



Dokumentace k projektu pro předměty EVO

Automatické ladění řídicího systému pro model letounu B737

12. dubna 2014

Autor: Lenka Turoňová, xturon02@fit.vutbr.cz

Obsah

1	Úvod	3
2	Analýza problému	4
2.1	Zadání projektu	4
2.2	Analýza problému	4
2.3	Řídící kontroléry a požadavky na systém	4
3	Návrh a implementace programu	6
3.1	Chromozóm	6
3.2	Fitness funkce	6
3.3	Evoluční strategie	6
4	Testování a vyhodnocení	8
4.1	Grafy	8
5	Závěr	12
6	Literatura	13

Kapitola 1

1 Úvod

Cílem projektu bylo navrhnout evoluční algoritmus, který poslouží pro optimalizaci nastavení kontrolérů velkého cestovního letadla letounu B737. Nastavení řídicích kontrolérů letounu je velmi složitý problém, jelikož jsou jednotlivé kontroléry navzájem provázány a navzájem se ovlivňují, což má za následek, že systém je velmi citlivý na jejich nastavení a malé změny mohou způsobit velké rozdíly ve výstupním signálu.

Pro potřebu návrhu algoritmu pro optimalizaci nastavení parametrů řídicích kontrolérů byl navržen nelineární model letounu B737, který slouží pro simulaci, na jejímž základě se vyhodnotí stabilita systému v závislosti na vstupním řídicím signálu.

Tento dokument popisuje návrh a implementaci evoluční strategie, která byla použita jako prostředek pro optimalizaci těchto parametrů, aby systém co nejvíce odpovídal očekávanému chování.

Kapitola 2

2 Analýza problému

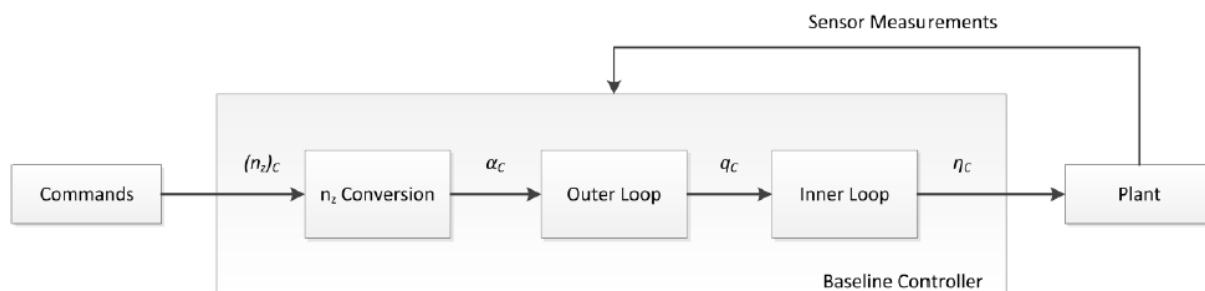
2.1 Zadání projektu

Cílem projektu bylo navržení evolučního algoritmu, který by měl optimalizovat sadu parametrů řídicího systému letounu B737 (genotyp). Optimalizace by měla probíhat na základě charakteristik přechodové funkce provozního násobku letounu (fitness funkce) s ohledem na kvalitu letu a citlivost řízení letounu.

Pro zjištění potřebných charakteristik (překmit, statická chyba, čas ustálení, bezpečnost v zesílení, bezpečnost ve fázi, Gibson Dropback Criterion (GDC) a Control Anticipation Parameter (CAP)) se očekává využití simulace nelineárního modelu letounu, který v sobě zahrnuje řídicí systém (sada kontrolérů). S výslednou sadou nalezených optimalizovaných parametrů (proporcionální a integrační zisky jednotlivých subsystémů nelineárního kontroléru založeného na nelineární dynamické inverzi) měla být provedena simulace a výsledky znázorněny graficky.

2.2 Analýza problému

Aby mohla být provedena simulace toho, jak přesně kopíruje letoun přechodovou funkci provozního násobku letounu ($n_{z \text{ MED}}$), bylo nutné vytvořit nelineární model, který v sobě zahrnuje sadu kontrolérů. Architektura nelineárního systému implementována pro potřeby simulace je znázorněna na níže uvedeném obrázku. Jak je vidět, kaskáda kontrolérů vyžaduje vstupní řídicí signál ($n_{z \text{ CMD}}$).



