDrag = 0.0
42.            varSumLength = len(varAverageAlphas)
43.
44.            for varSumIndex in range(varSumLength):
45.                varSumLift += varAverageLifts[varSumIndex]
46.                varSumDrag += varAverageDrags[varSumIndex]
47.
48.            varAverageDragLift = varSumDrag/varSumLift
49.
50.            if(varAverageDragLift > IAVariables.varIALastAverageValue):
51.                IAVariables.varIALastAverageValue = varAverageDragLift
52.
53.            varAverageValue = varAverageDragLift
54.            return varAverageValue
55.
56.    return None
57.
58. return None
59.
60. except:
61.    return None
```

Funkce výpočtů aerodynamických vlastností výpočetním programem Xfoil

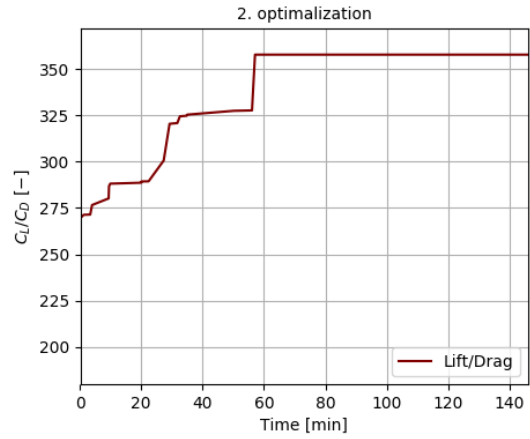
```
1. def fctXfoilExecuteCommands(self):
2.     varXfoilPath = os.path.abspath(r'..\XFOIL')
3.     os.chdir(varXfoilPath)
4.
5.     try:
6.         os.remove(self.varLogFileName + '.log')
7.     except:
8.         pass
9.
10.    try:
11.        self.varProcess=subprocess.Popen('xfoil.exe',stdin=subprocess.PIPE,stderr=subprocess.PIPE,
12.            stdout=subprocess.PIPE)
13.        self.varProcess.stderr.close()
14.
15.        # Hide window
16.        if(self.varHideGraphics):
17.            self.fctProcessCommand('PLOP')
18.            self.fctProcessCommand('G F')
19.            self.fctProcessCommand(' ')
20.        if(self.varAirfoil != 0):
21.            self.fctProcessCommand(self.varAirfoil)
22.        else:
23.            self.fctProcessCommand('LOAD ' + self.varDatFileName + '.dat')
24.        self.fctProcessCommand('PANE')
25.        self.fctProcessCommand('OPER')
26.        self.fctProcessCommand('VPAR')
27.        self.fctProcessCommand('N ' + str(self.varNcrit))
28.        self.fctProcessCommand(' ')
29.        self.fctProcessCommand('VISC ' + str(self.varRe))
30.        self.fctProcessCommand('PACC')
31.        self.fctProcessCommand(self.varLogFileName + '.log')
32.        self.fctProcessCommand(' ')
33.        self.fctProcessCommand('ASEQ ' + str(self.varAlphaMin) + ' ' + str(self.varAlphaMax) +
34.            ' ' + str(self.varAlphaStep))
35.        self.fctProcessCommand(' ')
36.        self.fctProcessCommand('QUIT')
37.
38.        self.varProcess.stdout.close()
39.        self.varProcess.stdin.close()
40.        self.varProcess.wait()
41.
42.    except:
43.        pass
```

Příloha D - Výsledky optimalizací

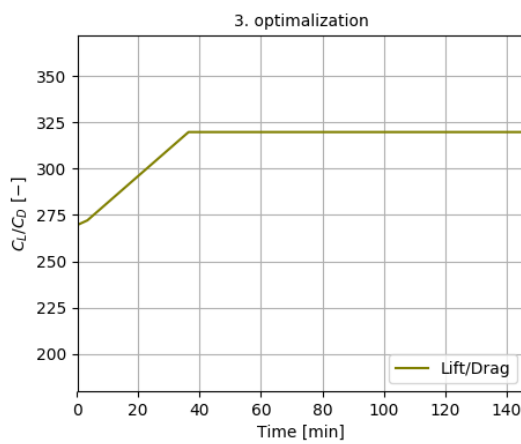
Optimalizace aEA o různých velikostech turnajového výběru



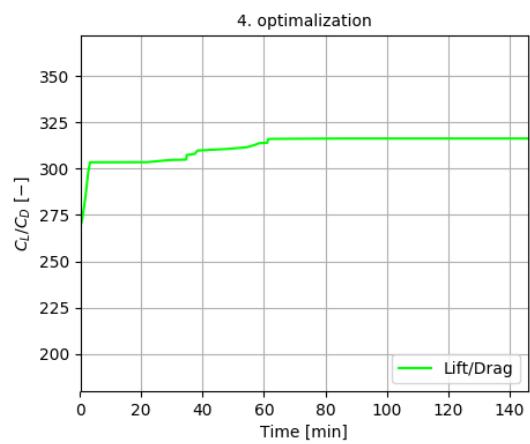
Obr. 1: aEA velikost turnaje 3 1/10.



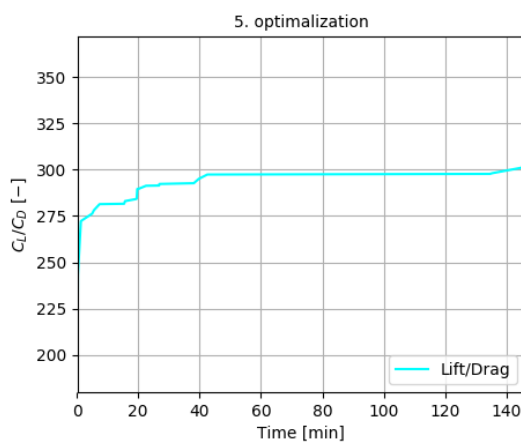
Obr. 2: aEA velikost turnaje 3 2/10.



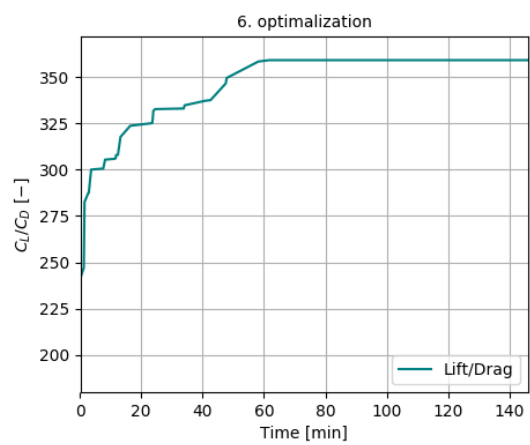
Obr. 3: aEA velikost turnaje 3 3/10.



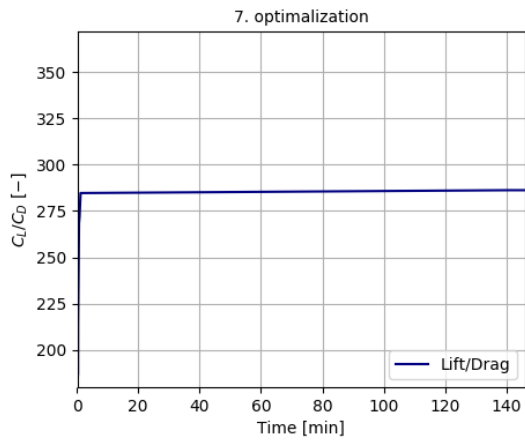
Obr. 4: aEA velikost turnaje 3 4/10.



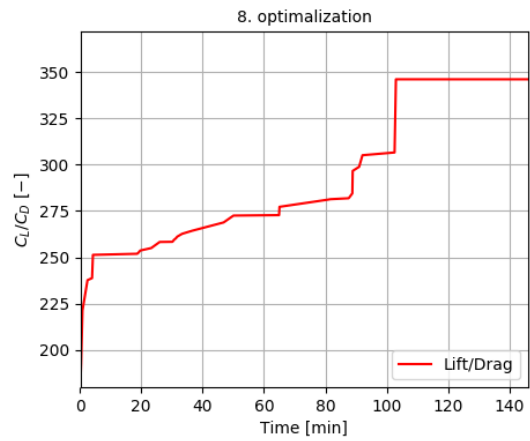
Obr. 5: aEA velikost turnaje 3 5/10.



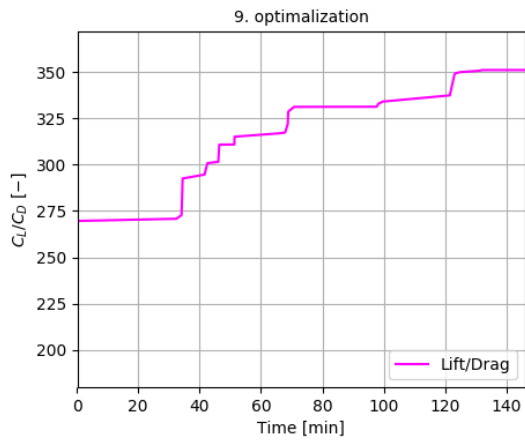
Obr. 6: aEA velikost turnaje 3 6/10.



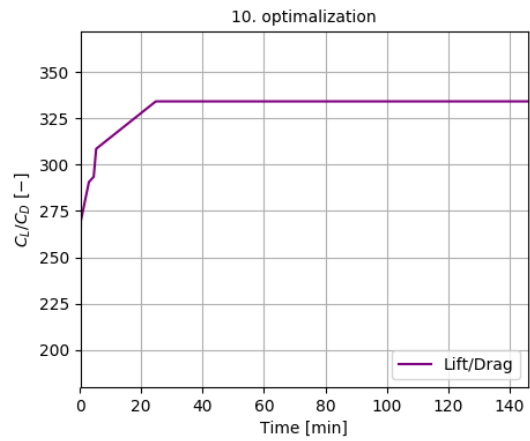
Obr. 7: aEA velikost turnaje 3 7/10.



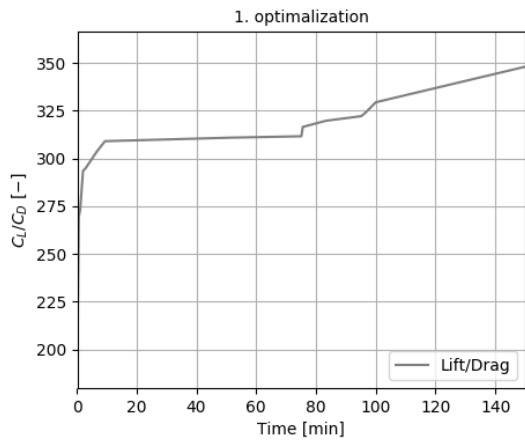
Obr. 8: aEA velikost turnaje 3 8/10.



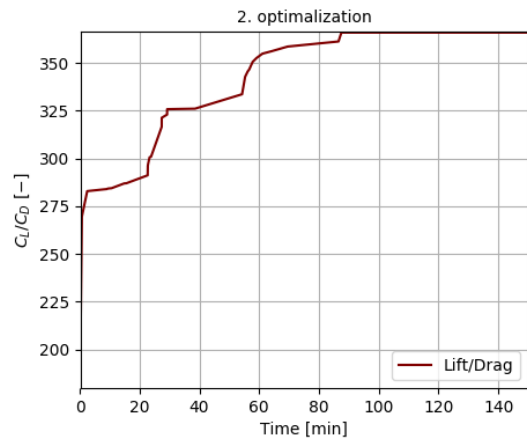
Obr. 9: aEA velikost turnaje 3 9/10.



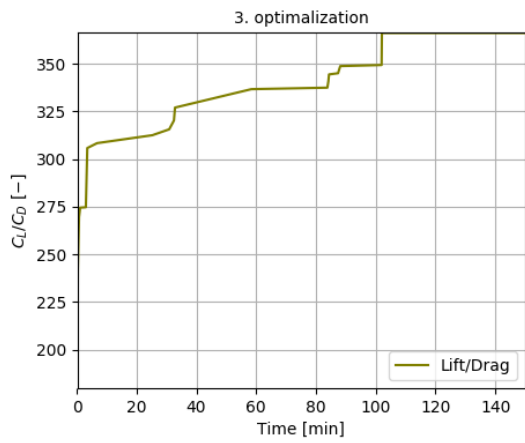
Obr. 10: aEA velikost turnaje 3 10/10.



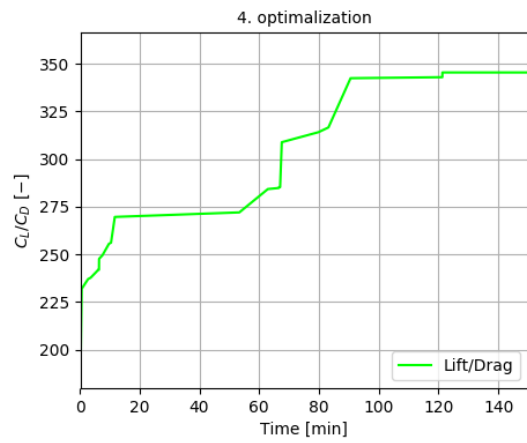
Obr. 11: aEA velikost turnaje 6 1/10.



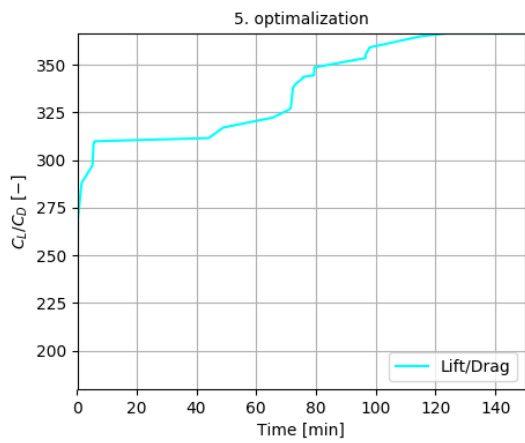
Obr. 12: aEA velikost turnaje 6 2/10.



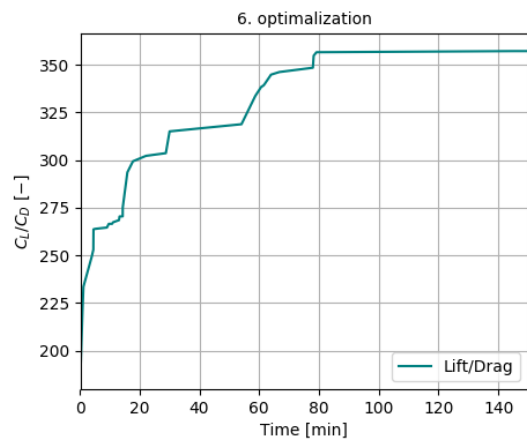
Obr. 13: aEA velikost turnaje 6 3/10.



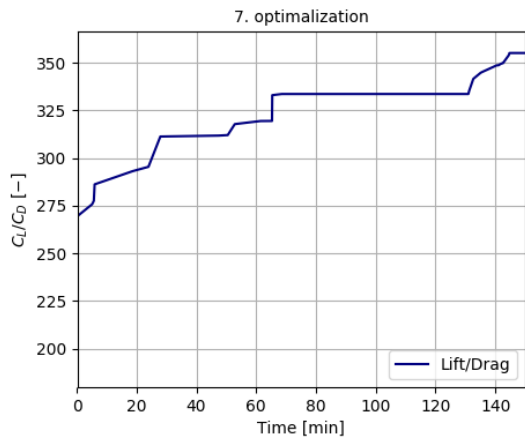
Obr. 14: aEA velikost turnaje 6 4/10.



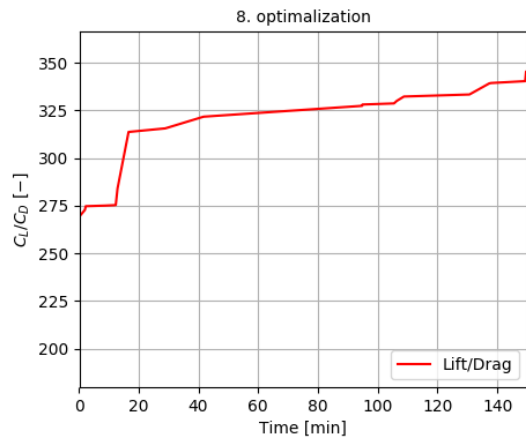
Obr. 15: aEA velikost turnaje 6 5/10.



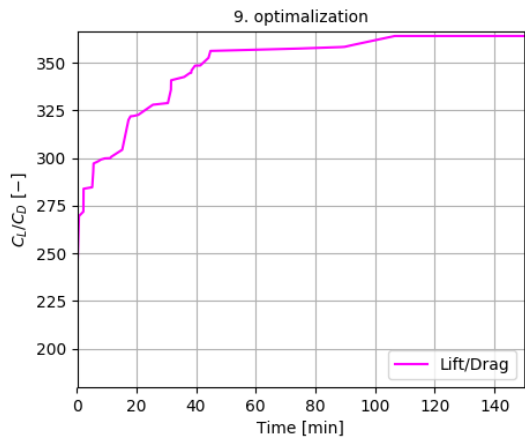
Obr. 16: aEA velikost turnaje 6 6/10.



