



TECHNICKÁ UNIVERZITA V LIBERCI
Fakulta mechatroniky, informatiky
a mezioborových studií ■

VÝVOJ BEZRÁZOVÝCH POLOHOVÝCH PROFILŮ PRO ASTRONOMICKÝ TELESKOP

Diplomová práce

Studijní program: N2612 – Elektrotechnika a informatika
Studijní obor: 3902T005 – Automatické řízení a inženýrská informatika
Autor práce: **Bc. Jakub Nosek**
Vedoucí práce: Ing. David Lindr, Ph.D.





TECHNICAL UNIVERSITY OF LIBEREC
Faculty of Mechatronics, Informatics
and Interdisciplinary Studies ■

DERIVATION OF THE JERK-FREE POSITION PROFILES FOR AN ASTRONOMICAL TELESCOPE

Diploma thesis

Study programme: N2612 – Electrical Engineering and Informatics
Study branch: 3902T005 – Automatic Control and Applied Computer Science
Author: **Bc. Jakub Nosek**
Supervisor: Ing. David Lindr, Ph.D.



ZADÁNÍ DIPLOMOVÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Bc. Jakub Nosek**
Osobní číslo: **M12000246**
Studijní program: **N2612 Elektrotechnika a informatika**
Studijní obor: **Automatické řízení a inženýrská informatika**
Název tématu: **Vývoj bezrázových polohových profilů pro astronomický teleskop**
Zadávací katedra: **Ústav mechatroniky a technické informatiky**

Z á s a d y p r o v y p r a c o v á n í :

1. Prostudujte zadávací dokumentaci firmy Project Soft a proveďte rešerši tematiky polohových profilů s omezením ryvu tzv. S-curve.
2. Seznamte se s metodikou odvozování polohových profilů s nulovými i nenulovými počátečními podmínkami.
3. Nalezněte analytické řešení a vytvořte funkci v prostředí Matlab pro generování polohových profilů. Otestujte správnost odvozených vztahů pro všechny možné stavy a počáteční podmínky.
4. Proveďte implementaci kódu pro generování polohových profilů do PLC a ověřte jeho funkci na reálném servopohonu.
5. Vypracujte závěrečnou textovou dokumentaci, zhodnoťte výsledky vaší práce a do elektronických příloh uveďte, mimo jiné, i zdrojové kódy vytvořených programů.


Rozsah grafických prací: **dle potřeby dokumentace**
Rozsah pracovní zprávy: **40–50 stran**
Forma zpracování diplomové práce: **tištěná/elektronická**
Seznam odborné literatury:

- [1] David Lindr, "Emin und Tmin Positionierung", Separáty výzkumné zprávy, Siemens Erlangen, Německo, 2014.
- [2] Kim Doang Nguyen; I-Ming Chen; Teck-Chew Ng, "Planning algorithms for s-curve trajectories," Advanced intelligent mechatronics, 2007 IEEE/ASME international conference on , vol., no., pp.1,6, 4-7 Sept. 2007.

Vedoucí diplomové práce: **Ing. David Lindr, Ph.D.**
Ústav mechatroniky a technické informatiky
Konzultant diplomové práce: **Ing. Tomáš Turek**
ProjectSoft a.s., Hradec Králové
Datum zadání diplomové práce: **10. října 2014**
Termín odevzdání diplomové práce: **15. května 2015**


prof. Ing. Václav Kopecký, CSc.
děkan




doc. Ing. Milan Kolář, CSc.
vedoucí ústavu

V Liberci dne 10. října 2014

Prohlášení

Byl jsem seznámen s tím, že na mou diplomovou práci se plně vztahuje zákon č. 121/2000 Sb., o právu autorském, zejména § 60 – školní dílo.

Beru na vědomí, že Technická univerzita v Liberci (TUL) nezasahuje do mých autorských práv užitím mé diplomové práce pro vnitřní potřebu TUL.

Užiji-li diplomovou práci nebo poskytnu-li licenci k jejímu využití, jsem si vědom povinnosti informovat o této skutečnosti TUL; v tomto případě má TUL právo ode mne požadovat úhradu nákladů, které vynaložila na vytvoření díla, až do jejich skutečné výše.

Diplomovou práci jsem vypracoval samostatně s použitím uvedené literatury a na základě konzultací s vedoucím mé diplomové práce a konzultantem.

Současně čestně prohlašuji, že tištěná verze práce se shoduje s elektronickou verzí, vloženou do IS STAG.

Datum: 15.5.2015

Podpis: 

Poděkování

Na tomto místě bych rád poděkoval především vedoucímu diplomové práce Ing. Davidu Lindrovi, Ph.D. za cenné rady, odborné vedení, trpělivost a pomoc při zpracování této diplomové práce. Samozřejmě mé poděkování patří také celé mé rodině a přátelům za všestrannou podporu při mém vysokoškolském studiu.

Abstrakt

Cílem této diplomové práce je vytvoření metody pro generování bezrázových polohových profilů, které se použijí pro řízení astronomických teleskopů. V úvodní kapitole je rozebrána teorie řízení s omezením ryvu, tzv. s-křivky.

Praktická část dokumentu se zabývá odvozením matematického popisu pro jednotlivé trajektorie v profilu, přizpůsobení profilu pro nenulové podmínky a otestování všech možných kombinací podmínek. V závěrečné části jsou testovány možné aplikace metody.

Klíčová slova: S-křivka, polohový profil, omezení ryvu

Abstract

The aim of this thesis is to create a method for generating bezrázových position profiles, which are used for the control of astronomical telescopes. In the introductory chapter discusses the theory of management limited jerk, so. S-curve.

The practical part of the document contains a derivation of the mathematical description for each trajectory in the profile, to match the profile of the non-zero conditions and test all possible combinations of conditions. The final section tested possible application methods.

Key words: S-curve, profile position, jerk limitation

Obsah

Poděkování.....	5
Abstrakt.....	6
Abstract.....	6
Seznam ilustrací.....	9
Seznam grafů.....	10
Seznam kódů.....	11
Seznam použitých termínů a zkratk.....	12
Úvod.....	13
1 Teorie řízení.....	14
1.1 Ryv.....	15
2 Prostředky.....	17
2.1 Matlab.....	17
2.1.1 Symbolic Math Toolbox.....	18
2.2 Nelineární rovnice.....	18
2.2.1 Metoda prosté iterace.....	20
3 Astronomický teleskop.....	21
3.1 Čočkové dalekohledy – refraktory.....	21
3.2 Zrcadlové dalekohledy – reflektory.....	22
4 Matematické odvození.....	24
4.1 Interval $(0;t_r)$	25
4.2 Interval $(t_r;t_a)$	25
4.3 Interval $(t_a;t_{r2})$	26
4.4 Interval $(t_{r2}; t_v)$	26
4.5 Interval $(t_v; t_{r3})$	27
4.6 Interval $(t_{r3}; t_{a2})$	28
4.7 Interval $(t_{a2}; t_{r4})$	29
4.8 Odvození časů.....	29
4.9 Programové odvození.....	32
5 Simulační ověření.....	34
5.1 Kompenzace limitů.....	36
5.2 Řešení nelinearity.....	40
6 Záporná dráha.....	44

6.1	Matematické odvození pro zápornou dráhu	44
6.2	Odvození časů pro zápornou dráhu	48
6.3	Kompensace limitů v záporné dráze	50
6.4	Řešení nelinearity.....	52
7	Omezení maxim	54
8	Sledování žádané dráhy	59
9	Testování.....	70
	Závěr.....	72
	Použitá literatura.....	74
A	Příloha - Řešení rovnice (106)	76
B	Příloha - Řešení rovnice (107)	77
C	Příloha - Řešení rovnice (108)	78
D	Příloha - Řešení rovnice (109)	79
E	CD příloha.....	80

Seznam ilustrací

Obr. 1- Lichoběžníkový s-curve model (a) a s-curve model třetího řádu (b)[12].....	15
Obr. 2 – Skutečná trajektorie stroje.....	16
Obr. 3 – Metoda půlení intervalu [10]	20
Obr. 4 – Keplerův dalekohled [5]	21
Obr. 5 – Galileiho dalekohled [5]	22
Obr. 6 – Zrcadlové dalekohledy: 1) Gregoryho, 2) Newtonův, 3) Cassegrainův, 4) Schmidtův a 5) Maksutovův[4]	23