Obr. 2.1. : Architektura řídicích kontrolérů

V závislosti na nastavení kontrolérů pak systém vykazuje jisté chování, které je reprezentováno výstupním signálem ($n_{z \text{ MED}}$), který bude předmětem zkoumání při vyhodnocování fitness funkce u evoluční strategie. Cílem je nastavit parametry kontrolérů na takovou hodnotu, aby funkce n_z co možná nejpresněji kopírovala vstupní funkci $n_{z \text{ CMD}}$.

2.3 Řídicí kontroléry a požadavky na systém

Předmětem evoluční strategie je optimalizace řídicích kontrolérů. Nelineární model letounu B737 sestává se sady kontrolérů, jejichž seznam je uveden v tabulce níže.

Značka	Název
n_{zp}	proporcionální složka regulátoru provozního násobku
n_{zl}	integrační složka regulátoru provozního násobku
T_α	časová konstanta regulátoru úhlu náběhu
α_p	proporcionální složka regulátoru úhlu náběhu
α_L	integrační složka regulátoru úhlu náběhu
T_q	časová konstanta regulátoru úhlové rychlosti klopení
q_p	proporcionální složka regulátoru úhlové rychlosti klopení

Tab. 2.1.: Parametry řídicích kontrolérů

Výše uvedené parametry je tedy potřeba nastavit tak, aby odezva systému na signál $n_{z\text{ CMD}}$ co nejlépe odpovídala očekávanému průběhu. To ověříme pomocí několika charakteristik. Srovnání vstupního a výstupního signálu je založeno na splnění (minimalizaci) těchto vlastností.

Značka	Název
OVS	překmit
SE	statická chyba
F_{Dropback}	penalizační koeficient Gibson Dropback kritéria
F_{PM}	penalizační koeficient bezpečnosti ve fázi
F_{GM}	penalizační koeficient bezpečnosti v zesílení
F_{ST}	penalizační koeficient pro čas ustálení odezvy
F_{RT}	penalizační koeficient pro dobu náběhu
GDC	Gibson Dropback Criterion
CAP	Control Anticipation Parameter

Tab. 2.2.: Zkoumané charakteristiky nelineárního modelu letounu

Snahou je, aby překmit a statická chyba byla co nejmenší. U ostatních parametrů je udržet je v jistém předem stanoveném rozmezí, proto u fitness funkce tyto hodnoty postihneme jistou vysokou penalizací v případě, že překročí stanovenou hranici, jelikož by systém reagoval nepřiměřeně a vypočítané nastavení by nebylo možné reálně použít.

Kapitola 3

3 Návrh a implementace programu

Nyní se podíváme na to, jak tedy bude evoluční strategie implementována. V úvodu je potřeba uvést, že v závislosti na skutečnosti, že parametry řídicích kontrolérů mají reálné hodnoty, byla pro optimalizaci použita metoda evoluční strategie, která jako vstup využívá chromozóm sestávající se z reálných čísel.

3.1 Chromozóm

V úvodní fázi je potřeba stanovit podobu chromozómu, který je základ pro evoluční strategii. Jak již bylo uvedeno v kapitole 2.3, předmětem problému je nastavení sedmi parametrů uvedených v Tab. 2.1. Z tohoto důvodu bude jedinec v populaci reprezentován pomocí níže znázorněného chromozómu.

n_{zp}	n_{zl}	T_α	α_p	α_L	T_q	q_p
----------	----------	------------	------------	------------	-------	-------

Obr. 3.1.: Chromozóm reprezentující jedince

3.2 Fitness funkce

Po vygenerování počáteční populace je nutné ohodnotit každého jedince pomocí fitness funkce, jejíž hodnotu se snažíme minimalizovat. V kapitole 2.3 jsme si uvedli charakteristiky, které budeme zkoumat. Můžeme je najít v tabulce Tab. 2.2.

Pro vyhodnocení fitness funkce je nutné nejdříve nastavit příslušné parametry, které jsou součástí chromozómu každého jedince a spustit simulaci nelineárního systému letounu, který je implementován.