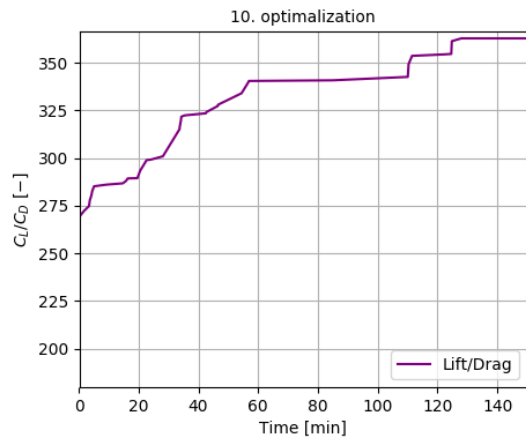
Obr. 17: aEA velikost turnaje 6 7/10.



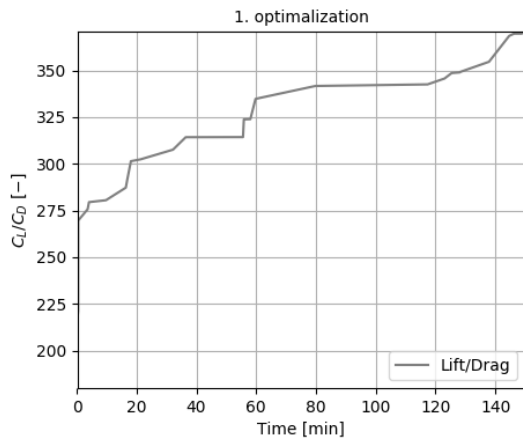
Obr. 18: aEA velikost turnaje 6 8/10.



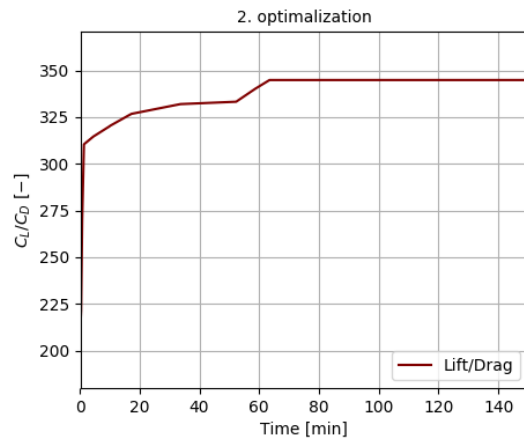
Obr. 19: aEA velikost turnaje 6 9/10.



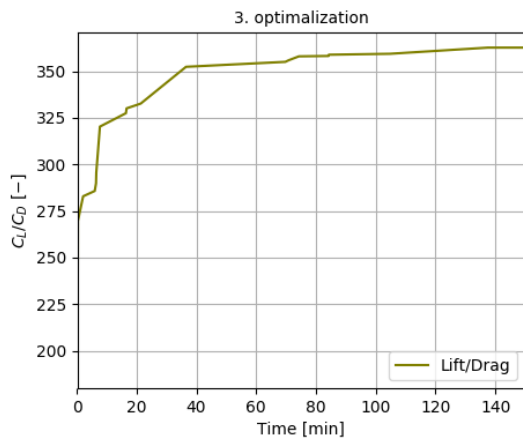
Obr. 20: aEA velikost turnaje 6 10/10.



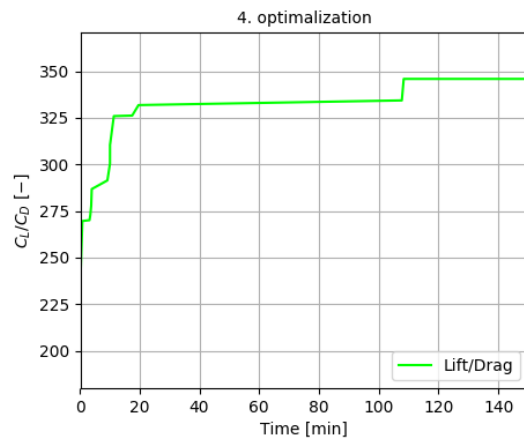
Obr. 21: aEA velikost turnaje 9 1/10.



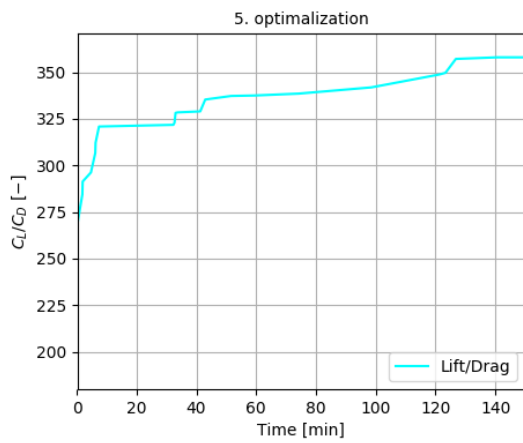
Obr. 22: aEA velikost turnaje 9 2/10.



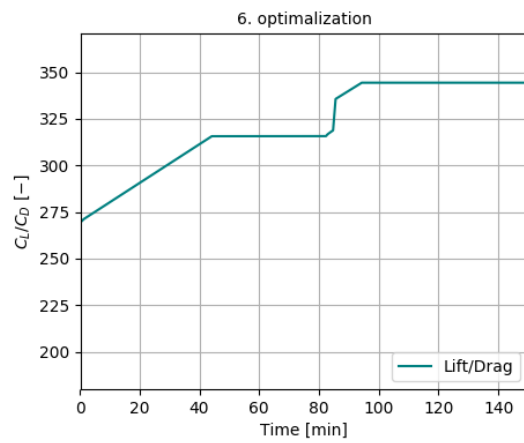
Obr. 23: aEA velikost turnaje 9 3/10.



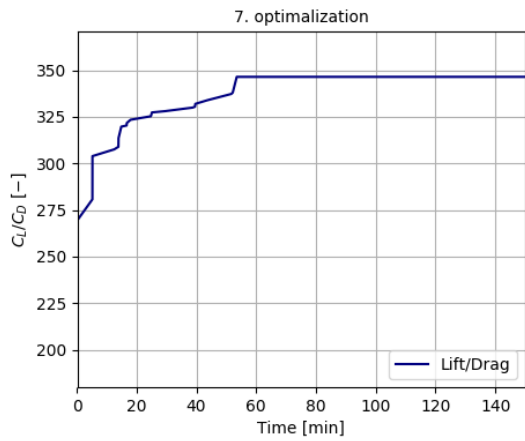
Obr. 24: aEA velikost turnaje 9 4/10.



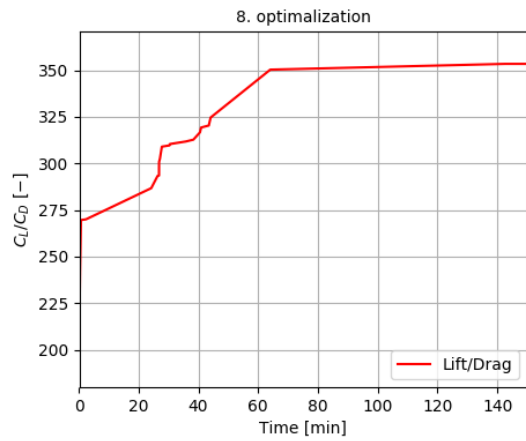
Obr. 25: aEA velikost turnaje 9 5/10.



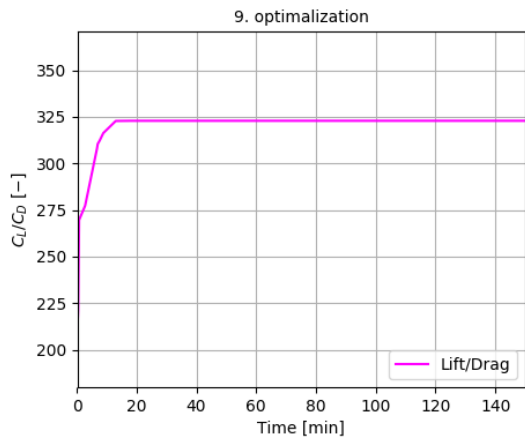
Obr. 26: aEA velikost turnaje 9 6/10.



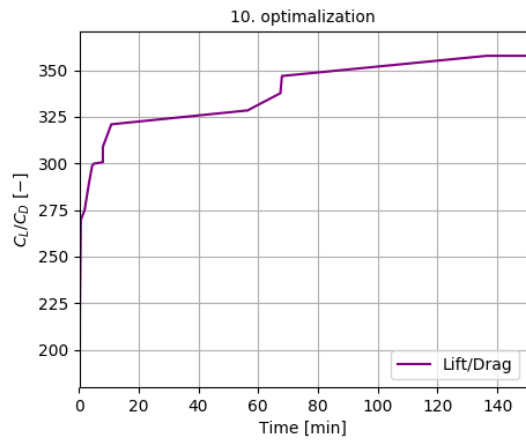
Obr. 27: aEA velikost turnaje 9 7/10.



Obr. 28: aEA velikost turnaje 9 8/10.

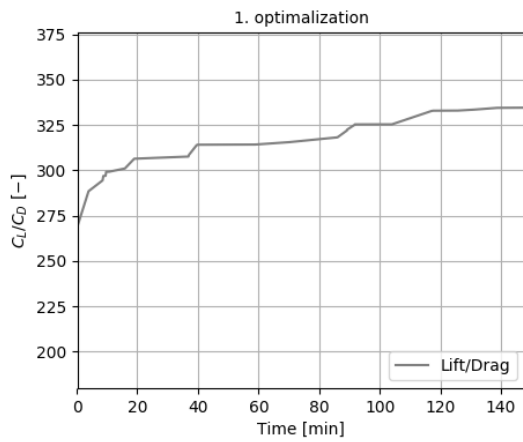


Obr. 29: aEA velikost turnaje 9 9/10.

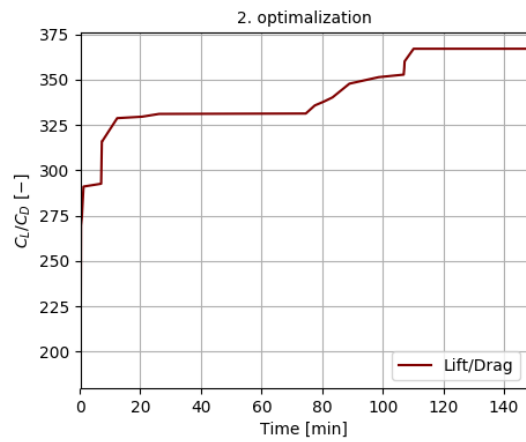


Obr. 30: aEA velikost turnaje 9 10/10.

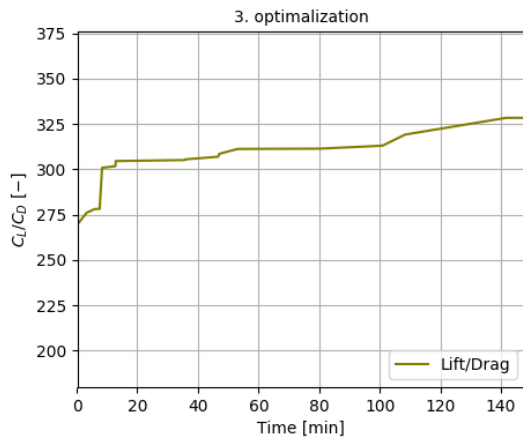
Optimalizace aEA se znovu vkládáním nejlepších jedinců



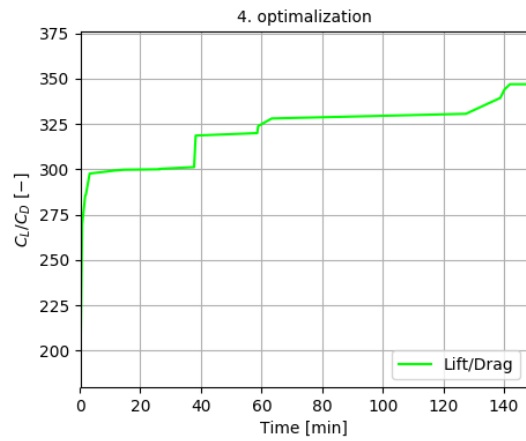
Obr. 31: aEA vkládání po 10 min. 1/10.



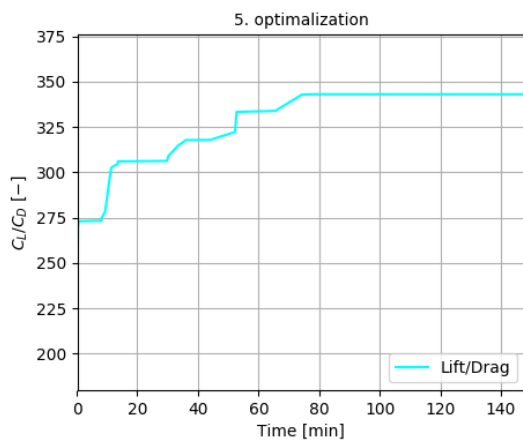
Obr. 32: aEA vkládání po 10 min. 2/10.



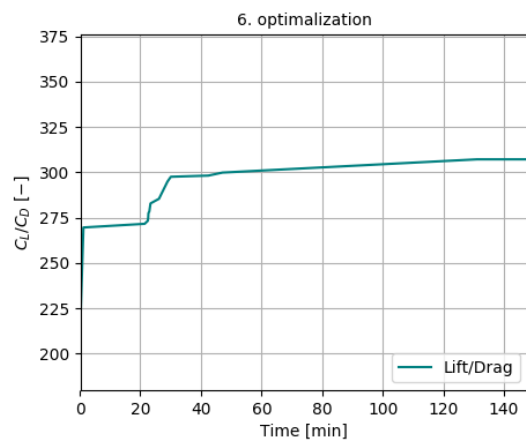
Obr. 33: aEA vkládání po 10 min. 3/10.



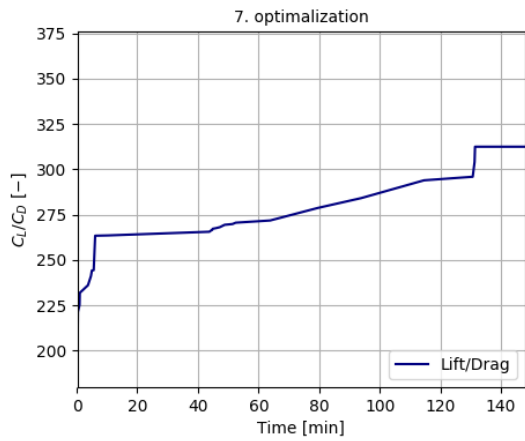
Obr. 34: aEA vkládání po 10 min. 4/10.



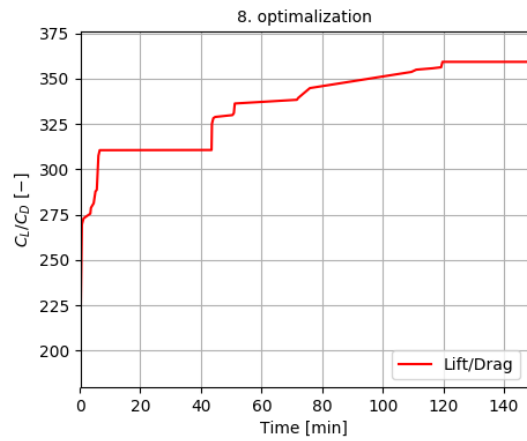
Obr. 35: aEA vkládání po 10 min. 5/10.



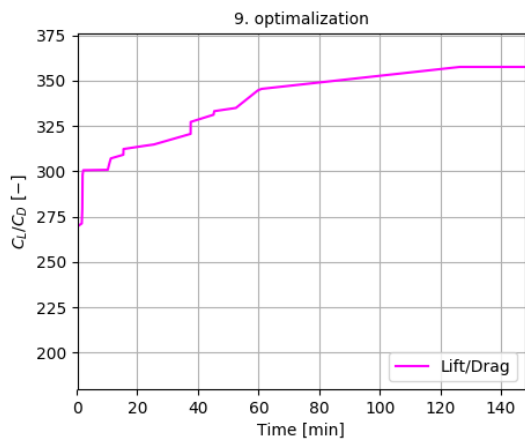
Obr. 36: aEA vkládání po 10 min. 6/10.