Seznam grafů

Graf 1 – Základní tvar	24
Graf 2 – Trajektorie ryvu a zrychlení	35
Graf 3 – Trajektorie rychlosti a dráhy	36
Graf 4 – Trajektorie ryvu a zrychlení pro kontrolní příklad 1	37
Graf 5 – Trajektorie rychlosti a dráhy pro kontrolní příklad 1	37
Graf 6 – Trajektorie ryvu a zrychlení pro kontrolní příklad 2	42
Graf 7 – Trajektorie rychlosti a dráhy pro kontrolní příklad 2	42
Graf 8 – Základní tvar 2	44
Graf 9 – Trajektorie ryvu a zrychlení pro kontrolní příklad 3	53
Graf 10 – Trajektorie rychlosti a dráhy pro kontrolní příklad 3	54
Graf 11 – Trajektorie ryvu a zrychlení pro kontrolní příklad 4	55
Graf 12 – Trajektorie rychlosti a dráhy pro kontrolní příklad 4	56
Graf 13 – Trajektorie ryvu a zrychlení pro kontrolní příklad 5	58
Graf 14 – Trajektorie rychlosti a dráhy pro kontrolní příklad 5	58
Graf 15 – Trajektorie ryvu pro sledování dráhy, příklad 1	60
Graf 16 – Trajektorie zrychlení pro sledování dráhy, příklad 1	60
Graf 17 – Trajektorie rychlosti pro sledování dráhy, příklad 1	61
Graf 18 – Trajektorie dráhy pro sledování dráhy, příklad 1	61
Graf 19 – Trajektorie ryvu pro sledování dráhy, příklad 2	63
Graf 20 – Trajektorie zrychlení pro sledování dráhy, příklad 2	64
Graf 21 – Trajektorie rychlosti pro sledování dráhy, příklad 2	64
Graf 22 – Trajektorie dráhy pro sledování dráhy, příklad 2	64
Graf 23 – Trajektorie ryvu pro první variantu	65
Graf 24 – Trajektorie ryvu pro druhou variantu	66
Graf 25 – Trajektorie zrychlení pro první variantu	66
Graf 26 – Trajektorie zrychlení pro druhou variantu	66
Graf 27 – Trajektorie rychlosti pro první variantu	67
Graf 28 – Trajektorie rychlosti pro druhou variantu	67
Graf 29 – Trajektorie dráhy pro první variantu	67
Graf 30 – Trajektorie dráhy pro druhou variantu	68
Graf 31 – Trajektorie ryvu pro sledování dráhy, příklad 2 s dobou predikce	69
Graf 32 – Trajektorie zrychlení pro sledování dráhy, příklad 2 s dobou predikce	69
Graf 33 – Trajektorie rychlosti pro sledování dráhy, příklad 2 s dobou predikce	69
Graf 34 – Trajektorie dráhy pro sledování dráhy, příklad 2 s dobou predikce	70

Seznam kódů

Kód 1 – Příklad odvození trajektorií.....	33
Kód 2 – Příklad řešení rovnic.....	33
Kód 3 – Vypsání do prostředí	35
Kód 4 – Kontrolní příklad 1.....	36
Kód 5 – Kontrolní příklad 2.....	41
Kód 6 – Kontrolní příklad 3, záporná dráha	53
Kód 7 – Špatně vypočítané hodnoty	54
Kód 8 – Kontrolní příklad 4.....	55
Kód 9 – Kontrolní příklad 5.....	57
Kód 10 – Vstupní podmínky skriptu pro sledování, příklad 1	60
Kód 11 – Vstupní podmínky skriptu pro sledování, příklad 2	63
Kód 12 – Vstupní podmínky skriptu pro porovnání variant	65
Kód 12 – Ukázka náhodných parametrů	71
Kód 12 – Ukázka výsledných časů	71

Seznam použitých termínů a zkratek

t	čas
v	rychlost
a	zrychlení
j	ryv
V_{\max}	maximální povolená rychlost
A_{\max}	maximální povolené zrychlení
R_{\max}	maximální povolený ryv

Úvod

Tato práce byla vytvořena na základě žádosti firmy ProjeSoft s požadavky na vytvoření metodiky pro generování bezrázových polohových profilů, které by byly využity v aplikacích. Toto téma se zabývá aktuálním požadavkem firmy a během práce na něm byl kladen velký důraz na splnění požadavků zadavatele.

Tématem této diplomové práce je vývoj bezrázový polohový profil pro řízení astronomického teleskopu v reálném provozu. Podmínky na bezrázovost byly dány kvůli absenci reziduálních vibrací, které se tvoří při pohybu. Bezrázového řízení lze docílit omezením hodnot ryvu, to se projeví lineárním růstem zrychlení.

Tato práce je především zaměřena na seznámení se metodikou odvozování polohových profilů s nenulovými počátečními podmínkami a s omezením hodnot ryvu, tzv. S-curve, odvození jednotlivých vztahů, vytvoření simulační funkce a její následné ověření v prostředí MATLAB. Dále pak si klade za cíl ověřit funkčnost na reálném servopohonu.

První část diplomové práce obsahuje teoretický úvod do teorie řízení s využitím omezení ryvu. Hlavní část diplomové práce obsahuje matematické odvození jednotlivých analytických řešení, odvození individuálních vztahů pro nenulové podmínky zrychlení, rychlosti a dráhy a následné simulační ověření funkčnosti.

Poslední část diplomové práce se věnuje simulační aplikaci vytvořené metody na vygenerování polohového profilu za předpokladu, že se žádané koncové hodnoty vyvíjí v čase, tedy simulace reálného použití.

1 Teorie řízení

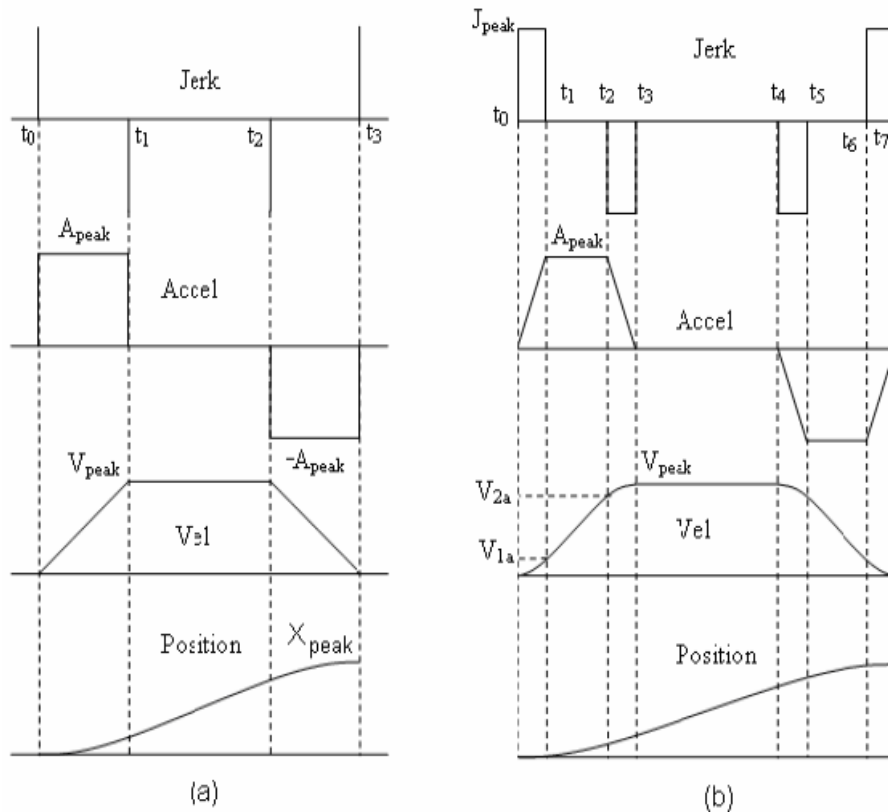
Teorii řízení lze chápat nejen ve smyslu kontroly mechanismů, ale můžeme ji použít i ve smyslu kontroly jakéhokoliv dynamického systému, například ekonomie, průmyslových procesů, dokonce i dějů v přírodě. Klíčovým při kontrole mechanismů je řízení pohybu, které má mnoho aplikací v průmyslu od výroby až po robotiku. Zapojením do uzavřené zpětné vazby spolu s regulátorem, dostaneme uzavřený regulační obvod umožňující ovládat rychlost a pozici, tím dosáhneme požadovaného pohybu. Při řízení pohybu je vždy cílem dosáhnout požadované dráhy s minimálními překmity v poloze i v rychlosti. Ve snaze splnit cíle lze naplánovat hladkou a mechanicky proveditelnou cestu.