Výstupem této simulace je signál n_z , který je srovnán se vstupním signálem $n_{z\,CMD}$. V případě překmitu a statické chyby se snažíme o jejich minimalizaci. Pro výslednou fitness funkci budou mít zhruba srovnatelnou váhu, proto je vynásobíme vahou rovnou 0.5. U dalších parametrů je snahou je udržet v jisté mezi, proto jejich překročení budou nastaveny na hodnotu 1 či 0, a následně penalizovány vysokou hodnotou, jelikož v případě překročení stanovené meze, by letadlo nepříjemně reagovalo.

Fitness funkce proto bude mít následující tvar:

$$FF = 0.5 \cdot OVS + 0.5 \cdot SE + (F_{Dropback} + F_{CAP} + F_{RM} + F_{GM} + F_{ST} + F_{RT}) \cdot pen$$

Penalizace v tomto případě byla stanovena na hodnotu rovnou 5000.

3.3 Evoluční strategie

Evoluční strategie má několik variant. Pro tento problém byla zvolena metoda 1:5. Pro srovnání byly implementovány obě varianty jak *čárková*, tak i *plusová*. Evoluční strategie je založena na níže uvedeném algoritmu:

```

x:= náhodně vygenerovaný vektor reálných čísel
t:=0;  $\sigma$ :=  $\sigma_{ini}$ ; x*=x;
while t < tmax do
  begin i:=0;k:=0;
  while i<imax do
    begin i:=i+1; x':=x+N(0,  $\sigma$ );
    if f(x') < f(x) then
      begin k:= k+1; x:= x';
      if f(x) < f(x*) then
        x*:=x;
      end;
    end;
  fi k/max < 0.2 then  $\sigma$ := cd*  $\sigma$  else
  fi k/max < 0.2 then  $\sigma$ := ci*  $\sigma$ ;

```

Výše uvedený algoritmus byl implementován následujícím způsobem. V prvotní fázi je vygenerován náhodně jeden jedinec se sedmi parametry požadovaného rozsahu, který je od parametrů očekáván. Následně se provede vygenerování (50) potomků pomocí mutace rodiče. Vygeneruje se náhodné číslo, které se přičte k jednotlivému parametru. Je potřeba přepočítat toto číslo, pokud se nachází se v požadovaném rozmezí.

Nová generace je následně ohodnocena fitness funkcí. Je spuštěna simulace s parametry jednotlivého jedince a na základě výstupu této simulace je vypočtena fitness funkce, která je uložena k jednotlivému jedinci.

V dalším kroku je spočten počet jedinců, jejichž fitness funkce je lepší než fitness funkce rodiče. Na základě této hodnoty je stanovena hodnota odchylky náhodného čísla pro mutaci rodiče v následující generaci.

V případě varianty *čárkové* se rodičovská fitness funkce nastaví na nekonečno, aby nebyla vybrána. V případě varianty *plusové* je ponechána, jelikož může být také vybrána. Z populace je tedy vybrán nejlepší jedinec, který je stanoven jako rodič a pokračuje se opět v generování populace nové.

Cyklus probíhá do té doby, než je překročen stanovený počet generací, který se má vytvořit. S parametry nejlepšího jedince je provedena opětovná simulace a výsledky jsou znázorněny v grafech.

Kapitola 4

4 Testování a vyhodnocení

Předmětem testování bylo především srovnání *čárkové* a *plusové* metody evoluční strategie. Pro testování bylo provedeno několik spuštění výpočtu s 50 vygenerovanými jedinci při počtu 100 generací. Následující grafy a tabulky reprezentují srovnání obou metod, průběh fitness funkce v průběhu vybraného výpočtu.