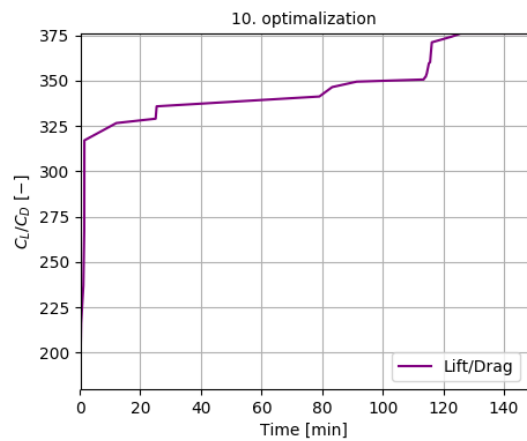
Obr. 37: aEA vkládání po 10 min. 7/10.



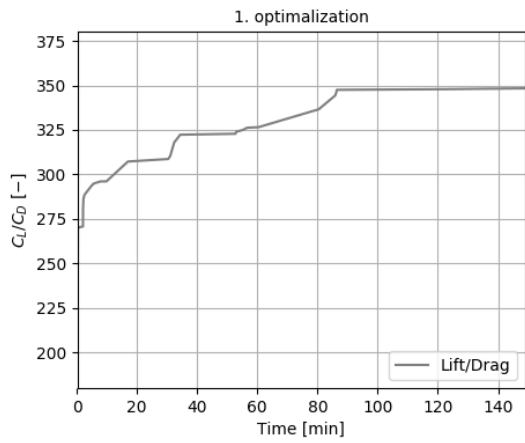
Obr. 38: aEA vkládání po 10 min. 8/10.



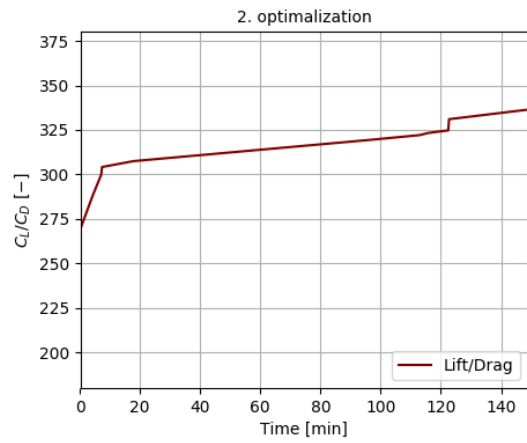
Obr. 39: aEA vkládání po 10 min. 9/10.



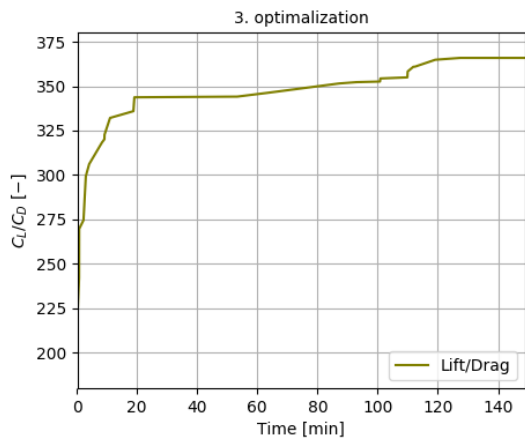
Obr. 40: aEA vkládání po 10 min. 10/10.



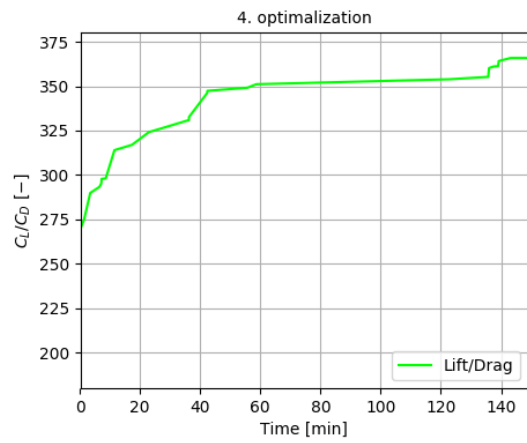
Obr. 41: aEA vkládání po 20 min. 1/10.



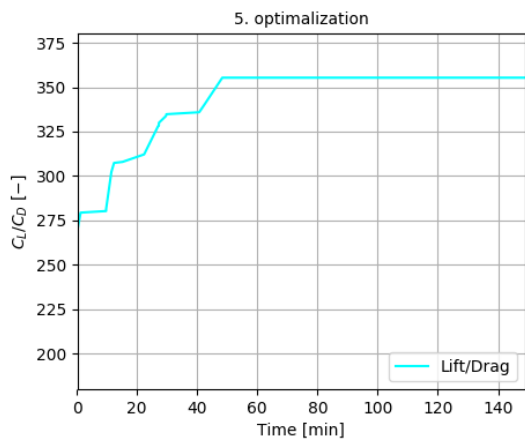
Obr. 42: aEA vkládání po 20 min. 2/10.



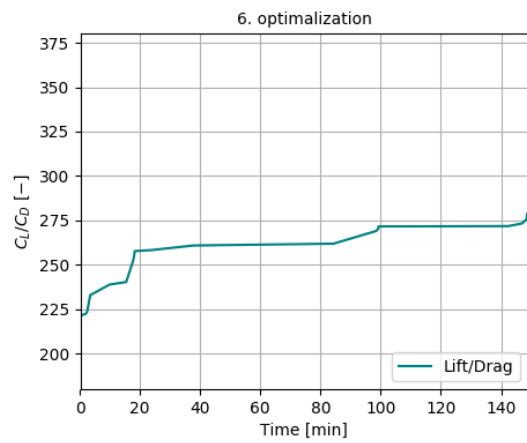
Obr. 43: aEA vkládání po 20 min. 3/10.



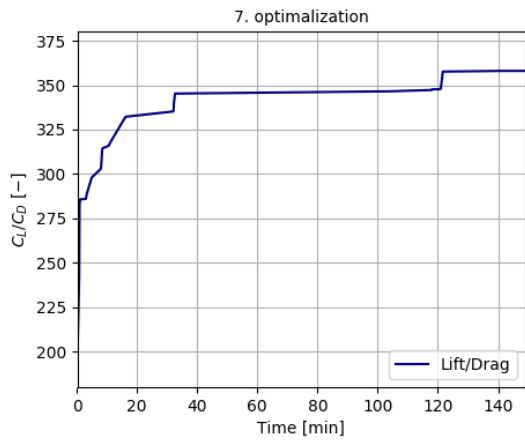
Obr. 44: aEA vkládání po 20 min. 4/10.



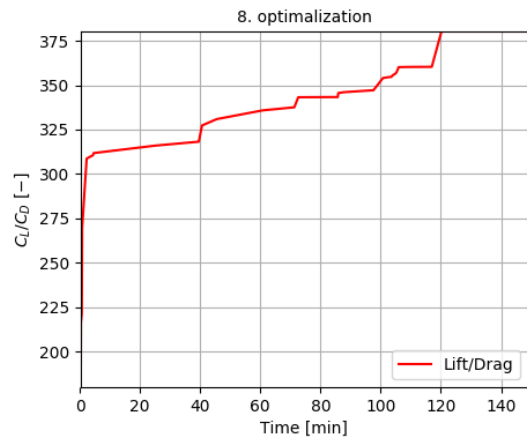
Obr. 45: aEA vkládání po 20 min. 5/10.



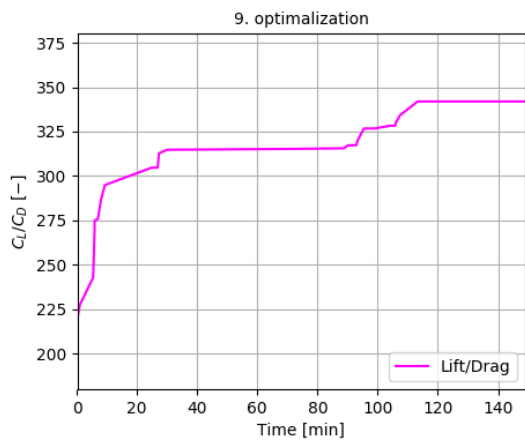
Obr. 46: aEA vkládání po 20 min. 6/10.



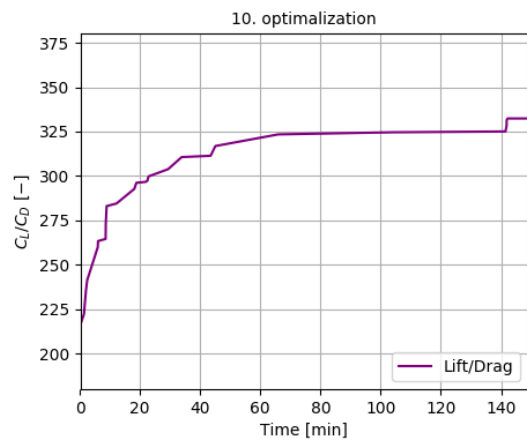
Obr. 47: aEA vkládání po 20 min. 7/10.



Obr. 48: aEA vkládání po 20 min. 8/10.

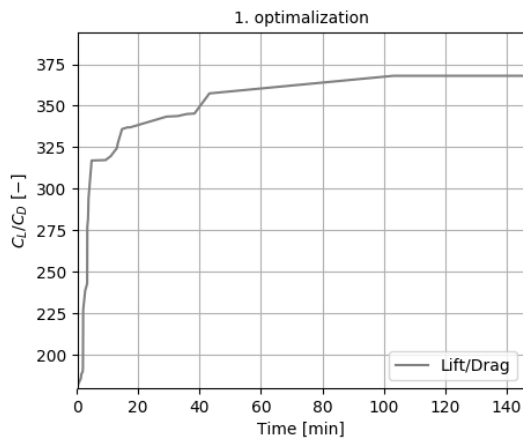


Obr. 49: aEA vkládání po 20 min. 9/10.

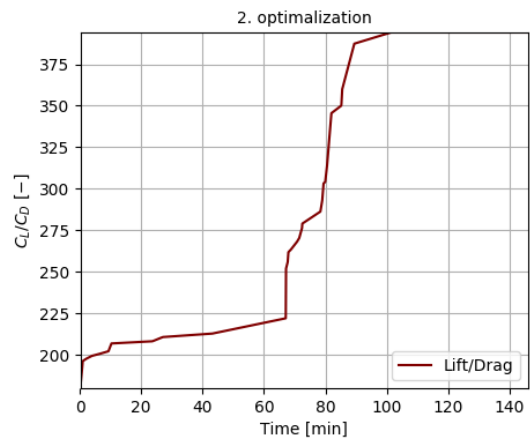


Obr. 50 aEA vkládání po 20 min. 10/10.

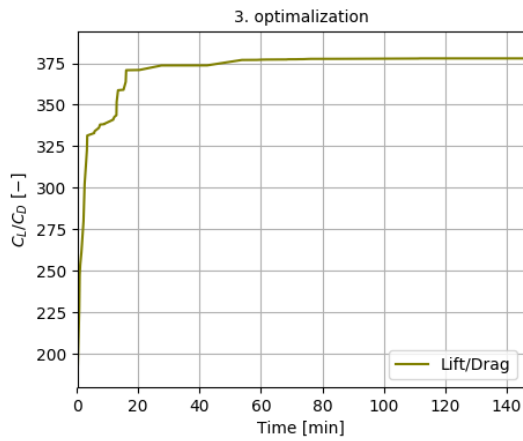
Optimalizace aPSO o různých nastaveních hodnot konstant



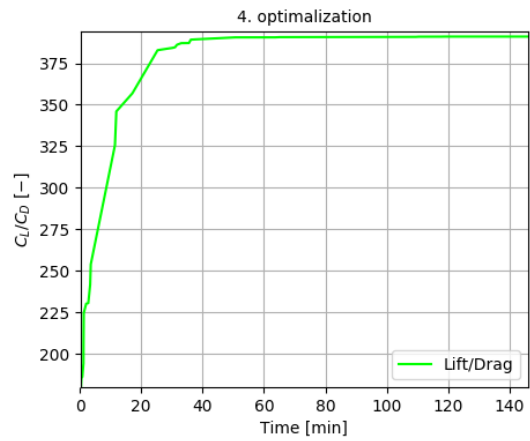
Obr. 51: aPSO ω -0.2 a c_2 2.8 1/10.



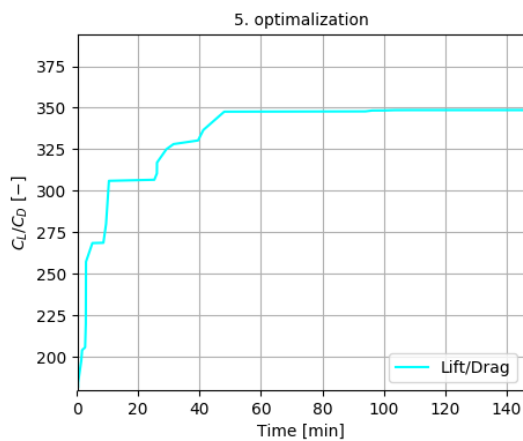
Obr. 52: aPSO ω -0.2 a c_2 2.8 2/10.



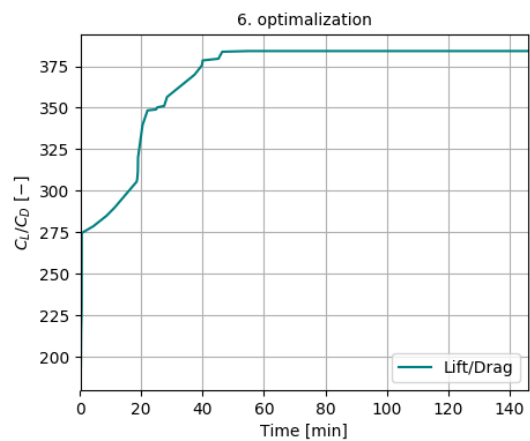
Obr. 53: aPSO ω -0.2 a c_2 2.8 3/10.



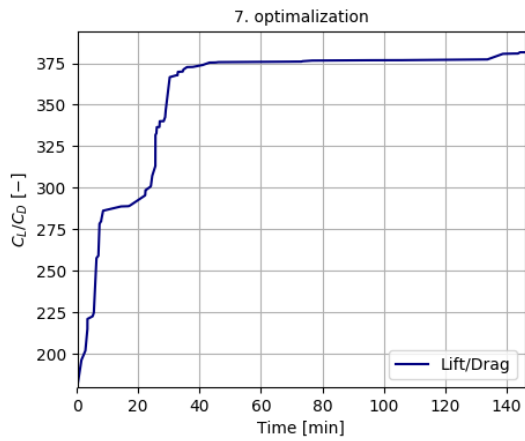
Obr. 54: aPSO ω -0.2 a c_2 2.8 4/10.



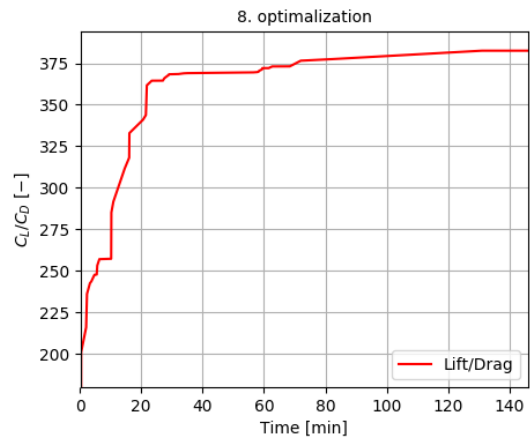
Obr. 55: aPSO ω -0.2 a c_2 2.8 5/10.



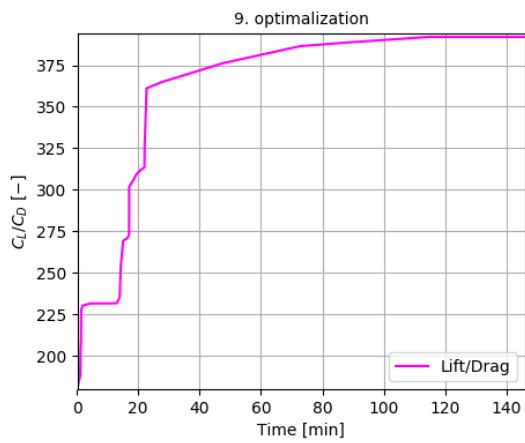
Obr. 56: aPSO ω -0.2 a c_2 2.8 6/10.



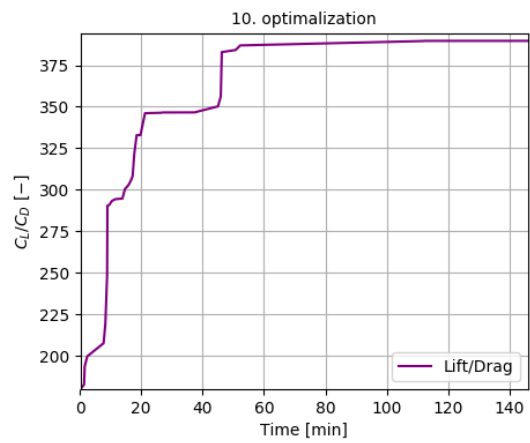
Obr. 57: aPSO ω -0.2 a c_2 2.8 7/10.