Návrh rychlostního profilu a jeho konstrukce hraje důležitou roli v řízení pohybu veškeré techniky. V konvenčním uzavřeném smyčkovém systému je pohyb řízen regulátorem a rychlostní profil je přesně definován systémem. Regulátor má zpětnou vazbu ze senzorů umístěných v různých částech řetězce týkající se rychlosti, polohy, zrychlení a ryvu. Naměřené hodnoty se porovnávají s těmi požadovanými a na základě odchylky upravuje regulátor akční zásah tak, aby eliminoval co nejvíce možných chyb. Na základě zpětné vazby vytváří systém optimální rychlostní profil, který je hladký, rychlý a umožňuje přesný pohyb.

V systémech s otevřenou smyčkou není rychlostní profil přesně definován a vypočítává se [9]. Vzhledem k tomu že není zpětná vazba, která by sloužila k regulaci odchylek, je pohyb úplně závislý na rychlosti. V koordinovaném pohybu je potřeba vzít v úvahu fyzické omezení ryvu, zrychlení a rychlosti jednotlivých os, protože osy zařízení mohou mít různá omezení. Vzhledem k počáteční a koncové poloze je žádané využít maximálních povolených hodnot tak, aby pohyb byl rychlý, přesný a hladký. Z toho důvodu jsou s-křivky nejlepších možností. Rychlostní profil v s-křivkách je hladký a diferencovatelný do druhého řádu. Také lze do s-křivky lehce zakomponovat fyzikální omezení týkající se ryvu, zrychlení a rychlosti. Výsledkem těchto implementací mohou být trajektorie znázorněné v Obr. 1

Lichoběžníkový profil rychlosti umožňuje motoru zrychlit na maximální hodnotu rychlosti při konstantním zrychlení a následně zpomalit na nulu při konstantním zpomalení. Touto metodou lze dosáhnout velice rychlých pohybů. Nicméně, jak je

vyobrazeno v Obr. 1, v časech t_0 , t_1 , t_2 a t_3 dochází ke skokovým změnám zrychlení, tato nespojitost způsobuje, že hodnoty ryvu by teoreticky měly dosáhnout nekonečna, ale to je prakticky nemožné. V praxi je hodnota neomezeného ryvu omezena maximální dodanou energií. To způsobuje překročení limitů a potřebu vyrušení zbytkových vibrací [1]. Oba tyto nedostatky způsobují, že stroj potřebuje více času k dosažení konečné polohy s požadovanou přesností.



Obr. 1- Lichoběžníkový s-curve model (a) a s-curve model třetího řádu (b)[12]

1.1 Ryv

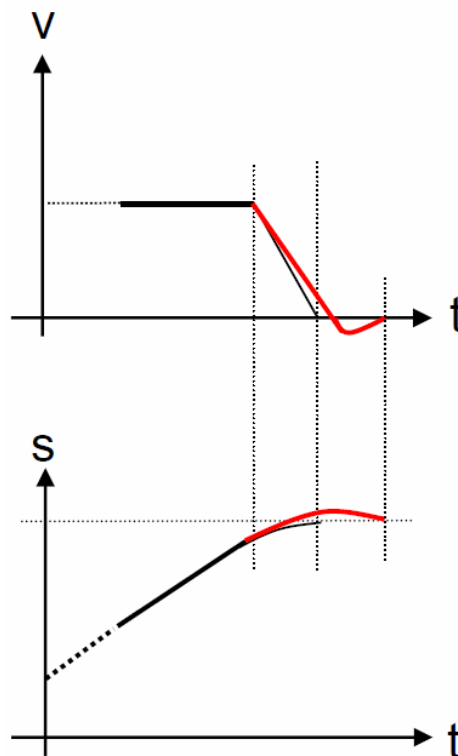
Pro popis systému z pohledu dynamiky se využívá ryv, anglicky jerk, značí se j a jeho jednotka je ms^{-3} , udává změnu zrychlení v čase. Ryv lze vypočítat podle vztahu:

$$\vec{j} = \frac{d\vec{a}}{dt} = \frac{d^2\vec{v}}{dt^2} = \frac{d^3\vec{s}}{dt^3} \quad (1)$$

Kde a je vektorem zrychlení pohybu, v je rychlost pohybu a s je dráha vykonaného pohybu.

V aplikacích pro řízení pohybu je kladen důraz na lineární pohyb z ustálené polohy do druhé. Díky omezení ryvu lze dosáhnout lineární změny zrychlení a tím získat kontrolu nad dráhou pohybu. Hlavními příklady omezení ryvu jsou aplikace v pohonu výtahu pro dopravu lidí a podpora nástrojů při obrábění. Uvádí se, [6], že většina cestujících za přijatelnou hodnotu ryvu pro vertikální směr udává 2 ms^{-3} . Ale nejlepší řady výtahů dosahují zrychlení $1,0 \text{ ms}^{-2}$ s maximální hodnotou ryvu $1,6 \text{ ms}^{-3}$. Pro nemocniční prostředí se udává doporučená hodnota ryvu $0,7 \text{ ms}^{-3}$. Ve všech případech je omezení ryvu vyžadováno pro pohodlí cestujících. Norma ISO 18738 [7] definuje měřit kvalitu jízdy ve výtahu s ohledem na ryv, zrychlení, vibrace a hluk. Dosažení nejkratší možné doby pohybu s využití maximálních hodnot ryvu, zrychlení a rychlosti bude mít za následek trajektorii dráhy ve třetím řádu.

Při neomezení ryvu dochází ke skokovým změnám při rozběhu a zastavení stroje. Do maximální hodnoty zrychlení se stroj dostane již v čase nula, viz Obr. 1a ,tento obrázek znázorňuje trajektorie, které by měl stroj sledovat. V čase t_3 v Obr. 1a dochází k zastavení stroje. Ve skutečnosti by stroj nedokázal v čase t_3 zastavit úplně a způsobil by překmit, viz Obr. 2.



Obr. 2 – Skutečná trajektorie stroje

Pro odhad o době rozběhu a doběhu lze využít optimalizovanou metodu [2]. Za pomoci s-křivek byla vytvořena dráhová trajektorie s přesným zastavením v koncovém bodě.

2 Prostředky

2.1 Matlab

V programu MATLAB je integrované prostředí pro vědeckotechnické výpočty, modelování, návrhy algoritmů, simulace, analýzu a prezentaci dat, paralelní výpočty, měření a zpracování signálů, návrhy řídicích a komunikačních systémů. Také slouží jak jako nástroj pro pohodlnou interaktivní práci, tak pro rozvoj velkého spektra aplikací. V dnešní době je prostředí MATLAB bráno jako celosvětový standart v oblasti technických výpočtů a simulací ve sféře vědy, výzkumu, průmyslu i v oblasti vzdělávání.

Název MATLAB vznikl zkrácením slov MATrix LABoratory (volně přeloženo „laboratoř s maticemi“), což napovídá, že klíčovou datovou strukturou jsou matice. Programovací jazyk, který je používán v prostředí vychází z jazyka Fortran.

MATLAB poskytuje uživatelům kromě grafických a výpočetních nástrojů také obsáhlé specializované knihovny funkcí spolu s výkonným programovacím jazykem čtvrté generace. Velké množství knihoven dělá z programu MATLAB všestranný nástroj, který lze uplatnit ve všech oblastech lidské činnosti.