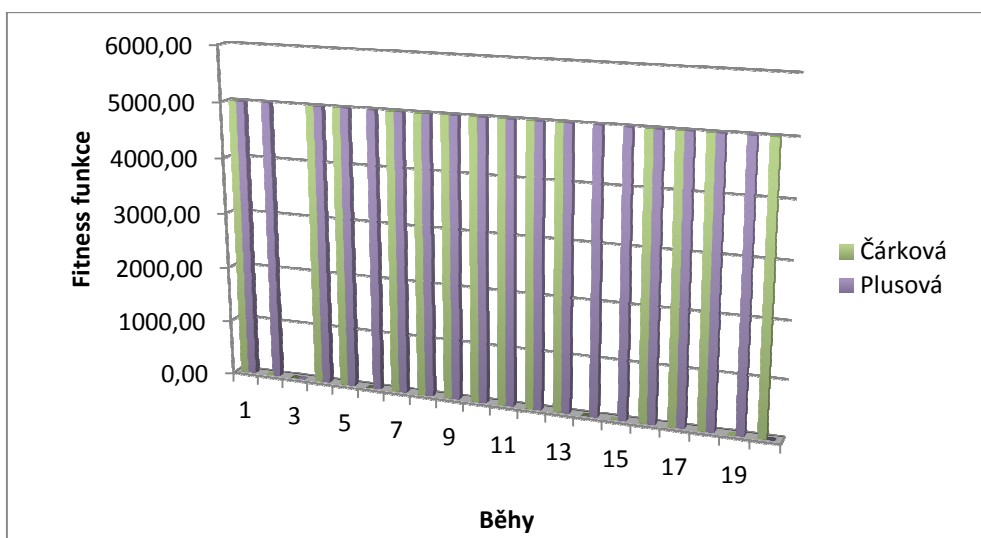
Je nutné uvést, že evoluční strategie je stochastická metoda, která vyžaduje několikanásobné provádění běhu a ne vždy se podaří nalézt řešení, které by mohlo být v praxi použito. Také parametry kontrolérů jsou velmi citlivé na změnu, takže se může velmi snadno stát, že signál překročí povolené charakteristiky a hodnota fitness funkce se citelně zvýší díky penalizaci, kterou bude hodnocení jedince postihnuto.

4.1 Grafy

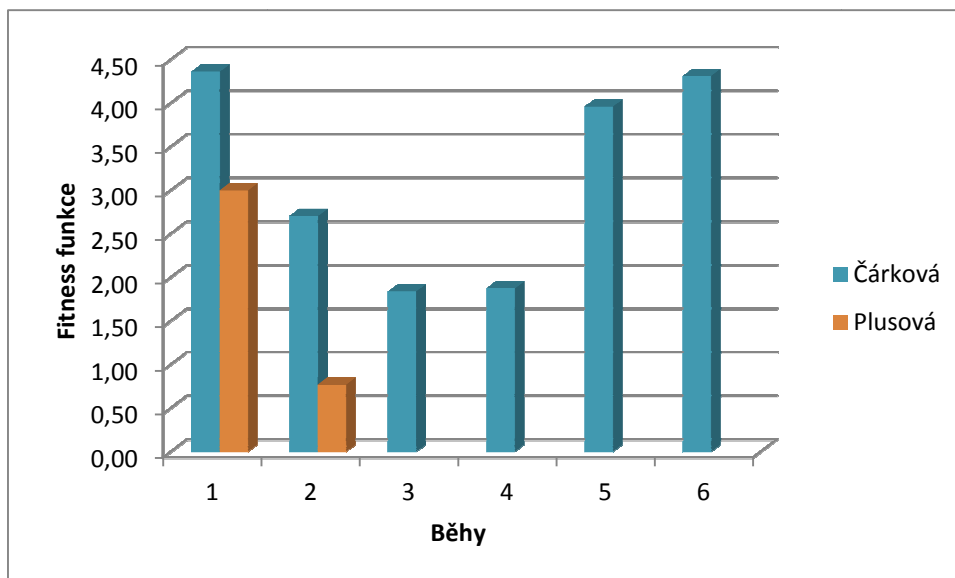
První graf znázorňuje srovnání obou variant evoluční strategie – *čárkové* a *plusové*. Byla provedena série 20 běhů pro každou variantu a výsledkem každého běhu byl jedinec s nejlepší fitness funkcí vybrán podle dané varianty algoritmu. Jejich hodnoty byly následně zaneseny do grafu.