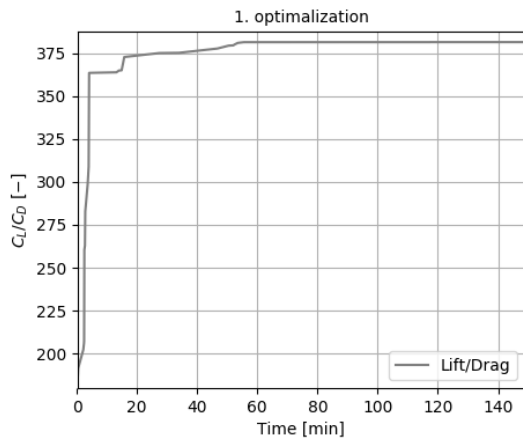
Obr. 58: aPSO ω -0.2 a c_2 2.8 8/10.



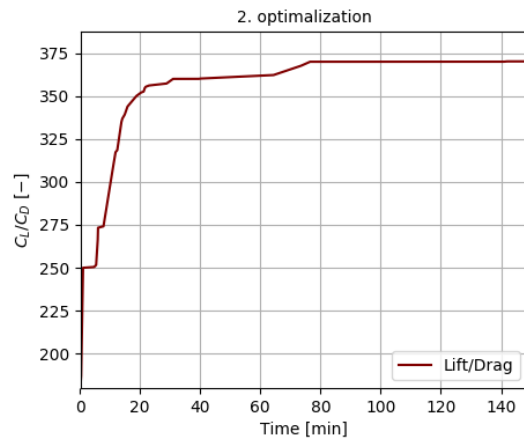
Obr. 59: aPSO ω -0.2 a c_2 2.8 9/10.



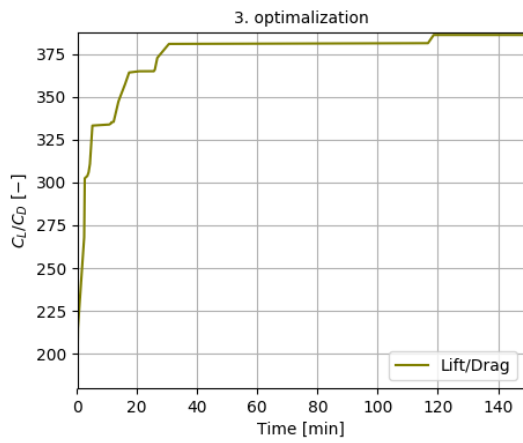
Obr. 60: aPSO ω -0.2 a c_2 2.8 10/10.



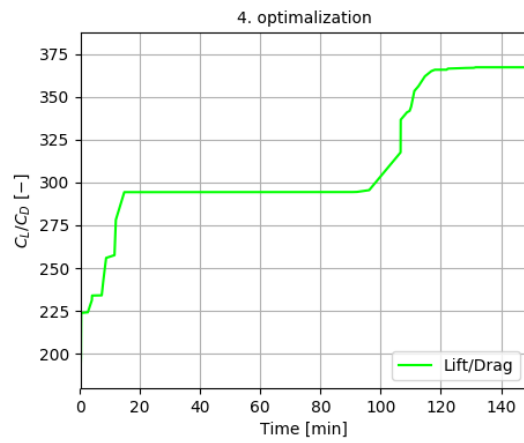
Obr. 61: aPSO ω -0.3328 a c_2 2.8446 1/10.



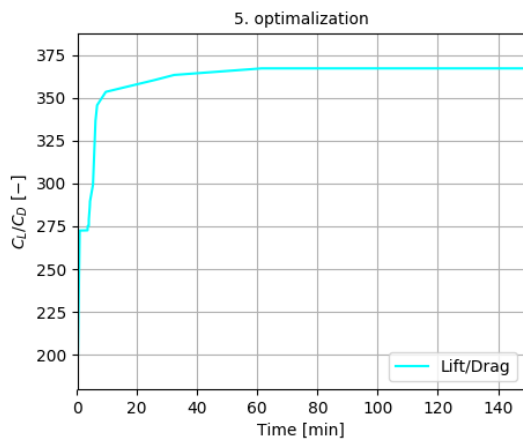
Obr. 62: aPSO ω -0.3328 a c_2 2.8446 2/10.



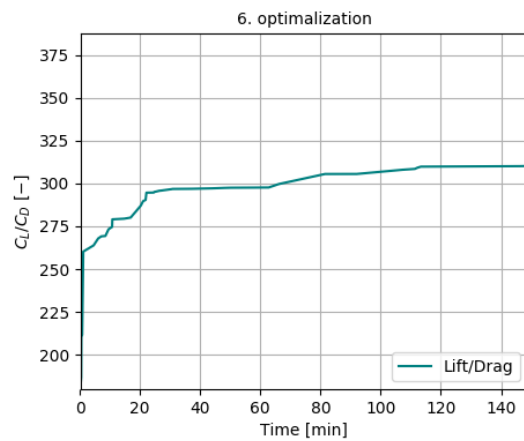
Obr. 63: aPSO ω -0.3328 a c_2 2.8446 3/10.



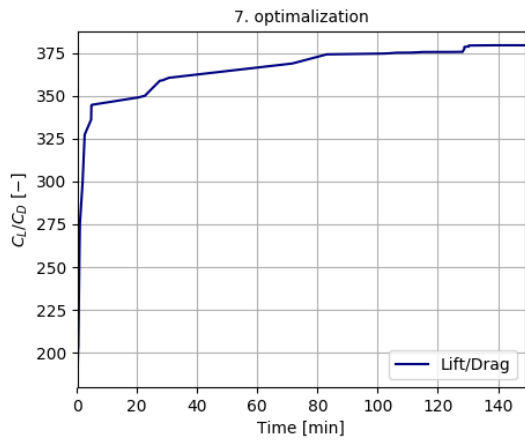
Obr. 64: aPSO ω -0.3328 a c_2 2.8446 4/10.



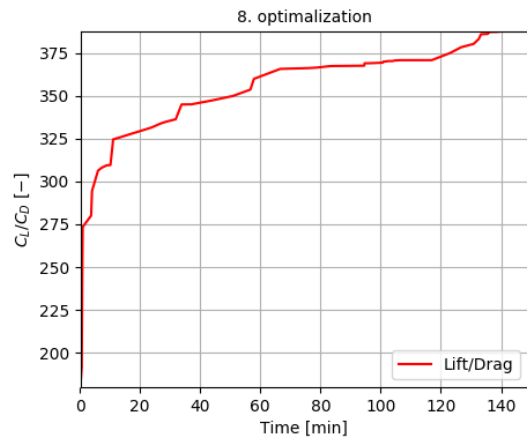
Obr. 65: aPSO ω -0.3328 a c_2 2.8446 5/10.



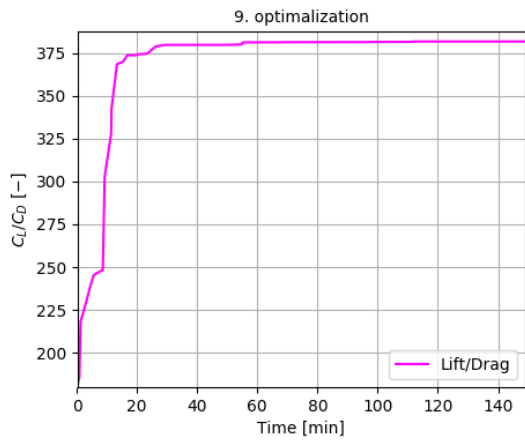
Obr. 66: aPSO ω -0.3328 a c_2 2.8446 6/10.



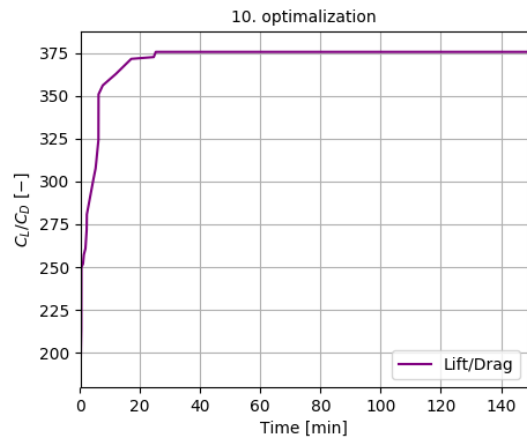
Obr. 67: aPSO ω -0.3328 a c_2 2.8446 7/10.



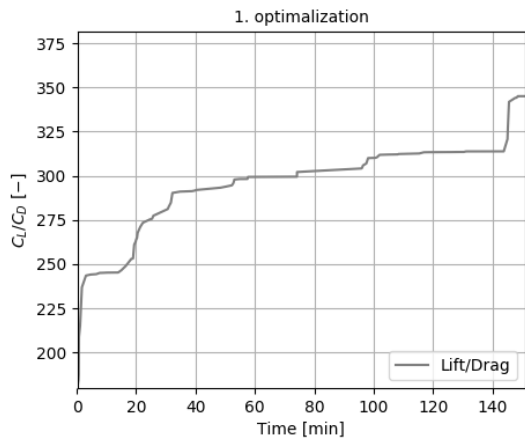
Obr. 68: aPSO ω -0.3328 a c_2 2.8446 8/10.



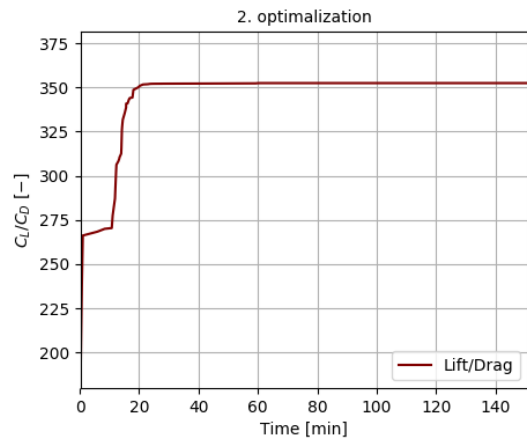
Obr. 69: aPSO ω -0.3328 a c_2 2.8446 9/10.



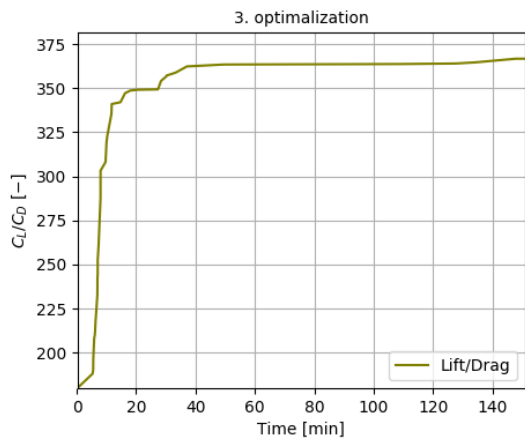
Obr. 70: aPSO ω -0.3328 a c_2 2.8446 10/10.



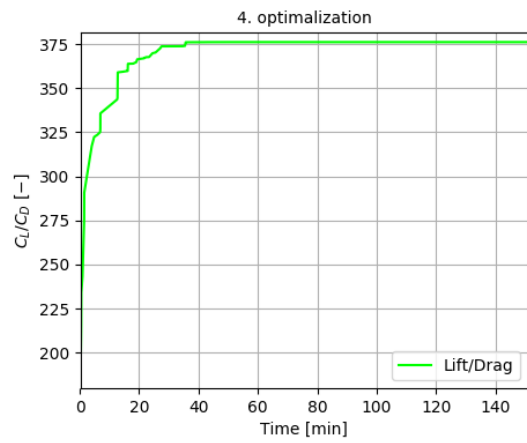
Obr. 71: aPSO ω -0.3085 a c_2 2.0273 1/10.



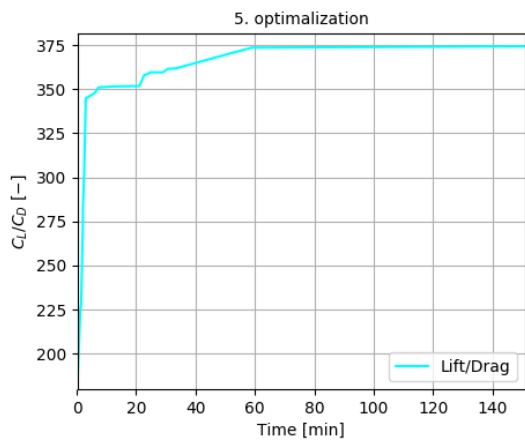
Obr. 72: aPSO ω -0.3085 a c_2 2.0273 2/10.



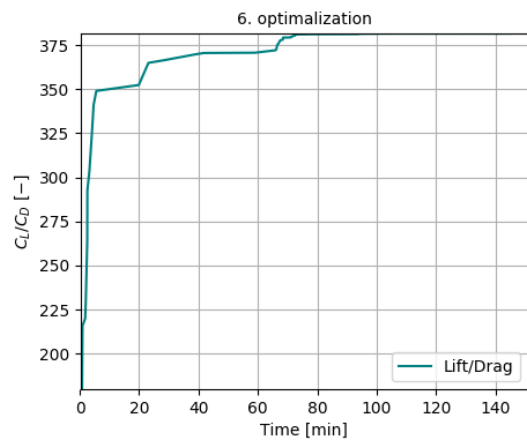
Obr. 73: aPSO ω -0.3085 a c_2 2.0273 3/10.



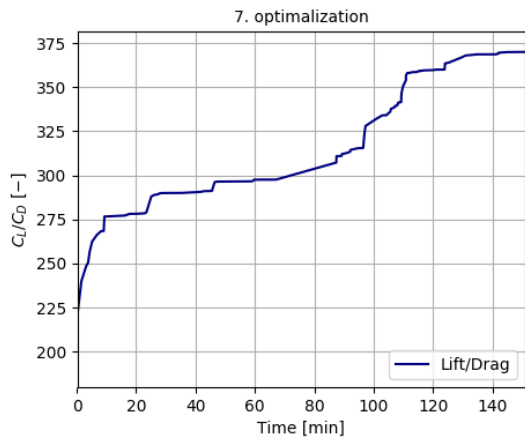
Obr. 74: aPSO ω -0.3085 a c_2 2.0273 4/10.



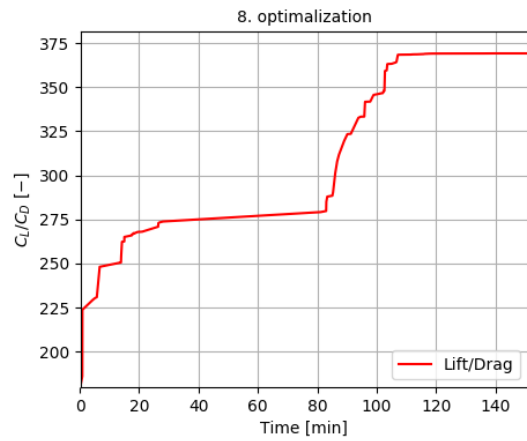
Obr. 75: aPSO ω -0.3085 a c_2 2.0273 5/10.



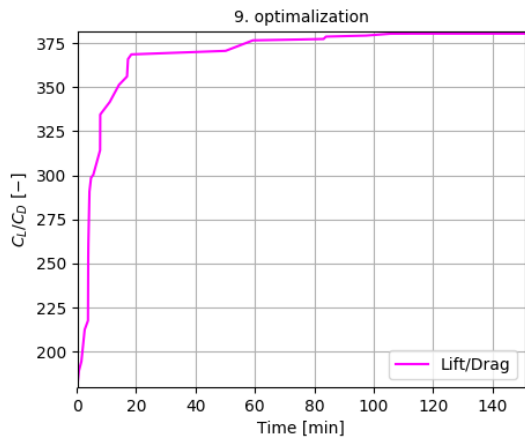
Obr. 76: aPSO ω -0.3085 a c_2 2.0273 6/10.



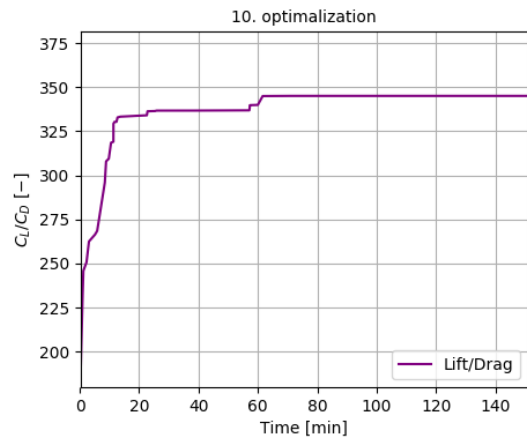
Obr. 77: aPSO ω -0.3085 a c_2 2.0273 7/10.