Ti, kteří potřebují řešit početně náročné úlohy, ale nepotřebují důkladně zkoumat jejich matematickou součást, využijí architekturu MATLABu. Ta umožňuje částečné oprostění od matematické podstaty problému. Vlastní jazyk prostředí, který je daleko jednodušší než C nebo Fortran, využívá více než milion uživatelů. Léty ověřené výpočetní jádro, které obsahuje optimalizační algoritmy, je nejsilnější stránkou MATLABu. Zapravdu mu dává jeho využití na nejlepších pracovištích po celém světě. Další výhodou je možnost jeho využití na více platformách (Windows, Linux, Solaris, Mac) [8].

2.1.1 Symbolic Math Toolbox

Práci se symbolickými proměnnými a aritmetikou s proměnnou přesností nabízí v programu Matlab specializovaný toolbox nazvaný „Symbolic Math Toolbox“. Tento zásobní modul poskytuje stovky symbolických funkcí sloužící ke spuštění MuPAD engine, který v tomto pak slouží pro výpočet úloh např. integrace, derivování, transformace a řešení rovnic nebo pro zjednodušování výrazů.

Výhodou tohoto toolboxu je obsah kompletního jazyka MuPAD, který je modifikován pro použití se symbolickými výrazy. MuPAD obsahuje vlastní knihovnu funkcí, která slouží k mnoha úkonům, jak z oblasti základních matematických výpočtů (lineární algebra, kalkulus), tak i pro oblasti více specializované (kombinatorika, teorie čísel). Samozřejmostí je taktéž rozšiřování vestavěných funkcí díky psaní vlastních knihoven či symbolických funkcí a v jazyce MuPAD. Všechny funkce z Symbolic Math Toolboxu mohou být volány klasicky z MATLABu nebo MuPAD notebook (používaný pro zápis, dokumentaci nebo vyhodnocení symbolických výpočtů).

2.2 Nelineární rovnice

Formulace:

Je dána funkce $f: R \rightarrow R$ definovaná na intervalu $\langle a, b \rangle$. Hledáme číslo x z intervalu $\langle a, b \rangle$ tak, aby platila rovnost $f(x) = 0$. Číslo x nazveme řešení nebo kořen rovnice. [10]

Najít přesné analytické řešení je možné jen ve velice primitivních případech, např. při řešení lineární rovnice, při řešení kvadratické rovnice nebo při řešení goniometrická rovnice. Proto je ve složitějších případech nutné pro odhalení kořenů použít numerickou metodu.

Jednou z variant numerických rovnic jsou rovnice založené na iteračních principech. Pokud byla zvolena správná iterační metoda, tak by měla odpovědět na otázky:

- Konverguje posloupnost iterací ke hledanému kořenu?
- Jestliže ano, jak rychle?

Jestliže je známá informace o přibližné poloze kořenu, určitý interval $\langle a, b \rangle$, ale ne předběžná poloha, tak je použita iterační metoda, jejíž konvergence není závislá na počáteční aproximaci. Tyto takzvané konvergentní metody mají ten handicap, že konvergují velice pomalu a hodí se především pro určení počáteční aproximace pro rychleji konvergující metodu. Tato rychlejší metoda je silně závislá na správnosti počáteční aproximace a na vlastnosti funkce f v okolí hledaného kořenu. Rozumnější je rozdělit řešení nelineárních rovnic na:

- startovací metody (vždy konvergentní metody)
- zpřesňující metody
- speciální metody (např. pro polynomy)

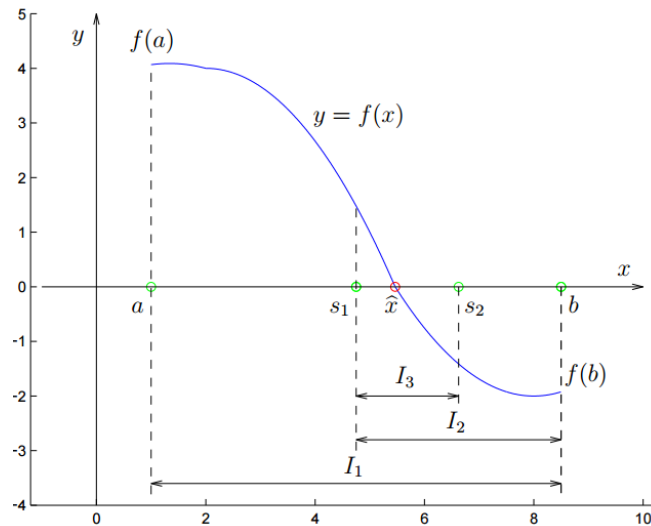
Toto rozdělení je pouze obecné. V některých případech neplatí, že zpřesňující metody vždy konvergují rychleji než startovací metody. Ty někdy mohou konvergovat mnohem rychleji.

Věta: [10]

Předpokládejme, že

- (i) *reálná funkce f je spojitá pro $x \in [a, b]$,*
- (ii) *$f(a)f(b) < 0$.*

Nejprimitivnější startovací metodou je takzvaná metoda půlení intervalu, bisekce. Za předpokladu že jsou splněny předpoklady předešlé věty, tak princip bisekce spočívá na půlení zadaného intervalu. Po rozpůlení se otestuje funkční hodnota uprostřed intervalu, má-li stejné znaménko jako funkční hodnota $f(a)$ a pak se do tohoto středu přesune bod a , v opačném případě se do něj přesune bod b . Celý postup se znovu opakuje. Pro zastavení tohoto iteračního postupu poslouží podmínka, která zkoumá velikost nově vzniklého intervalu, ta se musí snižovat. Obr. 3



Obr. 3 – Metoda půlení intervalu [10]

Metoda bisekce tedy vždy konverguje, ale velmi pomalu. Její výhodou je jednoduchost, ale i možnost předem určit počet potřebných kroků k dosažení požadované přesnosti. Nevýhodou, kromě pomalé konvergence, je nemožnost použití komplexního kořenu u polynomu.

2.2.1 Metoda prosté iterace

Princip metody spočívá v přepisu původní rovnice $f(x) = 0$ na tvar:

$$x = \varphi(x) \quad (2)$$

Existuje několik možností jak docílit tohoto přepisu. Každá konkrétní volba funkce φ ovlivní jak samotnou konvergenci metody, tak i její rychlost. Princip prosté iterace je v hledání průsečíku původní funkce $f(x)$ s přepsanou funkcí $\varphi(x)$. Nejprve je potřeba na intervalu $\langle a, b \rangle$ zvolit počáteční hodnotu iterace x_0 . Následně se zjišťují další hodnoty iterace x_k podle předpisu:

$$x_{k+1} = \varphi(x_k) \quad (3)$$

Pro zastavení iteračního postupu se volí podmínka závislá na rozdílu po sobě jdoucích iterací.

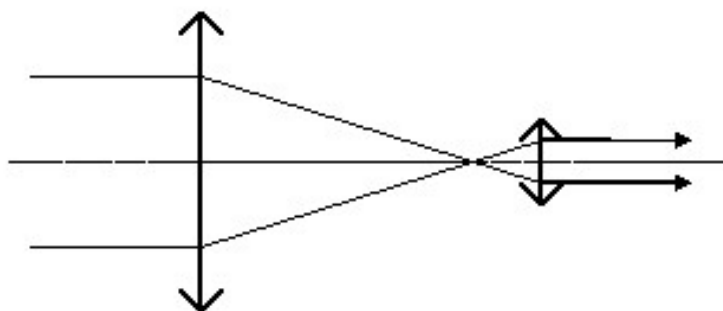
3 Astronomický teleskop

Astronomické teleskopy se dělí do dvou skupin: refraktory a reflektory. Refraktory využívají k zobrazení předmětu lom světla, kdežto reflektory využívají k zobrazení objektu jak lom světla, tak i jeho odraz. Obě tyto skupiny obsahují alespoň dva optické členy. Objektiv, který vytváří obraz pozorovaného objektu a okulár, ten slouží k pozorování obrazu vytvořeného objektivem.