Varianta/Běhy	1	2	3	4	5	6	7	8	9	10
Čárková	5000,7	4,3631	2,7069	5003	5000,7	1,84	5002,3	5001	5002,5	5002
Plusová	5001,3	5005,7	3	5000,8	5003,8	5001,9	5001,8	5000,8	5003,1	5003,1
Varianta/Běhy	11	12	13	14	15	16	17	18	19	20
Čárková	5002,6	5000,5	5003,4	1,875	3,9616	5001,3	5001,8	5002,7	4,312	5003,5
Plusová	5002,7	5002,6	5001,8	5002,8	5002,1	5002,3	5002,1	5002,7	5004	0,7714

Tabulka 1.: Fitness funkce v jednotlivých bězích čárkové a plusové varianty evoluční strategie



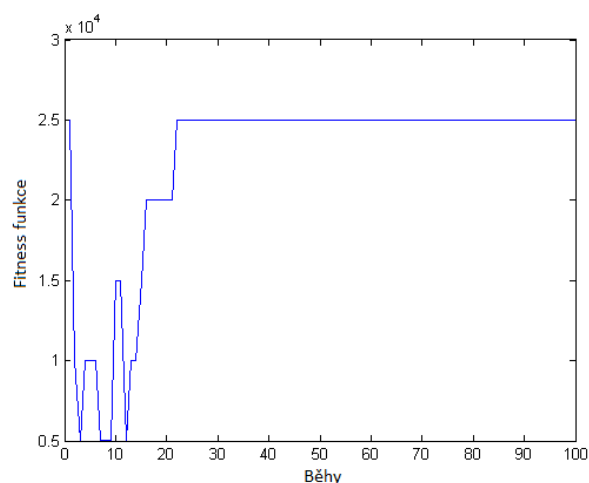
Graf. 4.1.: Fitness funkce v jednotlivých bězích (porovnání dvou variant evoluční strategie)



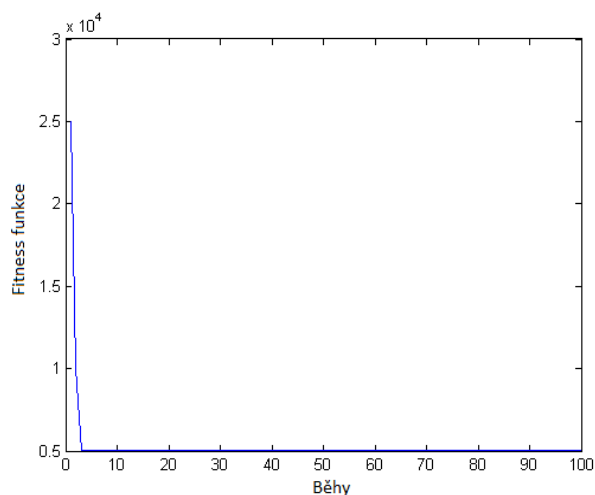
Graf 4.1. : Vybrané úspěšné běhy obou variant

Graf ukazuje, že čárková metoda bývá ve většině případů úspěšnější než ta plusová. Čárková dospěla k přijatelným výsledkům v šesti případech z dvaceti, kdežto plusová varianta pouze v dvou bězích z dvaceti.

Další graf reprezentuje, jak se fitness funkce u dvou vybraných běhů dané varianty vyvíjela s časem. Algoritmus u obou variant běžel po dobu 100 generací. U plusové varianty by se měla fitness funkce stále snižovat (nebo stagnovat), jelikož jako rodič další generace je vybrán jedinec z potomků, pokud má stejnou nebo nižší fitness funkci než rodič. U varianty čárkové může fitness funkce velmi kolísat, jelikož aktuální rodič je odebrán z procesu selekce, i kdyžby měl lepší fitness funkci.