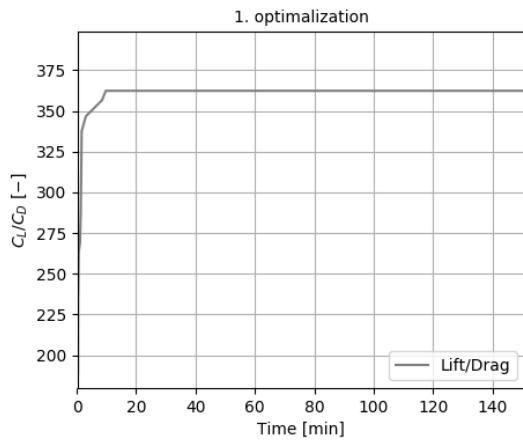
Obr. 78: aPSO ω -0.3085 a c_2 2.0273 8/10.



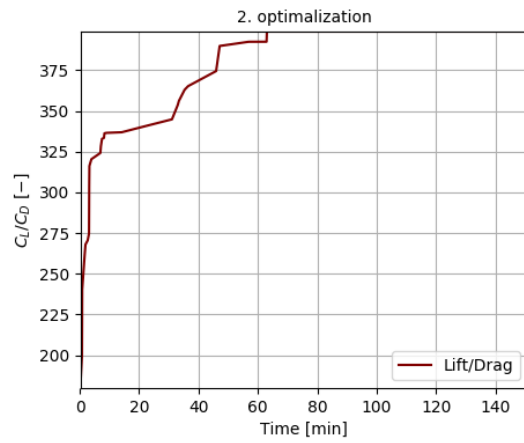
Obr. 79: aPSO ω -0.3085 a c_2 2.0273 9/10.



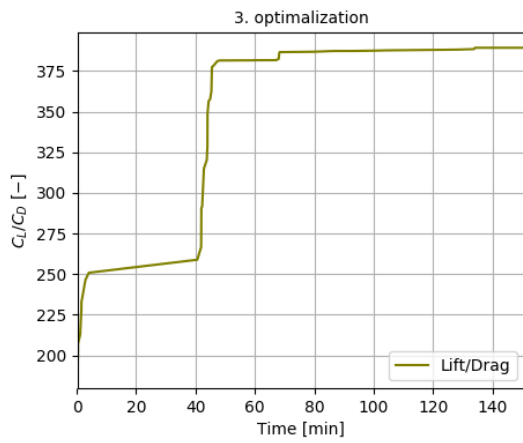
Obr. 80: aPSO ω -0.3085 a c_2 2.0273 10/10.



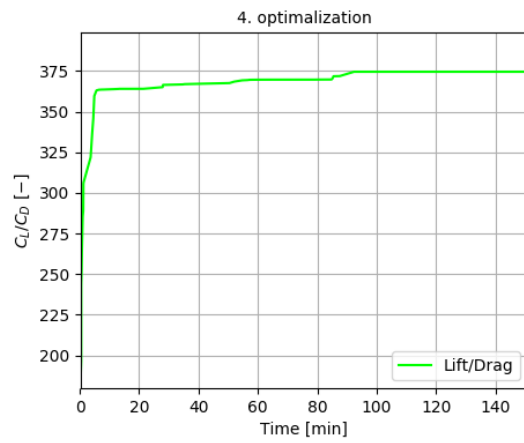
Obr. 81: aPSO ω 0.2840 a c_2 1.9466 1/10.



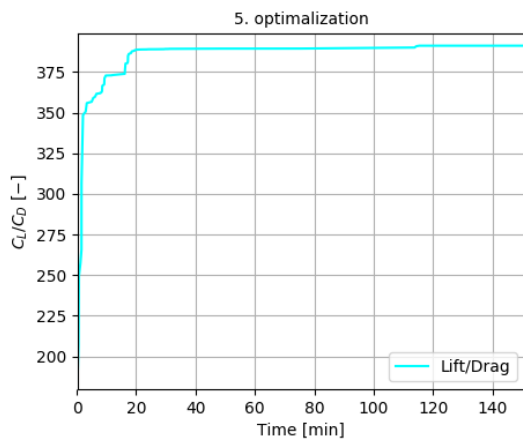
Obr. 82: aPSO ω 0.2840 a c_2 1.9466 2/10.



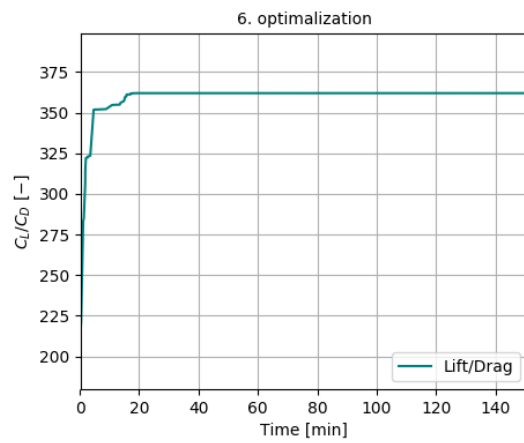
Obr. 83: aPSO ω 0.2840 a c_2 1.9466 3/10.



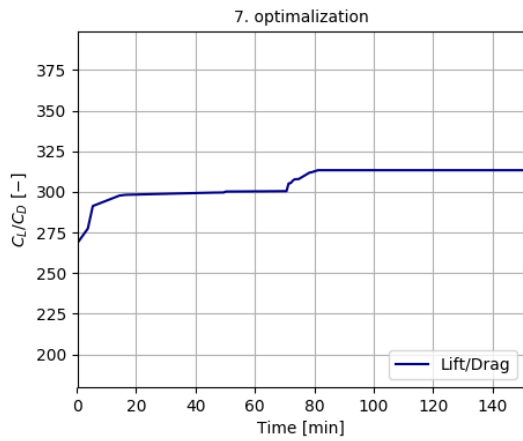
Obr. 84: aPSO ω 0.2840 a c_2 1.9466 4/10.



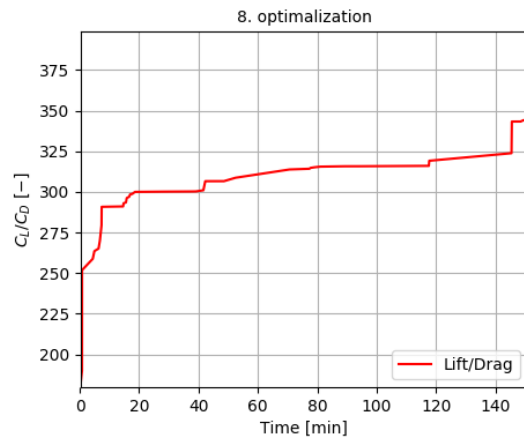
Obr. 85: aPSO ω 0.2840 a c_2 1.9466 5/10.



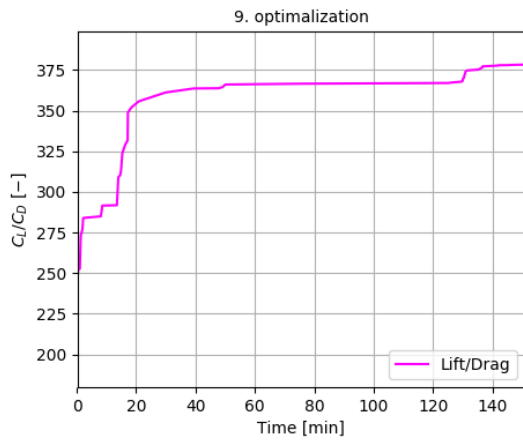
Obr. 86: aPSO ω 0.2840 a c_2 1.9466 6/10.



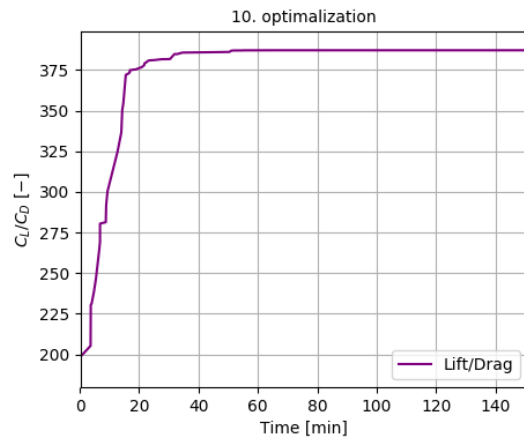
Obr. 87: aPSO ω 0.2840 a c_2 1.9466 7/10.



Obr. 88: aPSO ω 0.2840 a c_2 1.9466 8/10.

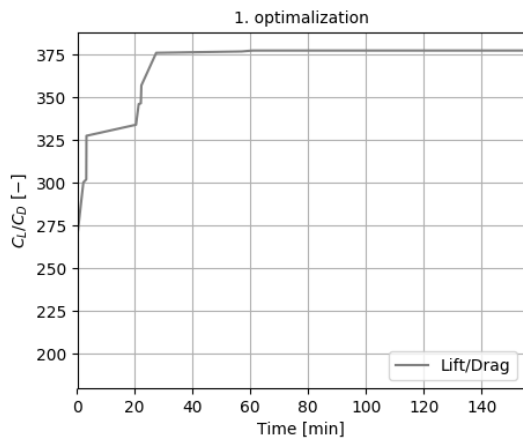


Obr. 89: aPSO ω 0.2840 a c_2 1.9466 9/10.

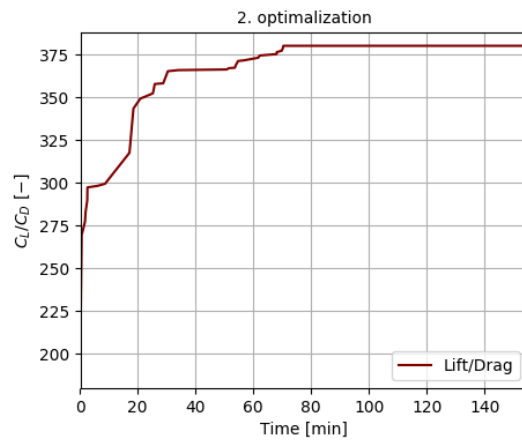


Obr. 90: aPSO ω 0.2840 a c_2 1.9466 10/10.

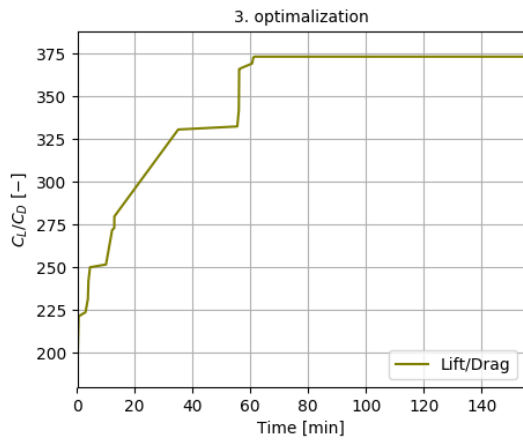
Různé časové intervaly střídání optimalizací aEA a aPSO



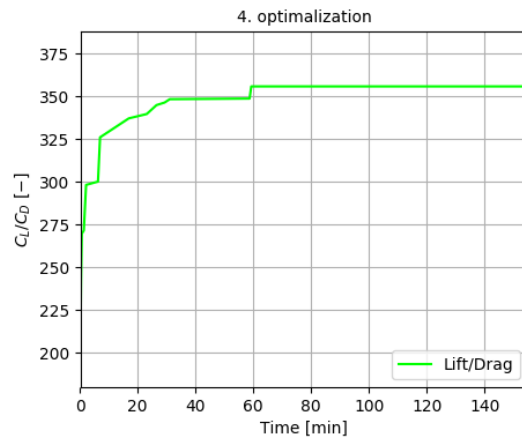
Obr. 91: PHO střídání po 15 min. 1/10.



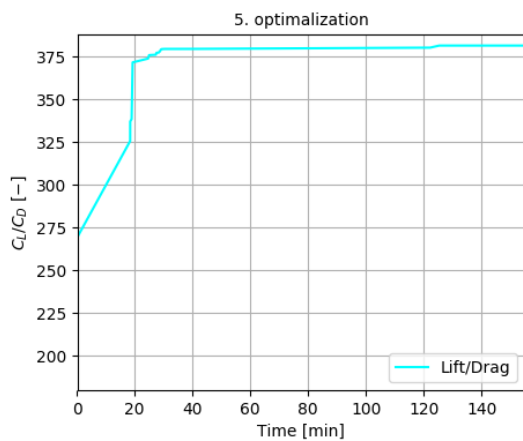
Obr. 92: PHO střídání po 15 min. 2/10.



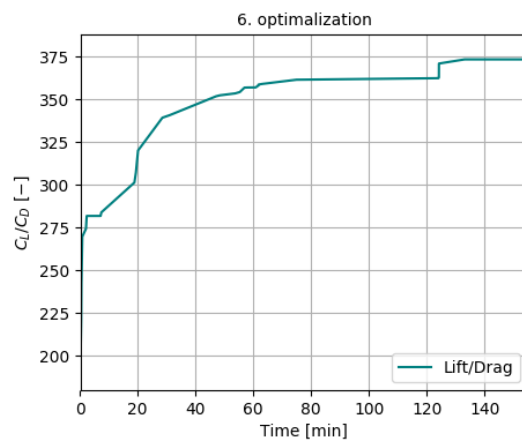
Obr. 93: PHO střídání po 15 min. 3/10.



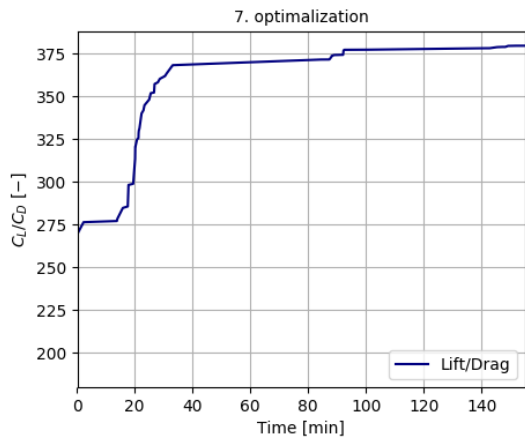
Obr. 94: PHO střídání po 15 min. 4/10.



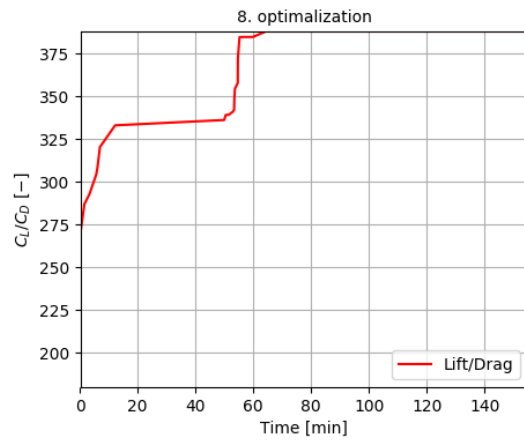
Obr. 95: PHO střídání po 15 min. 5/10.



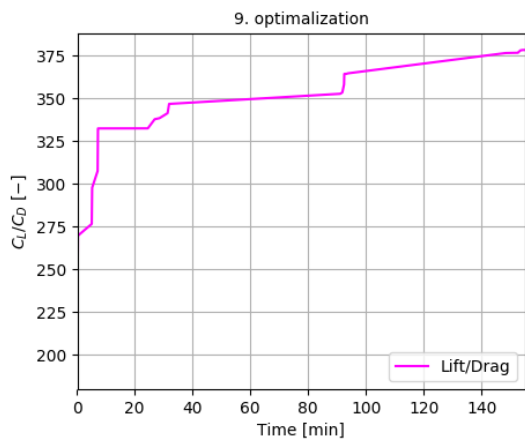
Obr. 96: PHO střídání po 15 min. 6/10.



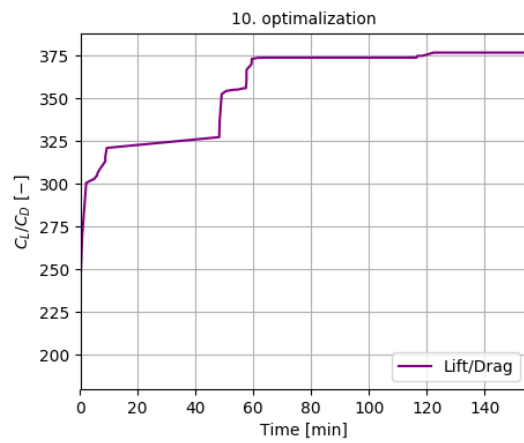
Obr. 97: PHO střídání po 15 min. 7/10.