Rozdíl mezi skupinami je v konstrukci. Objektiv refraktoru je tvořen jednou čočkou nebo soustavou čoček, zatímco reflektory mají objektiv tvořený zrcadlem, obvykle ve tvaru paraboloidu. Nejdůležitější výhodou reflektorů je absence barevné vady, prospěšnější uspořádání tubusu a snazší výroba zrcadel.

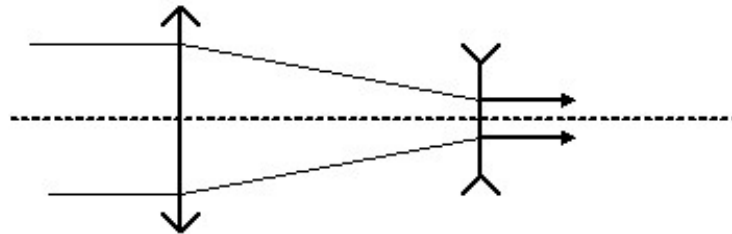
3.1 Čočkové dalekohledy – refraktory

Keplerův neboli hvězdářský dalekohled využívá dvě soustavy spojných čoček, které mají společnou optickou osu. Obraz pozorovaného předmětu se nachází v ohnisku okuláru, ten je skutečný, zmenšený a převrácený. Okulár slouží jako lupa pro pozorování. Obraz je sice zvětšený, ale zůstává převrácený, což je pro astronomická pozorování nedůležité. Tento typ dalekohledu je spíše znám pod jménem triedr a obsahuje další optické prvky, pravoúhlé optické hranoly, které vyrovnávají převrácení obrazu.



Obr. 4 – Keplerův dalekohled [5]

Druhým typem refraktoru je Galileiho neboli holandský dalekohled. Tento dalekohled je tvořen spojným objektivem a rozptylným okulárem. Obrazová ohniska objektivu i okuláru splývají v jednom bodě. Toto provedení dalekohledu se využívá jako divadelní kukátko, poskytuje přibližně čtyřnásobné zvětšení. V astronomii už používán není.



Obr. 5 – Galileiho dalekohled [5]

3.2 Zrcadlové dalekohledy – reflektory

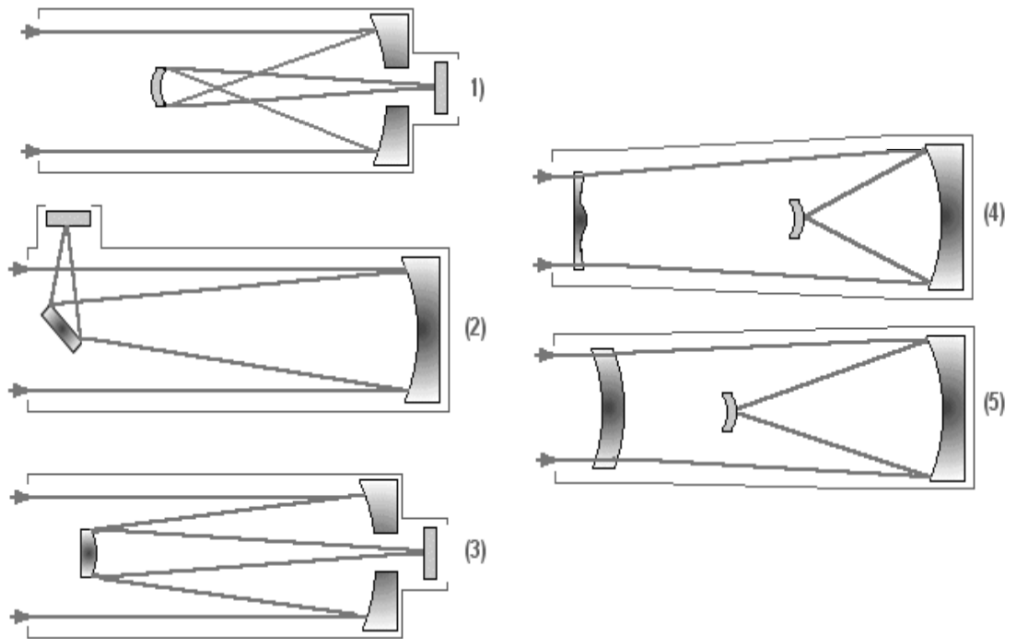
Gregoryho dalekohled je první zrcadlový teleskop. Sestavil ho skotský matematik James Gregory v roce 1663. Zpočátku se povrch zrcadla vytvářel ze směsi mědi a cínu. Dnes se používá napařování stříbra nebo hliníku. Objektiv tvoří dvě dutá zrcadla. Plocha má nejčastěji tvar eliptický, viz Obr. 6 - 1.

Issac Newton vytvořil roku 1668 odlišný druh teleskopu. Objektiv byl vytvořen kulovým, dnes parabolickým, zrcadlem. Paprsky jsou z tubusu vyvedeny za pomoci eliptického rovinného zrcadla, které svírá úhel 45° s optickou osou, viz Obr. 6 - 2. Tato snadná konstrukce je velmi populární u menších až středních teleskopů.

Cassegrainův teleskop z roku 1672 je tvořen primárním parabolickým a sekundárním hyperbolickým zrcadlem, viz Obr. 6 - 3. Světelnost pozorovaných objektů bývá u tohoto typu dalekohledu menší, je to způsobeno ohniskovou vzdáleností primárního zrcadla, která je prodloužena sekundárním.

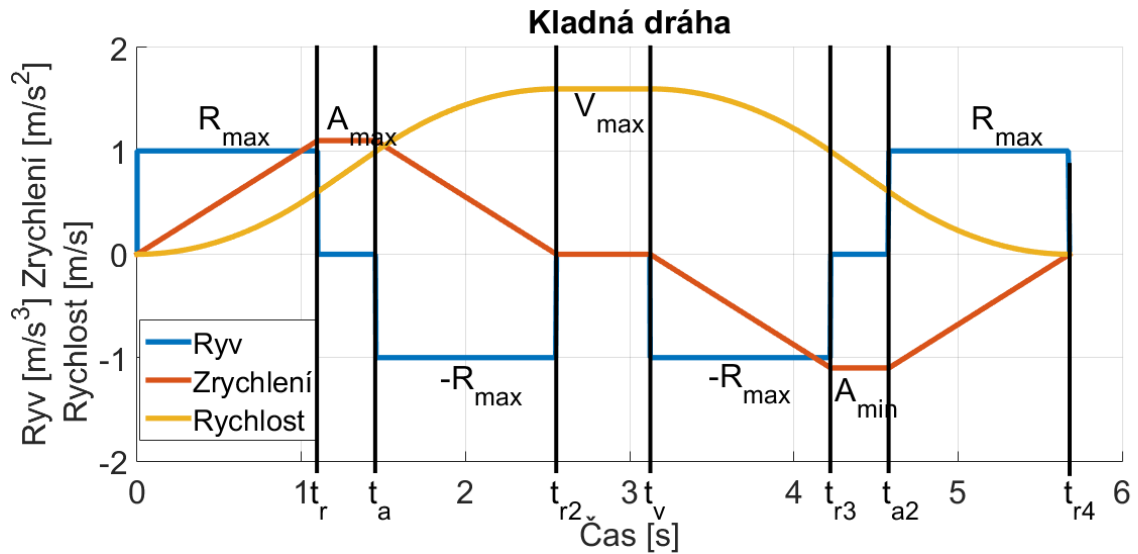
První katadioptrický dalekohled sestavil roku 1930 estonský astronom a optik Bernhard Schimdt. Objektiv je tvořen korekční deskou a kulovým zrcadlem, viz Obr. 6 - 4. Díky tomuto uspořádání lze objekty pozorovat ostře v širokém zorném poli.

Sovětský optik Dmitrij Dmitrijevič Maksutov zdokonalil předešlou varietu, korekční desku nahradil meniskem, viz Obr. 6 - 5.



Obr. 6 – Zrcadlové dalekohledy: 1) Gregoryho, 2) Newtonův, 3) Cassegrainův, 4) Schmidtův a 5) Maksutovův[4]

4 Matematické odvození



Graf 1 – Základní tvar

V Grafu 1 jsou znázorněné trajektorie ryvu, zrychlení a rychlosti. Při následném popisu bylo o intervalech uvažováno jako o samostatných částech, jen s minimální vazbou na předešlý interval. Tato vazba byla znázorněna přičtením předchozí hodnoty jedné z proměnných.