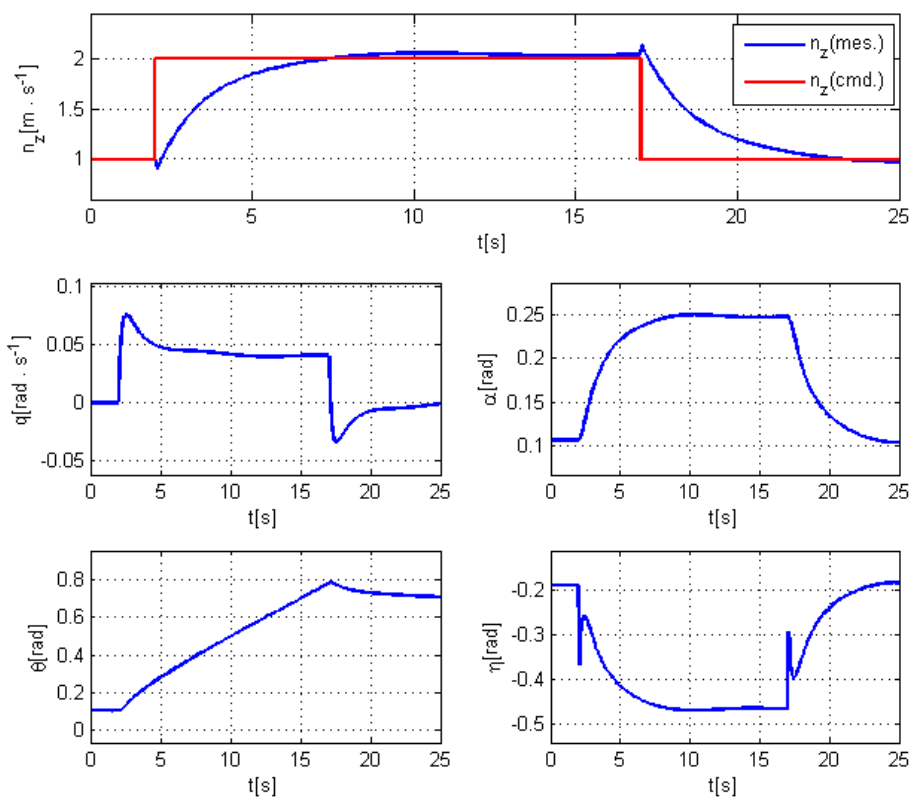
a



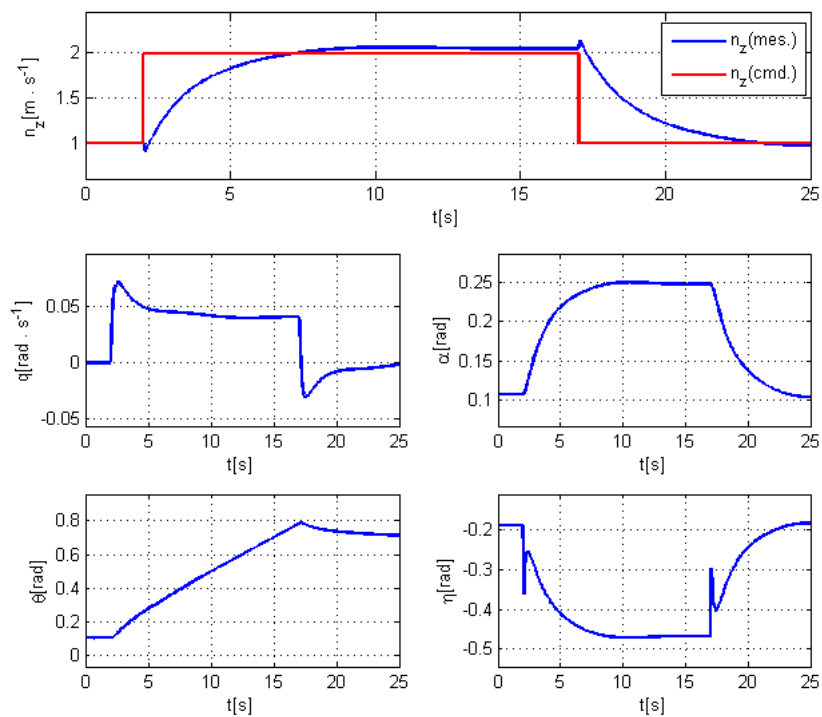
b

Graf. 4.2.: Průběh fitness funkce: (a) čárková metoda; (b) plusová metoda

Poslední dva grafy ukazují výsledky simulace s nejlepším jedincem dané varianty. Na prvním grafu je porovnání vstupního řídicího signálu $n_{z\text{CMD}}$ a výstupního signálu modelu $n_{z\text{MES}}$.