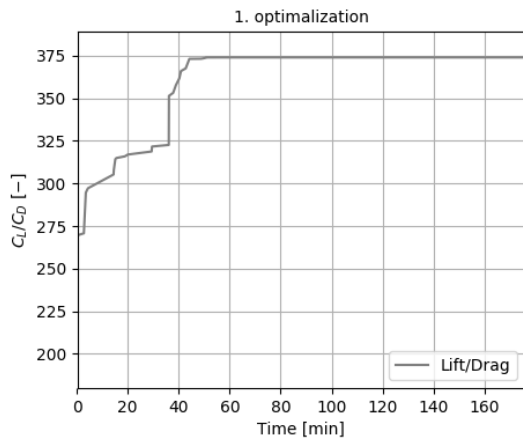
Obr. 98: PHO střídání po 15 min. 8/10.



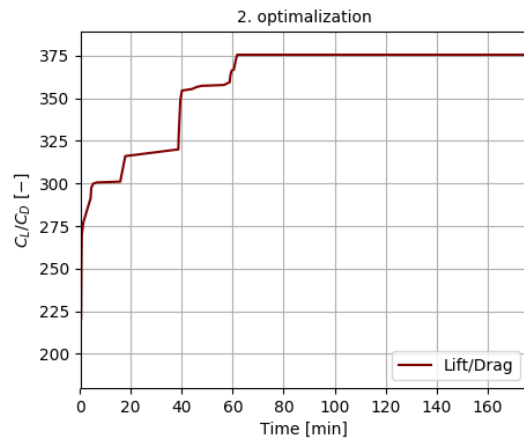
Obr. 99: PHO střídání po 15 min. 9/10.



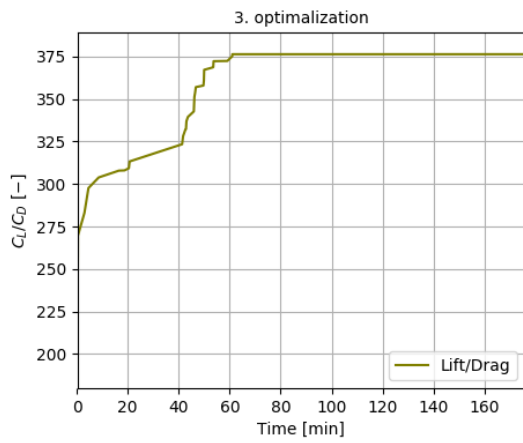
Obr. 100: PHO střídání po 15 min. 10/10.



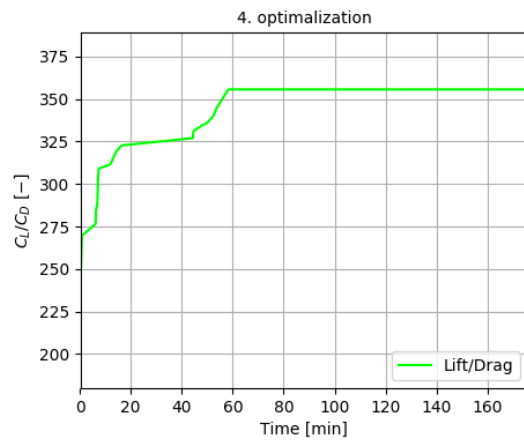
Obr. 101: PHO střídání po 30 min. 1/10.



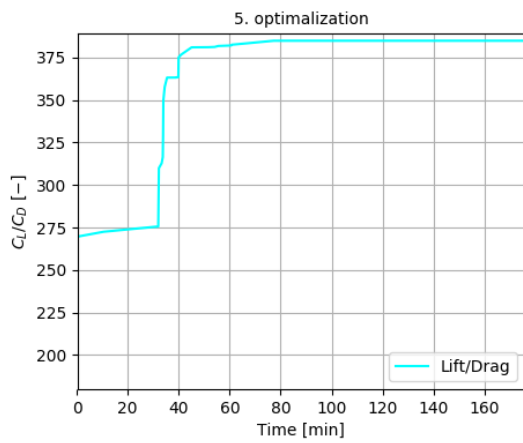
Obr. 102: PHO střídání po 30 min. 2/10.



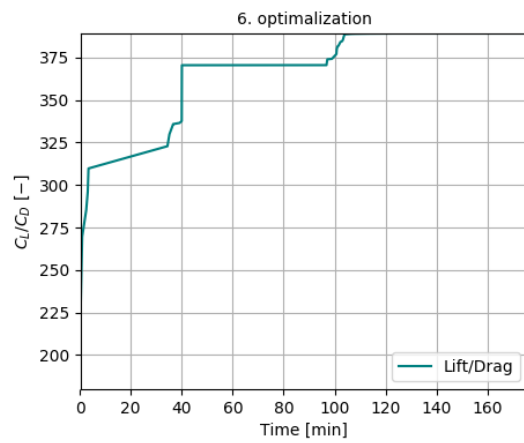
Obr. 103: PHO střídání po 30 min. 3/10.



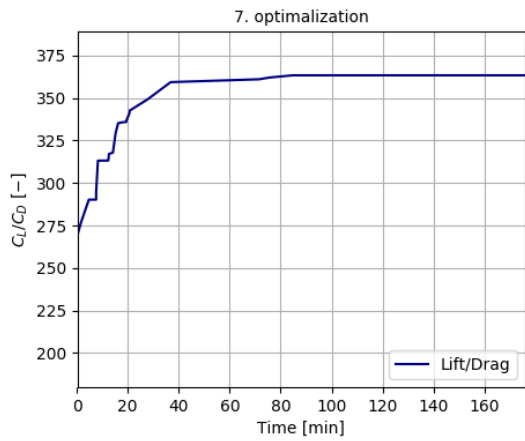
Obr. 104: PHO střídání po 30 min. 4/10.



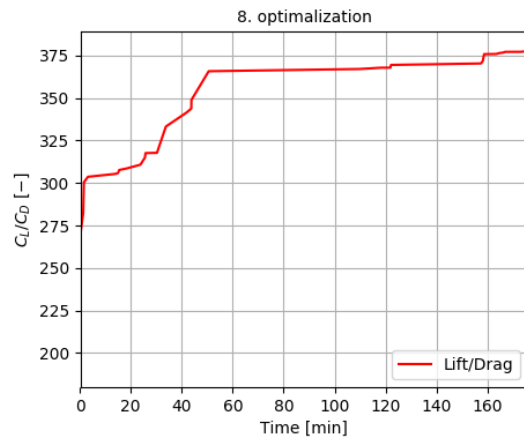
Obr. 105: PHO střídání po 30 min. 5/10.



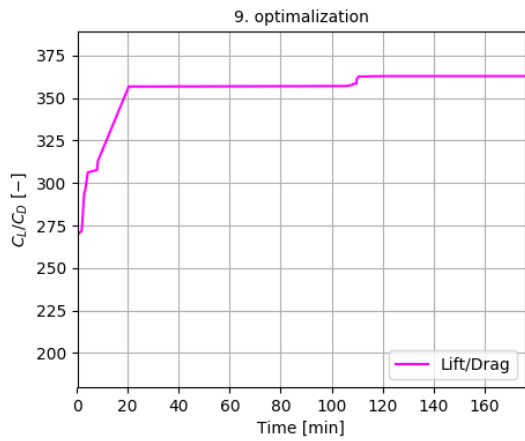
Obr. 106: PHO střídání po 30 min. 6/10.



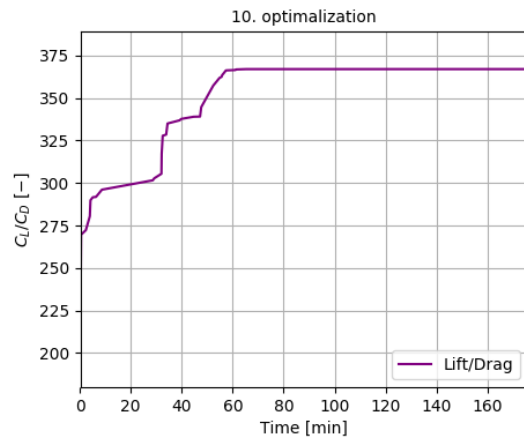
Obr. 107: PHO střídání po 30 min. 7/10.



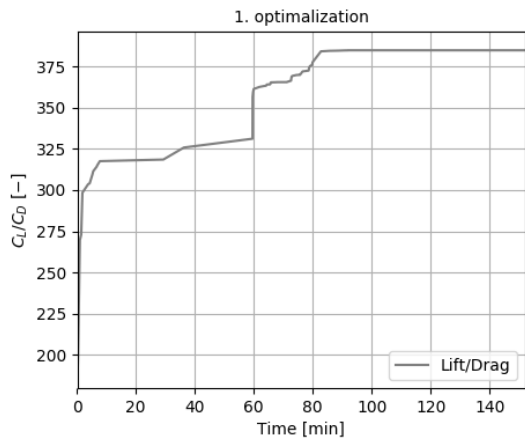
Obr. 108: PHO střídání po 30 min. 8/10.



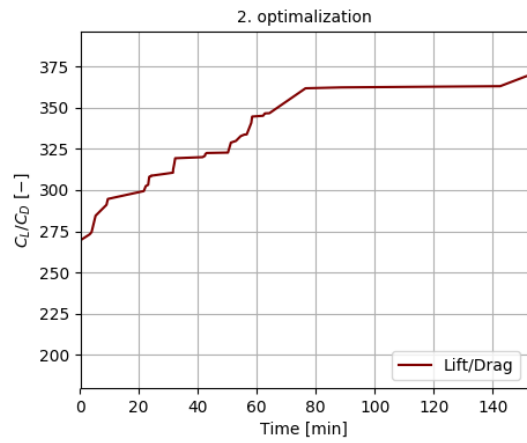
Obr. 109: PHO střídání po 30 min. 9/10.



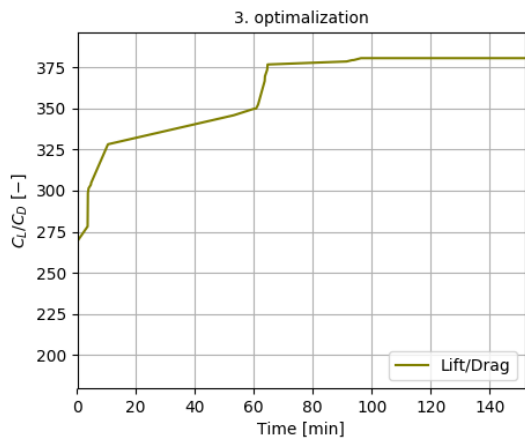
Obr. 110: PHO střídání po 30 min. 10/10.



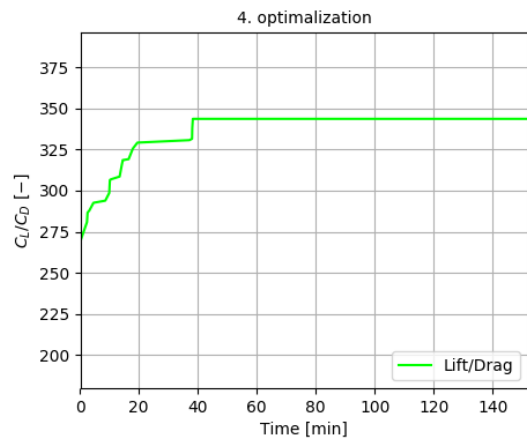
Obr. 111: PHO střídání po 45 min. 1/10.



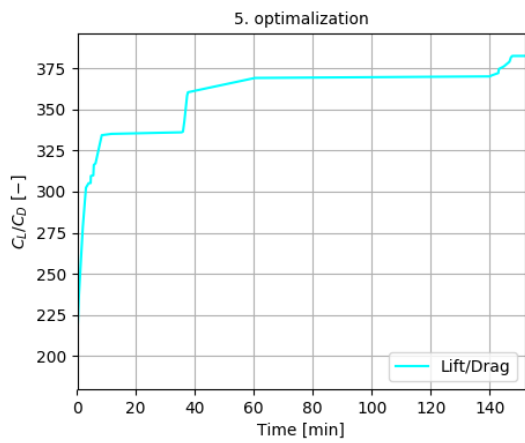
Obr. 112: PHO střídání po 45 min. 2/10.



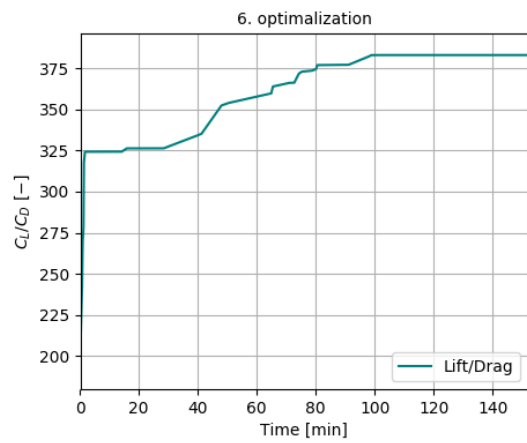
Obr. 113: PHO střídání po 45 min. 3/10.



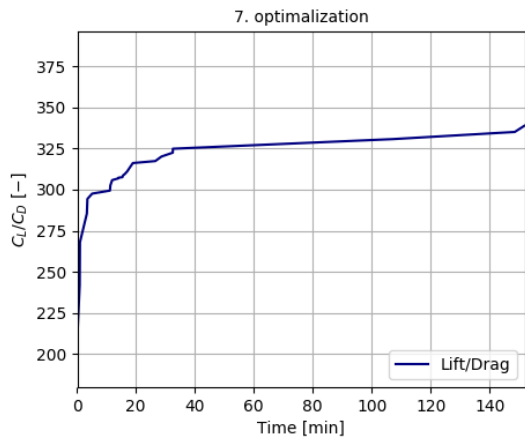
Obr. 114: PHO střídání po 45 min. 4/10.



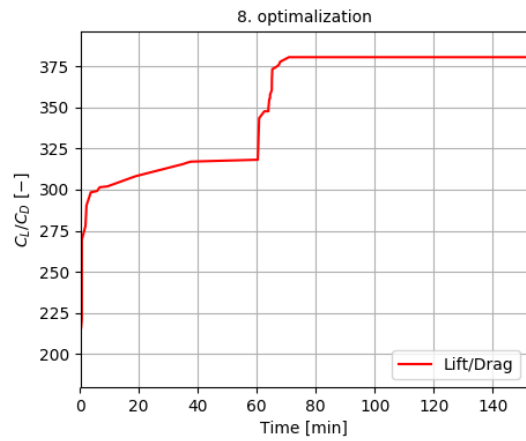
Obr. 115: PHO střídání po 45 min. 5/10.



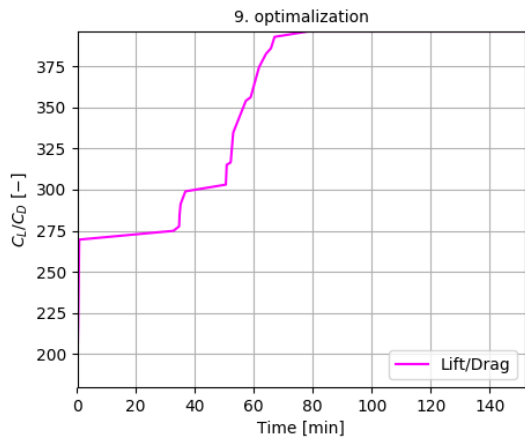
Obr. 116: PHO střídání po 45 min. 6/10.



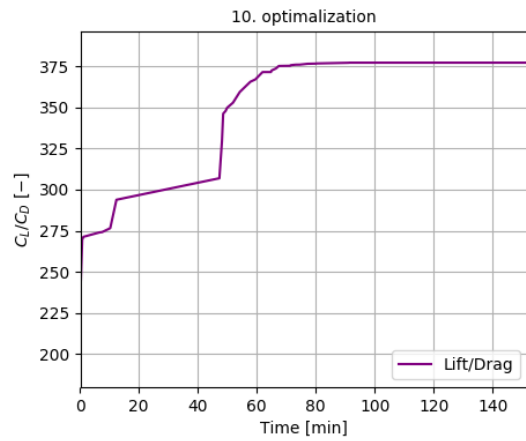
Obr. 117: PHO střídání po 45 min 7/10.



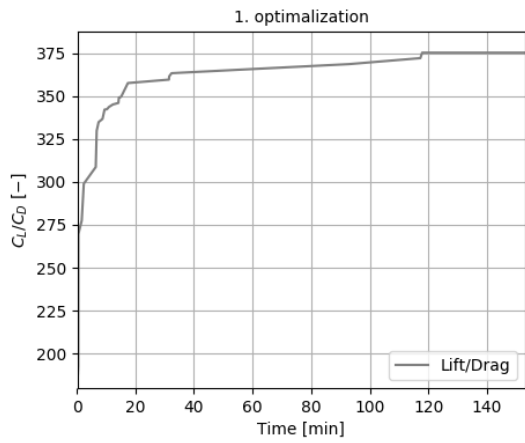
Obr. 118: PHO střídání po 45 min. 8/10.