Ryv skokově mění svoji hodnotu, dosahuje buď kladné hodnoty maximálního povoleného ryvu anebo jeho záporné hodnoty, v grafu je znázorněna jako R_{max} . Těchto hodnot dosahuje pouze v okamžiku zrychlení, tedy kdy soustavu zrychluje nebo zpomaluje [11]. Zrychlení mění svoji hodnotu lineárně, dosahuje buď kladného maxima, to je označeno A_{max} , nebo záporného maxima, A_{min} . Trajektorie rychlosti lze popsat polynomem druhého řádu, ta je omezena maximální hodnotou V_{max} , té se dosáhne v intervalu mezi časy t_{r2} a t_v .

Počáteční hodnoty v Grafu 1 jsou zobrazeny jako nulové, ale v reálných úlohách mohou být nenulové, proto jsou při odvozování počáteční hodnoty reprezentovány jako: a_0, r_0 a s_0 . Začáteční hodnota ryvu r_0 je zanedbána, neboť v čase nula proběhne skok na jeho maximální hodnotu.

Stejný problém nastává s koncovými hodnotami, v Grafu 1 jsou opět vyobrazené jako nulové. Hodnoty lze měnit pouze pro zrychlení, rychlost a dráhu. Koncová hodnota ryvu nelze měnit a je brána jako nulová. V následující části budou popsány rovnice pro jednotlivé intervaly trajektorií:

4.1 Interval (0;t_r)

Rovnici popisují trajektorii ryvu, zrychlení, rychlosti a dráhy v čase 0 až t_r. V tomto intervalu dosahuje ryv povoleného maxima. Zrychlení se lineárně mění a ze své počáteční hodnoty a₀ vystoupá do svého maxima A_{max} v čase t_r.

$$r(t_1) = R_{max} \quad (4)$$

$$a(t_1) = \int r(t_1) * dt_1 + a_0 = a_0 + R_{max} * t_1 \quad (5)$$

$$v(t_1) = \int a(t_1) * dt_1 + v_0 = \frac{R_{max} * t_1^2}{2} + a_0 * t_1 + v_0 \quad (6)$$

$$s(t_1) = \int v(t_1) * dt_1 + s_0 = \frac{R_{max} * t_1^3}{6} + \frac{a_0 * t_1^2}{2} + v_0 * t_1 + s_0 \quad (7)$$

4.2 Interval (t_r;t_a)

Zrychlení při vstupu do tohoto intervalu nabylo své maximální hodnoty a je mechanicky nemožné ho nadále zvyšovat. Z toho důvodu se hodnota ryvu mění skokově na nulu, tím se splní mechanická omezení. Trajektorie rychlosti zde má lineární charakteristiku. Smysl tohoto intervalu spočívá v dosažení maximální rychlosti V_{max}, které dosáhne až ve čtvrtém intervalu.

$$r(t_2) = 0 \quad (8)$$

$$a(t_2) = \int r(t_2) * dt_2 + a(t_1 = t_r) = A_{max} \quad (9)$$

$$v(t_2) = \int a(t_2) * dt_2 + v(t_1 = t_r) \quad (10)$$

$$= \frac{R_{max} * t_r^2}{2} + a_0 * t_r + v_0 + A_{max} * t_2$$

$$\begin{aligned}
s(t_2) &= \int v(t_2) * dt_2 + s(t_1 = t_r) \\
&= s_0 + t_r * v_0 + t_2 * \left(\frac{R_{max} * t_r^2}{2} + a_0 * t_r + v_0 \right) \\
&\quad + \frac{A_{max} * t_2^2}{2} + \frac{R_{max} * t_r^3}{6} + \frac{a_0 * t_r^2}{2}
\end{aligned} \tag{11}$$

4.3 Interval ($t_a; t_{r2}$)

V tomto intervalu musí zrychlení sestoupit ze svého maxima na nulovou hodnotu. Tím zajistíme, že rychlost neporoste nad své maximum, V_{max} . Pokud počáteční zrychlení a_0 mělo nulovou hodnotu, tak časy t_r a t_{r2} jsou si rovny.

$$r(t_3) = -R_{max} \tag{12}$$

$$a(t_3) = \int r(t_3) * dt_3 + a(t_2 = t_a) = A_{max} - R_{max} * t_3 \tag{13}$$

$$\begin{aligned}
v(t_3) &= \int a(t_3) * dt_3 + v(t_2 = t_a) \\
&= \frac{R_{max} * t_r^2}{2} + a_0 * t_r + v_0 + A_{max} * t_3 + A_{max} * t_a \\
&\quad - \frac{R_{max} * t_3^2}{2}
\end{aligned} \tag{14}$$

$$\begin{aligned}
s(t_3) &= \int v(t_3) * dt_3 + s(t_2 = t_a) \\
&= s_0 + t_r * v_0 + t_a * \left(\frac{R_{max} * t_r^2}{2} + a_0 * t_r + v_0 \right) \\
&\quad + \frac{A_{max} * t_3^2}{2} + \frac{A_{max} * t_a^2}{2} + t_3 \\
&\quad * \left(\frac{R_{max} * t_r^2}{2} + a_0 * t_r + v_0 + A_{max} * t_a \right) - \frac{R_{max} * t_3^3}{6} \\
&\quad + \frac{R_{max} * t_r^3}{6} + \frac{a_0 * t_r^2}{2}
\end{aligned} \tag{15}$$

4.4 Interval ($t_{r2}; t_v$)

Trajektorie rychlosti v tomto intervalu je statická, má stále hodnotu V_{max} , viz rovnice (18). Maximální hodnota rychlosti je dána mechanickými nebo elektrickými vlastnostmi stroje.

K trajektorii dráhy z minulého intervalu přibyla rovnice přímky. Cílem toto intervalu je využít maximální rychlosti k přiblížení k žádané dráze.

$$r(t_4) = 0 \quad (16)$$

$$a(t_4) = \int r(t_4) * dt_4 + a(t_3 = t_{r2}) = 0 \quad (17)$$

$$v(t_4) = \int a(t_4) * dt_4 + v(t_3 = t_{r2}) = V_{max} \quad (18)$$

$$\begin{aligned} s(t_4) &= \int v(t_4) * dt_4 + s(t_3 = t_{r2}) \\ &= s_0 + V_{max} * t_4 + t_r * v_0 + t_a \\ &\quad * \left(\frac{R_{max} * t_r^2}{2} + a_0 * t_r + v_0 \right) + \frac{A_{max} * t_a^2}{2} + \frac{A_{max} * t_{r2}^2}{2} \\ &\quad + t_{r2} * \left(\frac{R_{max} * t_r^2}{2} + a_0 * t_r + v_0 + A_{max} * t_a \right) \\ &\quad + \frac{R_{max} * t_r^3}{6} - \frac{R_{max} * t_{r2}^3}{6} + \frac{a_0 * t_r^2}{2} \end{aligned} \quad (19)$$

4.5 Interval (t_v ; t_{r3})

Aby se splnily koncové podmínky pro trajektorii, je potřeba snížit rychlost na koncovou hodnotu v_{end} , toho docílíme pouze záporným zrychlením. Proto zde dochází ke skokové změně ryvu na záporné maximum, viz rovnice (20). Zrychlení se lineárně mění se směrnici $-R_{max}$. Z nulové hodnoty v čase t_v sestoupí na své minimum A_{min} v čase t_{r3} . Trajektorie rychlosti je kvadratická a trajektorie dráhy je kubická.