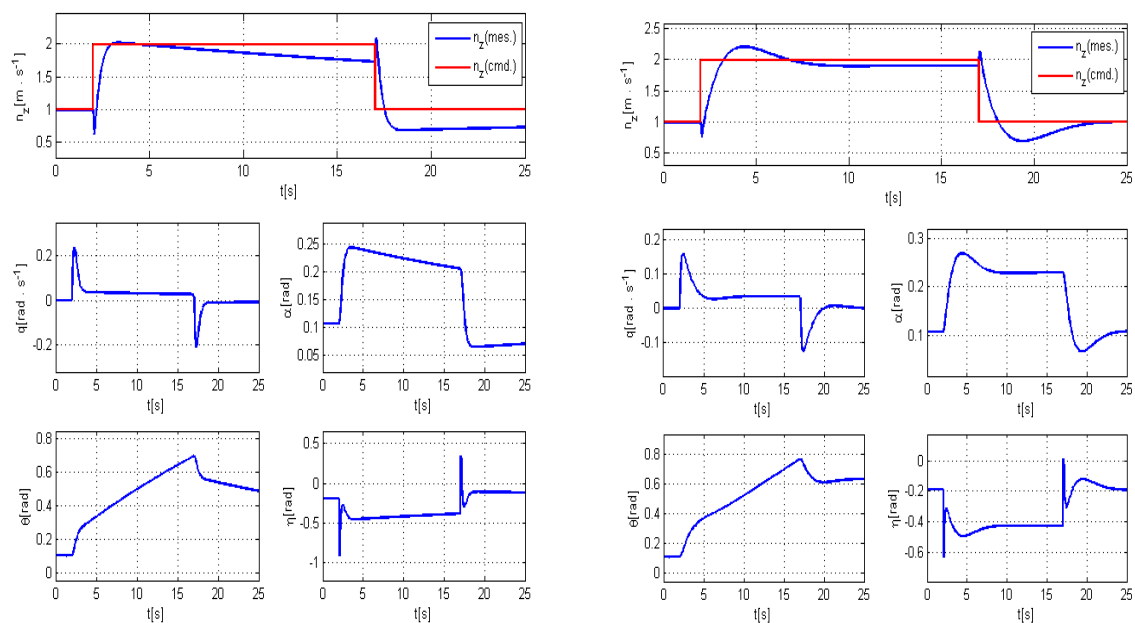


Graf. 4.3. Čárková varianta



Graf. 4.4.: Plusová varianta

V obou případech nekopíruje výstupní signál vstupní zcela přesně, je však vidět, že oproti ukázce původních jedinců, které je možné vidět níže, bylo nastavení poměrně úspěšně optimalizováno.



Graf. 4.5.: Původní jedinci

Kapitola 5

5 Závěr

Program byl naimplementován v jazyce *Matlab*. Díky tomuto programu bylo možné optimalizovat nelineární řídicí systém letounu B737 sestávající se ze sady kontrolérů, jejichž správné nastavení bylo cílem evoluční strategie, která byla použita pro optimalizaci.

Cílem projektu bylo nastavení těchto parametrů, takovým způsobem, aby výstupní signál $n_{z\text{ MES}}$ co nej přesněji kopíroval vstupní přechodovou funkci $n_{z\text{ CMD}}$. Toho bylo dosaženo použitím evoluční strategie, a to dvou variant *čárkové*, v níž se stane v nové generaci rodičem nejlepší jedinec z vygenerovaných potomků s pravidlem 1:5, a *plusové*, u které se v nové generaci rodičem ten jedinec, který je z množiny všech vygenerovaných potomků a rodičů ten nejlepší, dle vypočtené fitness funkce.

Testování proběhlo ve dvaceti bžích s nastavení počtu generací na 100 a s počtem 50 jedinců. Výsledky ukázaly, že čárková varianta evoluční strategie sice dojde ve větším počtu bžů k přijatelnému řešení, které nepřekračuje stanované meze, avšak plusová varianta by nejspíše poskytovala optimálnější výsledky. Jeden bž však trvá velmi dlouho, a tak nebylo možné získat zatím větší množství výsledků, které by to potvrdily. Vzhledem k tomu, jak je nastavení parametrů řídicích kontrolérů citlivé, poskytuje evoluční strategie velmi užitečný způsob optimálního nastavení.

6 Literatura

[1] <http://www.fit.vutbr.cz/research/pubs/index.php.cs?id=10440>