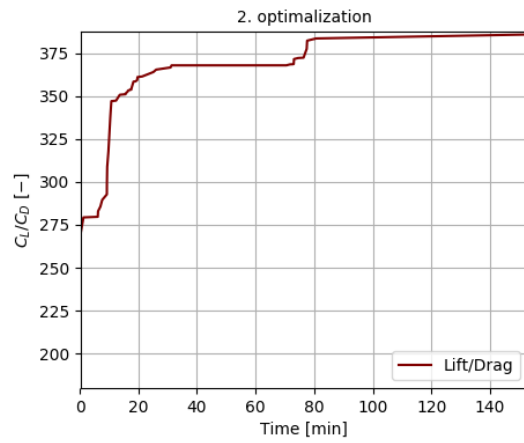
Obr. 119: PHO střídání po 45 min. 9/10.



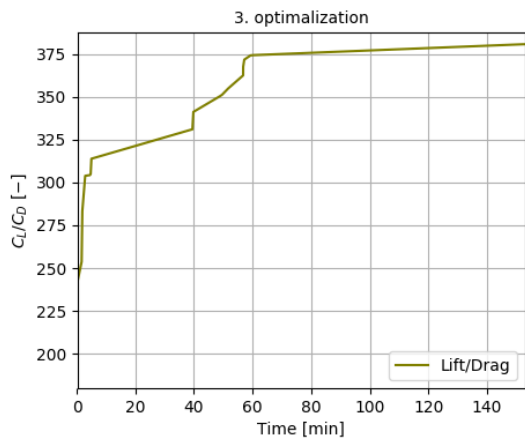
Obr. 120: PHO střídání po 45 min. 10/10.



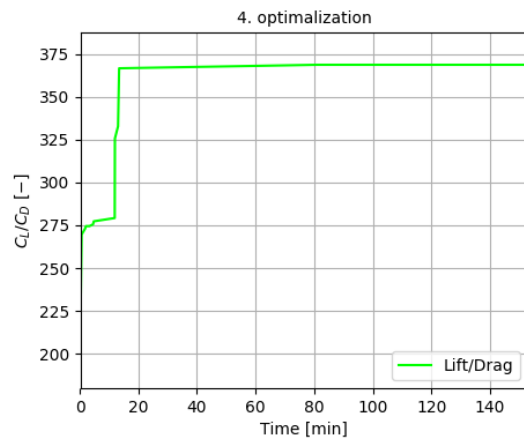
Obr. 121: PHO střídání po 5 a 45 min. 1/10.



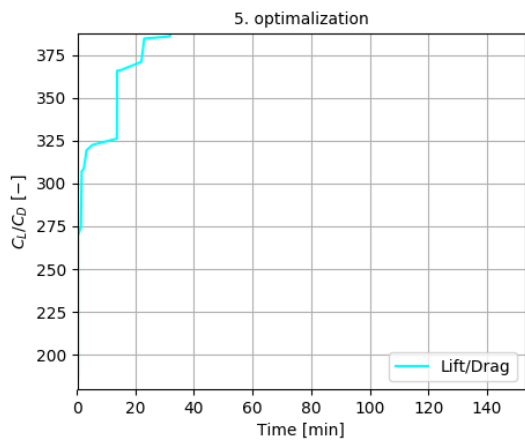
Obr. 122: PHO střídání po 5 a 45 min. 2/10.



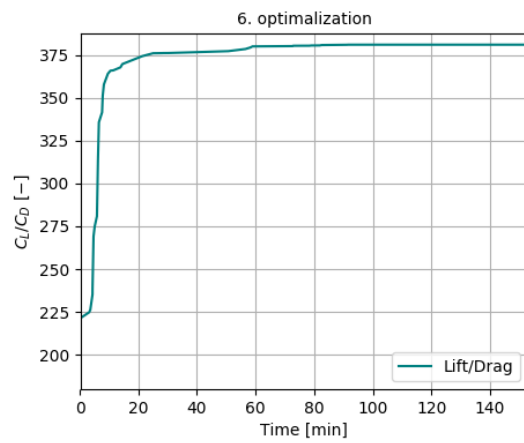
Obr. 123: PHO střídání po 5 a 45 min. 3/10.



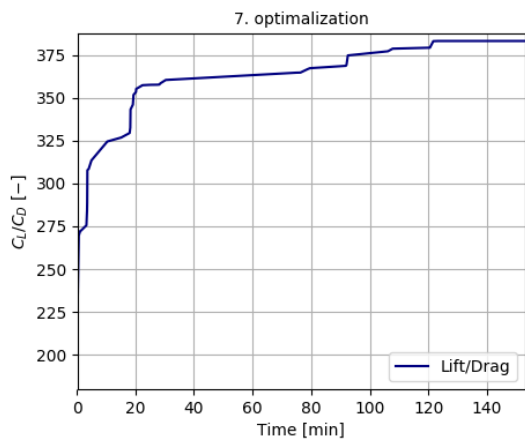
Obr. 124: PHO střídání po 5 a 45 min. 4/10.



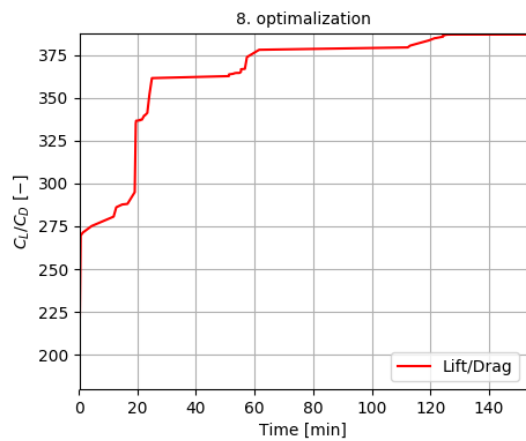
Obr. 125: PHO střídání po 5 a 45 min. 5/10.



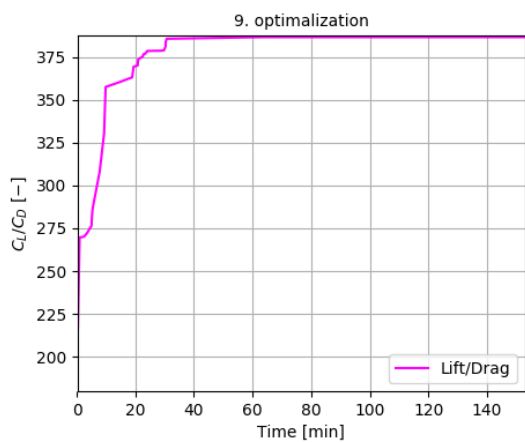
Obr. 126: PHO střídání po 5 a 45 min. 6/10.



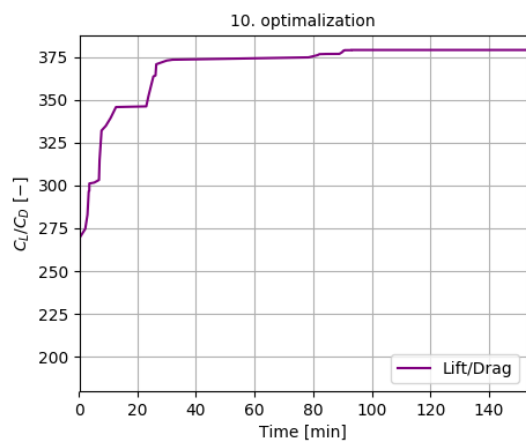
Obr. 127: PHO střídání po 5 a 45 min. 7/10.



Obr. 128: PHO střídání po 5 a 45 min. 8/10.

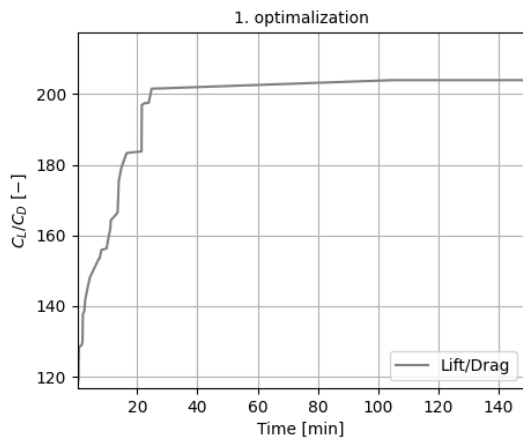


Obr. 129: PHO střídání po 5 a 45 min. 9/10.

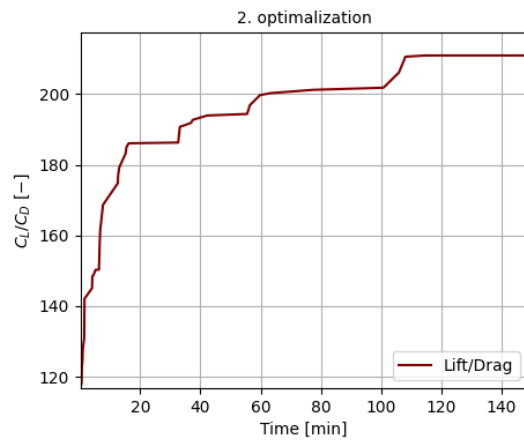


Obr. 130: PHO střídání po 5 a 45 min. 10/10

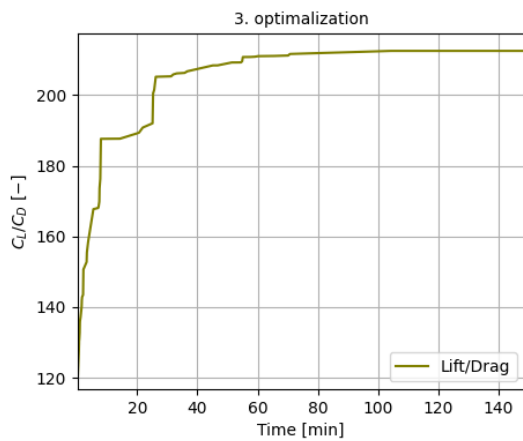
Optimalizace aPSO s výchozím profilem



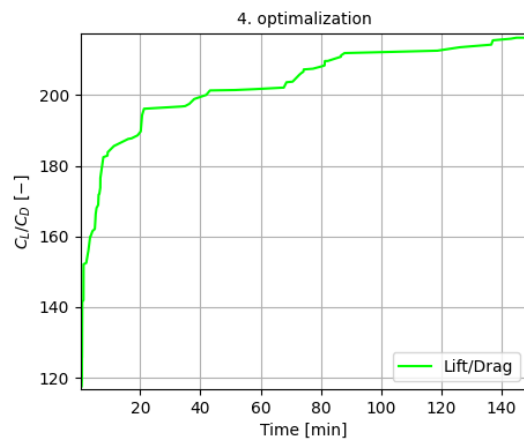
Obr. 131: aPSO s výchozím profilem 1/10.



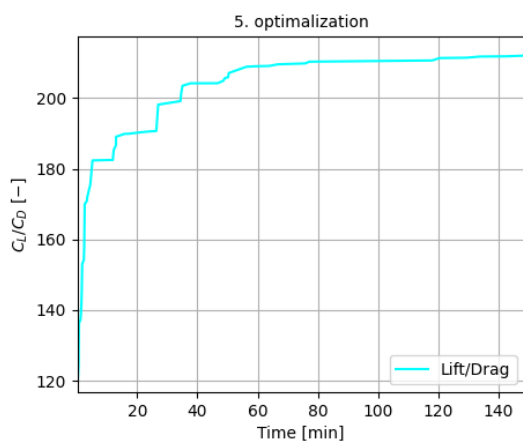
Obr. 132: aPSO s výchozím profilem 2/10.



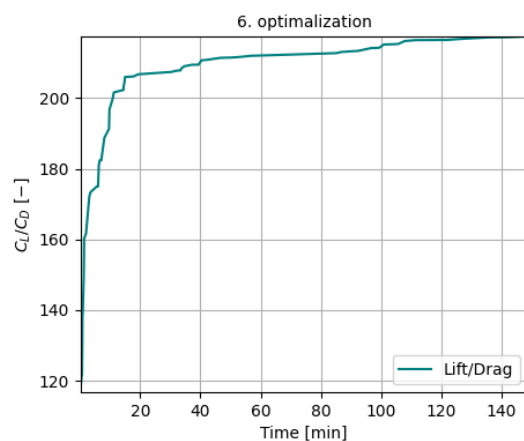
Obr. 133: aPSO s výchozím profilem 3/10.



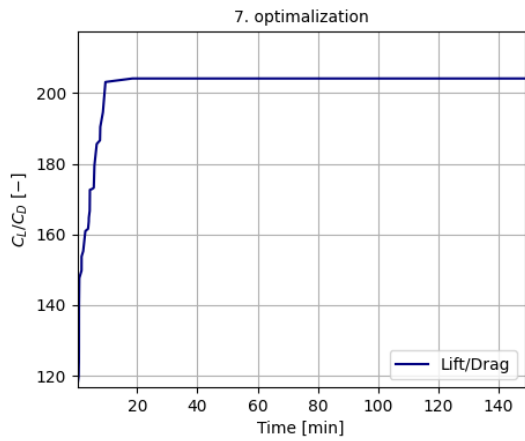
Obr. 134: aPSO s výchozím profilem 4/10.



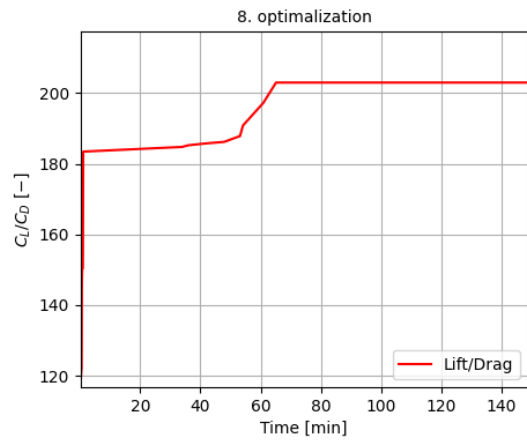
Obr. 135: aPSO s výchozím profilem 5/10.



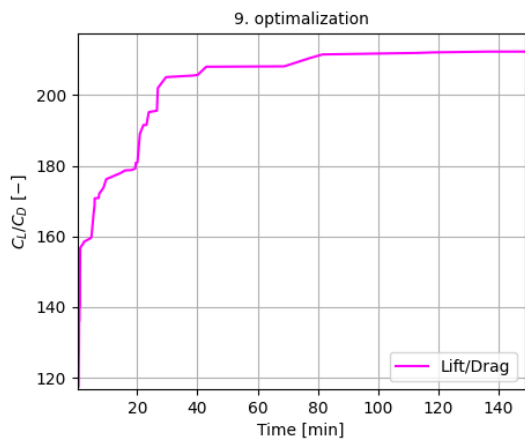
Obr. 136: aPSO s výchozím profilem 6/10.



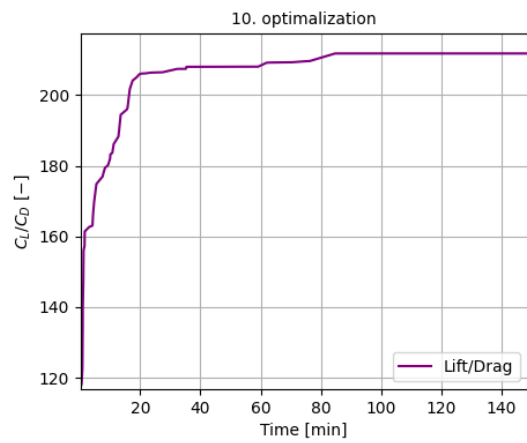
Obr. 137: aPSO s výchozím profilem 7/10.



Obr. 138: aPSO s výchozím profilem 8/10.

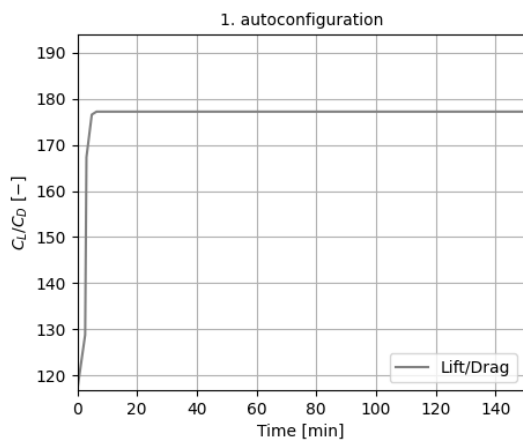


Obr. 139: aPSO s výchozím profilem 9/10.