$$r(t_5) = -R_{max} \quad (20)$$

$$a(t_5) = \int r(t_5) * dt_5 + a(t_4 = t_v) = -R_{max} * t_5 \quad (21)$$

$$v(t_5) = \int a(t_5) * dt_5 + v(t_4 = t_v) = \frac{V_{max} - (R_{max} * t_5^2)}{2} \quad (22)$$

$$\begin{aligned}
s(t_5) &= \int v(t_5) * dt_5 + v(t_4 = t_v) \\
&= s_0 + V_{max} * t_5 + V_{max} * t_v + t_r * v_0 + t_a \\
&\quad * \left(\frac{R_{max} * t_r^2}{2} + a_0 * t_r + v_0 \right) + \frac{A_{max} * t_a^2}{2} \\
&\quad + \frac{A_{max} * t_{r2}^2}{2} + t_{r2} \\
&\quad * \left(\frac{R_{max} * t_r^2}{2} + a_0 * t_r + v_0 + A_{max} * t_a \right) - \frac{R_{max} * t_5^3}{6} \\
&\quad + \frac{R_{max} * t_r^3}{6} - \frac{R_{max} * t_{r2}^3}{6} + \frac{a_0 * t_r^2}{2}
\end{aligned} \tag{23}$$

4.6 Interval (t_{r3} ; t_{a2})

Zrychlení dosáhlo svého minima a je nemožné ho dále zvyšovat, proto dochází ke skokové změně ryvu na hodnotu nula. Tím je zamezeno dalšímu snižování zrychlení. Smysl toho intervalu je stejný jako u druhého intervalu, spočívá v tom, aby se dosáhlo koncové rychlosti v_{end} . Samotná kombinace pátého a sedmého intervalu by stačit nemusela, z toho důvodu je zaveden tento interval.

$$r(t_6) = 0 \tag{24}$$

$$a(t_6) = \int r(t_6) * dt_6 + a(t_5 = t_{r3}) = A_{min} \tag{25}$$

$$v(t_6) = \int a(t_6) * dt_6 + v(t_5 = t_{r3}) = V_{max} + A_{min} * t_6 - \frac{R_{max} * t_{r3}^2}{2} \tag{26}$$

$$\begin{aligned}
s(t_6) &= \int v(t_6) * dt_6 + s(t_5 = t_{r3}) \\
&= s_0 + t_6 * \left(\frac{V_{max} - (R_{max} * t_{r3}^2)}{2} \right) + V_{max} * t_{r3} + V_{max} \\
&\quad * t_v + t_r * v_0 + t_a * \left(\frac{R_{max} * t_r^2}{2} + a_0 * t_r + v_0 \right) \\
&\quad + \frac{A_{min} * t_6^2}{2} + \frac{A_{max} * t_a^2}{2} + \frac{A_{max} * t_{r2}^2}{2} + t_{r2} \\
&\quad * \left(\frac{R_{max} * t_r^2}{2} + a_0 * t_r + v_0 + A_{max} * t_a \right) + \frac{R_{max} * t_r^3}{6} \\
&\quad - \frac{R_{max} * t_{r2}^3}{6} - \frac{R_{max} * t_{r3}^3}{6} + \frac{a_0 * t_r^2}{2}
\end{aligned} \tag{27}$$

4.7 Interval (t_{a2} ; t_{r4})

V posledním intervalu je nutné zvýšit hodnotu zrychlení, tak aby konečná hodnota byla rovna a_{end} . Z toho důvodu dochází ke skokové změně ryvu na R_{max} . Pokud jsou požadované hodnoty koncového zrychlení a koncová rychlosti nulové, tak časy t_{r3} a t_{r4} si jsou rovny. V čase t_{r4} je dosažena žádaná hodnota dráhy s požadovanými hodnotami zrychlení a rychlosti.

$$r(t_7) = R_{max} \quad (28)$$

$$a(t_7) = \int r(t_7) * dt_7 + a(t_6 = t_{a2}) = A_{min} + R_{max} * t_7 \quad (29)$$

$$\begin{aligned} v(t_7) &= \int a(t_7) * dt_7 + v(t_6 = t_{a2}) \\ &= \frac{R_{max} * t_7^2}{2} + A_{min} * t_7 + V_{max} + A_{min} * t_{a2} \\ &\quad - \frac{R_{max} * t_{r3}^2}{2} \end{aligned} \quad (30)$$

$$\begin{aligned} s(t_7) &= \int v(t_7) * dt_7 + s(t_6 = t_{a2}) \\ &= s_0 + t_{a2} * \left(\frac{V_{max} - R_{max} * t_{r3}^2}{2} \right) + t_7 \\ &\quad * \left(V_{max} + A_{min} * t_{a2} - \frac{R_{max} * t_{r3}^2}{2} \right) + V_{max} * t_{r3} \\ &\quad + V_{max} * t_v + t_r * v_0 + t_a \\ &\quad * \left(\frac{R_{max} * t_r^2}{2} + a_0 * t_r + v_0 \right) + \frac{A_{min} * t_7^2}{2} + \frac{A_{min} * t_{a2}^2}{2} \\ &\quad + \frac{A_{max} * t_a^2}{2} + \frac{A_{max} * t_{r2}^2}{2} + t_{r2} \\ &\quad * \left(\frac{R_{max} * t_r^2}{2} + a_0 * t_r + v_0 + A_{max} * t_a \right) + \frac{R_{max} * t_7^3}{6} \\ &\quad + \frac{R_{max} * t_r^3}{6} - \frac{R_{max} * t_{r2}^3}{6} - \frac{R_{max} * t_{r3}^3}{6} + \frac{a_0 * t_r^2}{2} \end{aligned} \quad (31)$$

4.8 Odvození časů

Pokud jsou známy všechny trajektorie ve všech časových intervalech, tak lze vyjádřit vzorce pro jednotlivé časy. To umožní postupné generování trajektorie pro servopohon.

K vypočítání času t_r je potřeba znát maximální limity a trajektorii zrychlení v prvním intervalu, rovnice (5). Na úplném konci prvního intervalu je zrychlení rovno svému maximu A_{max} . Z této podmínky lze odvodit vztah pro t_r :

$$a(t_1 = t_r) = A_{max} \rightarrow \boxed{t_r = \frac{A_{max} - a_0}{R_{max}}} \quad (32)$$

Čas t_{r2} lze vypočítat velice podobným způsobem, je potřeba zná trajektorii zrychlení pro třetí interval, rovnice (13). Na konci třetího intervalu musí být zrychlení nulové, aby nedošlo k překročení maxima rychlosti. Z toho lze vyjádřit vzat pro t_{r2} :

$$a(t_3 = t_{r2}) = 0 \rightarrow \boxed{t_{r2} = \frac{A_{max}}{R_{max}}} \quad (33)$$

Odvození vztahu pro čas t_a už je o něco složitější oproti časům t_r a t_{r2} . Čas t_a vyjadřuje využití maximálního zrychlení pro dosažení maximální rychlosti, té bude dosaženo na konci třetího intervalu. Je tedy potřeba znát trajektorii rychlosti ve třetím intervalu, rovnice (14), a dát ji rovnou maximální hodnotě rychlosti V_{max} :

$$v(t_3 = t_{r2}) = V_{max} \rightarrow t_a = -\frac{\frac{R_{max} * t_r^2}{2} + a_0 * t_r - V_{max} + v_0 + A_{max} * t_{r2} - \frac{R_{max} * t_{r2}^2}{2}}{A_{max}} \quad (34)$$

Vztahy pro výpočet časů t_r a t_{r2} , rovnice (32) a (33), se vložily do připraveného vztahu, rovnice (34). Výsledný vztah pro čas t_a :

$$\boxed{t_a = \frac{V_{max} - v_0}{A_{max}} - \frac{2 * A_{max}^2 - a_0^2}{2 * A_{max} * R_{max}}} \quad (35)$$

Velice podobně jako času t_{r2} lze odvodit vztah pro čas t_{r3} . Rozdíl spočívá v tom, že místo snížení zrychlení z A_{max} na nulu, se sníží z nuly do svého minima A_{min} . Je potřeba dát vztah pro trajektorii zrychlení (rovnice (21)) rovnou minimální hodnotě v čase t_{r3} . Z toho lze odvodit vztah:

$$a(t_5 = t_{r3}) = A_{min} \rightarrow \boxed{t_{r3} = \frac{A_{min}}{-R_{max}}} \quad (36)$$