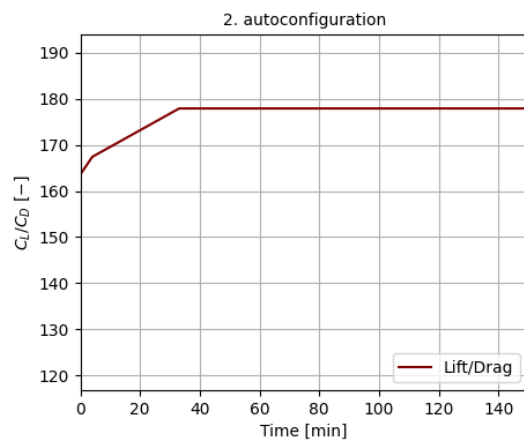


Obr. 140: aPSO s výchozím profilem 10/10

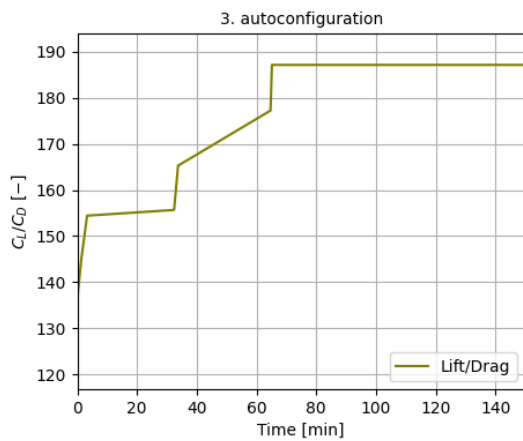
Automatická konfigurace IRACE s výchozím profilem



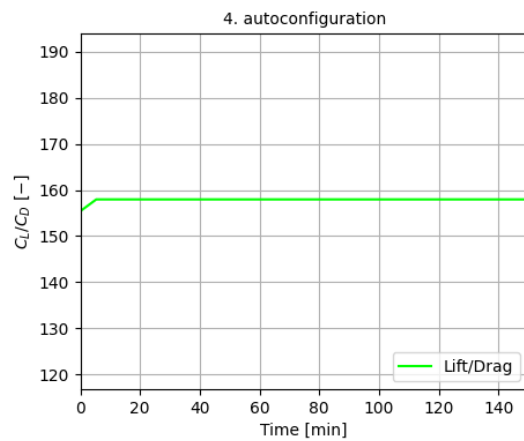
Obr. 141: IRACE s výchozím profilem 1/10.



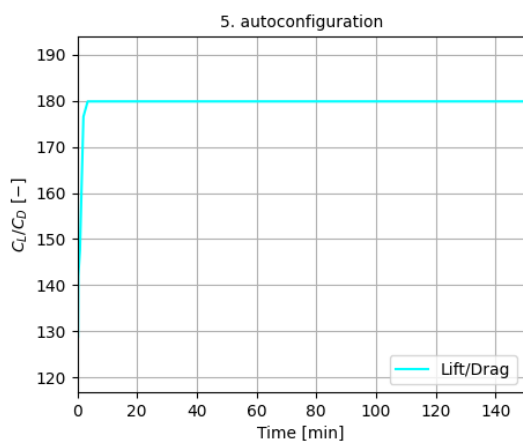
Obr. 142: IRACE s výchozím profilem 2/10.



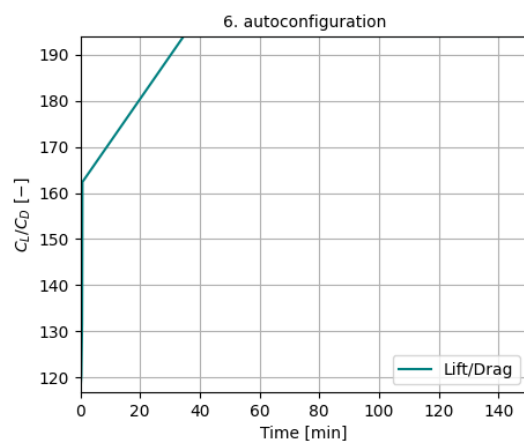
Obr. 143: IRACE s výchozím profilem 3/10.



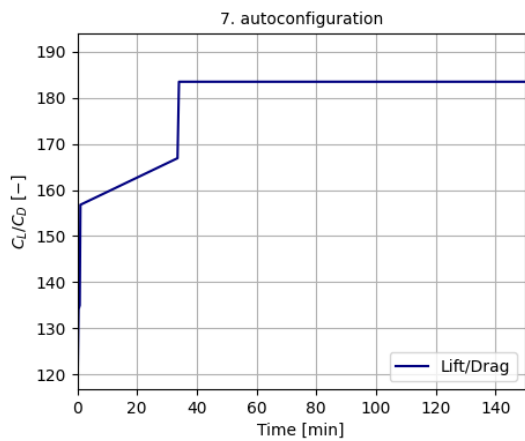
Obr. 144: IRACE s výchozím profilem 4/10.



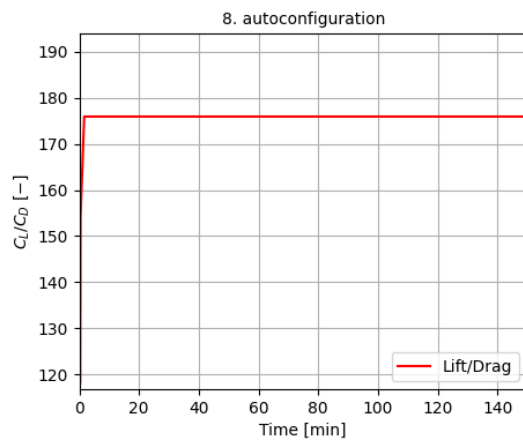
Obr. 145: IRACE s výchozím profilem 5/10.



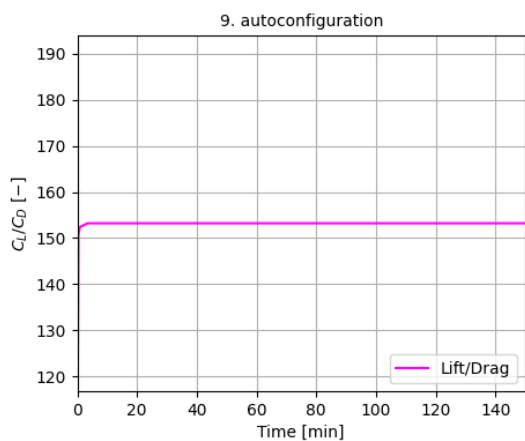
Obr. 146: IRACE s výchozím profilem 6/10.



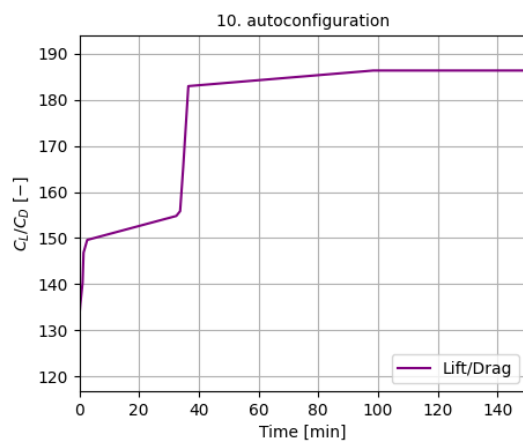
Obr. 147: IRACE s výchozím profilem 7/10.



Obr. 148: IRACE s výchozím profilem 8/10.



Obr. 149: IRACE s výchozím profilem 9/10.



Obr. 150: IRACE s výchozím profilem 10/10

Příloha E - Výsledky aerodynamických vlastností

XFOIL		Version 6.99				
Calculated polar for: EXTREME VALUE						
1 1 Reynolds number fixed			Mach number fixed			
xtrf =	1.000 (top)	1.000 (bottom)				
Mach =	0.000	Re =	5.000 e 6	Ncrit =	9.000	
alpha	CL	CD	CDp	CM	Top_Xtr	Bot_Xtr
-15.000	-0.6063	0.01961	0.01612	-0.1994	0.6572	0.0067
-14.000	-0.5185	0.01651	0.01280	-0.1999	0.6567	0.0069
-13.000	-0.4185	0.01444	0.01057	-0.2001	0.6561	0.0070
-12.000	-0.3166	0.01228	0.00819	-0.2003	0.6555	0.0079
-11.000	-0.2072	0.01098	0.00679	-0.2008	0.6548	0.0087
-10.000	-0.0944	0.01012	0.00585	-0.2015	0.6537	0.0092
-9.000	0.0177	0.00928	0.00493	-0.2021	0.6521	0.0108
-8.000	0.1328	0.00876	0.00437	-0.2031	0.6514	0.0118
-7.000	0.2490	0.00812	0.00371	-0.2044	0.6512	0.0145
-6.000	0.3660	0.00763	0.00322	-0.2058	0.6510	0.0187
-5.000	0.4829	0.00714	0.00281	-0.2073	0.6508	0.0340
-4.000	0.6011	0.00692	0.00264	-0.2091	0.6506	0.0428
-3.000	0.7194	0.00677	0.00253	-0.2111	0.6503	0.0491
-2.000	0.8375	0.00666	0.00249	-0.2131	0.6499	0.0577
-1.000	0.9552	0.00657	0.00249	-0.2152	0.6494	0.0762
-0.000	1.0729	0.00651	0.00252	-0.2174	0.6488	0.0951
1.000	1.1914	0.00629	0.00243	-0.2200	0.6477	0.1377
2.000	1.3117	0.00574	0.00209	-0.2231	0.6453	0.2437
3.000	1.4280	0.00533	0.00204	-0.2256	0.6429	0.4440
4.000	1.5424	0.00534	0.00224	-0.2279	0.6399	0.5425
5.000	1.6589	0.00517	0.00226	-0.2307	0.6375	0.6207
6.000	1.7948	0.00467	0.00242	-0.2384	0.6320	1.0000
7.000	1.9056	0.00481	0.00257	-0.2404	0.6206	1.0000
8.000	1.8654	0.01029	0.00749	-0.2151	0.4833	1.0000
9.000	1.5954	0.03530	0.03239	-0.1749	0.3997	1.0000
10.000	1.6225	0.04087	0.03775	-0.1679	0.3540	1.0000
11.000	1.6559	0.04604	0.04262	-0.1620	0.2988	1.0000
12.000	1.6904	0.05141	0.04779	-0.1566	0.2561	1.0000
13.000	1.7268	0.05662	0.05272	-0.1517	0.2053	1.0000
14.000	1.7469	0.06340	0.05917	-0.1456	0.1500	1.0000
15.000	1.7672	0.07041	0.06593	-0.1402	0.1050	1.0000

Tab. 1: Výsledné aerodynamické vlastnosti při optimalizaci jednoho extrému.

XFOIL

Version 6.99

Calculated polar for: AVERAGE VALUE

1 1 Reynolds number fixed

Mach number fixed

xtrf = 1.000 (top)

1.000 (bottom)

Mach = 0.000

Re =

5.000 e 6

Ncrit =

9.000

alpha	CL	CD	CDp	CM	Top_Xtr	Bot_Xtr
-15.000	-1.3130	0.01793	0.01362	-0.0519	0.6802	0.0077
-14.000	-1.2138	0.01603	0.01152	-0.0500	0.6436	0.0081
-13.000	-1.1094	0.01450	0.00982	-0.0484	0.6145	0.0084
-12.000	-1.0016	0.01321	0.00837	-0.0472	0.5879	0.0087
-11.000	-0.8912	0.01212	0.00715	-0.0463	0.5647	0.0088
-10.000	-0.7801	0.01085	0.00573	-0.0455	0.5453	0.0097
-9.000	-0.6662	0.00998	0.00478	-0.0450	0.5268	0.0106
-8.000	-0.5505	0.00932	0.00404	-0.0448	0.5092	0.0112
-7.000	-0.4344	0.00859	0.00323	-0.0447	0.4927	0.0127
-6.000	-0.3168	0.00811	0.00270	-0.0448	0.4762	0.0137
-5.000	-0.1989	0.00765	0.00219	-0.0451	0.4601	0.0163
-4.000	-0.0802	0.00729	0.00181	-0.0455	0.4441	0.0195
-3.000	0.0388	0.00700	0.00150	-0.0461	0.4281	0.0250
-2.000	0.1581	0.00677	0.00129	-0.0469	0.4144	0.0345
-1.000	0.2774	0.00657	0.00115	-0.0477	0.4010	0.0566
-0.000	0.3967	0.00637	0.00108	-0.0487	0.3875	0.1112
1.000	0.5155	0.00603	0.00110	-0.0499	0.3734	0.2695
2.000	0.6332	0.00544	0.00122	-0.0512	0.3593	0.5909
3.000	0.7503	0.00515	0.00143	-0.0523	0.3448	0.8223
4.000	0.8685	0.00508	0.00166	-0.0533	0.3300	1.0000
5.000	0.9855	0.00543	0.00192	-0.0544	0.3147	1.0000
6.000	1.1020	0.00584	0.00224	-0.0557	0.2986	1.0000
7.000	1.2175	0.00632	0.00264	-0.0570	0.2808	1.0000
8.000	1.3312	0.00693	0.00314	-0.0582	0.2596	1.0000
9.000	1.4435	0.00759	0.00373	-0.0593	0.2388	1.0000
10.000	1.5522	0.00851	0.00452	-0.0602	0.2120	1.0000
11.000	1.6559	0.00974	0.00559	-0.0605	0.1804	1.0000
12.000	1.7519	0.01139	0.00704	-0.0601	0.1449	1.0000
13.000	1.8358	0.01358	0.00903	-0.0583	0.1108	1.0000
14.000	1.8840	0.01704	0.01236	-0.0527	0.0808	1.0000
15.000	1.8996	0.02531	0.02066	-0.0517	0.0605	1.0000

Tab. 2: Výsledné aerodynamické vlastnosti při optimalizaci širšího ustáleného letu.

XFOIL

Version 6.99

Calculated polar for: NACA2412

1 1 Reynolds number fixed

Mach number fixed

xtrf = 1.000 (top)

1.000 (bottom)

Mach = 0.000

Re =

5.000 e 6

Ncrit = 9.000

alpha	CL	CD	CDp	CM	Top_Xtr	Bot_Xtr
-15.000	-1.3936	0.01619	0.01215	-0.0558	0.9988	0.0074
-14.000	-1.2867	0.01407	0.00994	-0.0573	0.9959	0.0089
-13.000	-1.1752	0.01235	0.00816	-0.0590	0.9930	0.0108
-12.000	-1.0671	0.01080	0.00658	-0.0595	0.9872	0.0140
-11.000	-0.9451	0.00933	0.00514	-0.0629	0.9835	0.0210
-10.000	-0.8250	0.00836	0.00421	-0.0652	0.9715	0.0277
-9.000	-0.7134	0.00775	0.00359	-0.0653	0.9423	0.0320
-7.000	-0.5049	0.00685	0.00249	-0.0618	0.8722	0.0368
-6.000	-0.4000	0.00657	0.00208	-0.0600	0.8253	0.0387
-5.000	-0.2942	0.00636	0.00173	-0.0584	0.7722	0.0394
-4.000	-0.1885	0.00615	0.00134	-0.0568	0.7124	0.0429
-3.000	-0.0818	0.00604	0.00109	-0.0553	0.6498	0.0466
-2.000	0.0233	0.00581	0.00086	-0.0536	0.5868	0.0912
-1.000	0.1289	0.00559	0.00073	-0.0521	0.5303	0.1658
-0.000	0.2346	0.00544	0.00069	-0.0506	0.4792	0.2579
1.000	0.3394	0.00534	0.00074	-0.0490	0.4257	0.3737
2.000	0.4404	0.00515	0.00087	-0.0466	0.3700	0.5653
3.000	0.5375	0.00521	0.00113	-0.0435	0.2985	0.7481
4.000	0.6300	0.00544	0.00149	-0.0391	0.2290	0.8987
5.000	0.7526	0.00621	0.00208	-0.0418	0.1494	0.9785
6.000	0.8798	0.00743	0.00289	-0.0459	0.0700	0.9927
7.000	1.0038	0.00829	0.00357	-0.0491	0.0464	0.9974
8.000	1.1275	0.00903	0.00424	-0.0523	0.0391	0.9998
9.000	1.2136	0.00989	0.00502	-0.0474	0.0287	1.0000
10.000	1.2902	0.01100	0.00605	-0.0408	0.0174	1.0000
11.000	1.3687	0.01216	0.00719	-0.0349	0.0128	1.0000
12.000	1.4407	0.01341	0.00846	-0.0281	0.0104	1.0000
13.000	1.5045	0.01488	0.00997	-0.0204	0.0085	1.0000
14.000	1.5634	0.01687	0.01203	-0.0131	0.0070	1.0000
15.000	1.6206	0.01917	0.01442	-0.0068	0.0065	1.0000

Tab. 3: Aerodynamické vlastnosti profilu NACA 2412.