Vztah pro čas t_{r4} lze odvodit z koncové podmínky pro zrychlení, to musí být rovno a_{end} . Dá se trajektorie zrychlení v posledním intervalu (rovnice (29)) rovna hodnotě a_{end} a to v čase t_{r4} :

$$a(t_7 = t_{r4}) = a_{end} \rightarrow \boxed{t_{r4} = \frac{A_{min} - a_{end}}{-R_{max}}} \quad (37)$$

Při odvozování vztahu pro čas t_{a2} lze postupovat podobně jako v případě času t_a . Čas t_{a2} odpovídá době, po kterou se využívá maximální zpomalení a to z důvodu, aby se rychlost co nejrychleji přiblížila své koncové hodnotě v_{end} , té ale dosáhne až na úplném konci. Jako při odvozování času t_a , tak je možno dát trajektorii rychlosti pro poslední interval (rovnice (26)) rovnou koncové hodnotě rychlosti v_{end} v čase t_{r4} :

$$v(t_7 = t_{r4}) = v_{end} \rightarrow$$

$$t_{a2} = -\frac{\frac{R_{max} * t_{r4}^2}{2} + A_{min} * t_{r4} + V_{vend} - V_{end} - \frac{R_{max} * t_{r3}^2}{2}}{A_{min}} \quad (38)$$

Po dosazení vztahů pro časy t_{r3} a t_{r4} (rovnice (36) a (37)) do připraveného vztahu (rovnice (38)) a následné úpravě, je výsledný vztah pro t_{a2} :

$$\boxed{t_{a2} = \frac{\frac{A_{min}^2 - a_{end}^2}{2}}{A_{min} * R_{max}} - \frac{V_{max} - v_{end}}{A_{min}}} \quad (39)$$

Poslední čas, který ještě chybí odvodit, je t_v . Ten odpovídá trvání čtvrtého intervalu, ve kterém se využívá maximální rychlost k přiblížení žádané hodnoty dráhy. Aby byla možnost nastavení žádané dráhy, je nutné odvodit tento čas v závislosti na dráze. Předchozí časy jsou závislé na zrychlení nebo rychlosti. Pouze v jediném intervalu je rovna trajektorie dráhy žádané hodnotě a to v posledním (rovnice (31)). Pokud je v čase t_{r4} dráha rovná koncové hodnotě s_{req} , lze odvodit vztah pro čas t_v následovně:

$$\begin{aligned}
& s(t_7 = t_{r4}) = s_{req} \rightarrow \\
t_v = & \left(s_0 - s_{req} + t_{a2} * \left(\frac{V_{max} - (R_{max} * t_{r3}^2)}{2} \right) + t_{r4} \right. \\
& * \left(V_{max} + A_{min} * t_{a2} - \frac{R_{max} * t_{r3}^2}{2} \right) + V_{max} * t_{r3} + t_r \\
& * v_0 + t_a * \left(\frac{R_{max} * t_r^2}{2} + a_0 * t_r + v_0 \right) + \frac{A_{min} * t_{a2}^2}{2} \\
& + \frac{A_{max} * t_a^2}{2} + \frac{A_{min} * t_{r4}^2}{2} + \frac{A_{max} * t_{r2}^2}{2} + t_{r2} \\
& * \left(\frac{R_{max} * t_r^2}{2} + a_0 * t_r + v_0 + A_{max} * t_a \right) + \frac{R_{max} * t_r^3}{6} \\
& - \left. \left(\frac{R_{max} * t_{r2}^3}{6} - \frac{R_{max} * t_{r3}^3}{6} + \frac{R_{max} * t_{r4}^3}{6} + \frac{a_0 * t_r^2}{2} \right) \right) \\
& / V_{max}
\end{aligned} \tag{40}$$

Pokud se dosadí vztahy pro jednotlivé časy z předešlých rovnic (32), (33), (35), (36), (37) a (39) do připraveného vztahu (rovnice (40)) dostaneme vztah pro t_v :

$$\begin{aligned}
t_v = & \frac{A_{min}}{2 * R_{max}} - \frac{A_{max}}{2 * R_{max}} + \frac{V_{max}}{2 * A_{min}} - \frac{V_{max}}{2 * A_{max}} - \frac{s_0}{V_{max}} + \frac{s_{req}}{V_{max}} \\
& - \frac{a_0^3}{3 * R_{max}^2 * V_{max}} + \frac{a_{end}^3}{3 * R_{max}^2 * V_{max}} + \frac{v_0^2}{2 * A_{max} * V_{max}} \\
& - \frac{v_{end}^2}{2 * A_{max} * V_{max}} + \frac{a_0^4}{8 * A_{min} * R_{max}^2 * V_{max}} \\
& - \frac{a_{end}^4}{8 * A_{min} * R_{max}^2 * V_{max}} - \frac{A_{max} * v_0}{2 * R_{max} * V_{max}} \\
& + \frac{A_{min} * v_{end}}{2 * R_{max} * V_{max}} + \frac{a_0 * v_0}{R_{max} * V_{max}} - \frac{a_{end} * v_{end}}{R_{max} * V_{max}} \\
& + \frac{A_{max} + a_0^2}{4 * R_{max}^2 * V_{max}} - \frac{A_{min} + a_{end}^2}{4 * R_{max}^2 * V_{max}} \\
& - \frac{a_0^2 * v_0}{2 * A_{max} * R_{max} * V_{max}} + \frac{a_{end}^2 * v_{end}}{2 * A_{min} * R_{max} * V_{max}}
\end{aligned} \tag{41}$$

4.9 Programové odvození

Pro vyřešení výše popsaných integrací a rovnic byl vytvořen skript v programu MATLAB. Za pomoci jeho symbolického toolboxu byly vytvořeny proměnné, kde každá zastupovala jednu z trajektorií v každém intervalu, tedy celkově bylo vytvořeno 28 výrazů. Pro vyřešení integrací byla využita funkce tohoto toolboxu, která umožňuje

počítání určitého tak i neurčitého integrálu. Při volbě neurčitého jsou vstupy funkce dvě proměnné, jedna obsahuje výraz, který se bude integrovat, druhá je skalární proměnná, podle které se bude integrovat. Kód 1 zobrazuje základní odvození vztahů pro všechny trajektorie pro první tři intervaly. V prvním intervalu byly brány v potaz počáteční hodnoty. Při vytváření jednotlivých kroků skriptu bylo postupováno podle zásad, které byly popsány v dřívější kapitole (Matematické odvození).

```
%1
r1=R_max
a1=int(r1,tr)+a0
v1=int(a1,tr)+v0
s1=int(v1,tr)+s0

%2
r2=0
a2=Amax
v2=v1+int(a2,ta)
s2=s1+int(v2,ta)

%3
r3=-Rmax
a3=a2+int(r3,tr2)
v3=v2+int(a3,tr2)
s3=s2+int(v3,tr2)
```

Kód 1 – Příklad odvození trajektorií

Pro vyřešení rovnic jednotlivých časů byla použita funkce opět ze symbolického toolboxu. Funkce solve se používá pro řešení algebraických rovnic i pro soustavy lineárních rovnic o více neznámých. Pokud výraz obsahuje více symbolických proměnných, tak je potřeba dodat za parametr název proměnné, která bude brána jako neznámá, s ostatními se bude zacházet jako s konstantami. V Kódu 2 jsou znázorněné rovnice pro řešení jednotlivých časů. Při snaze o vyřešení rovnic byly využity podklady, které byly odvozeny v kapitole Odvození časů.

```
tr = solve(a1==Amax,tr)
tr2= solve(a3==0,tr2)
ta= solve(v3==Vmax,ta)
tr3 = solve(a5==Amin,tr3)
tr4= solve(a7==aend,tr4)
ta2= solve(v7==vend,ta2)
tv =solve(s7==sreq,tv)
```

Kód 2 – Příklad řešení rovnic

5 Simulační ověření

Pro ověření správnosti trajektorií byl vytvořen simulační skript, který měl za úkol kombinovat dosavadní vztahy a ověřit jejich správnost. Samotný skript se skládal ze čtyř bloků. První blok obsahoval nastavení vstupních podmínek, možnost nastavení počáteční hodnoty zrychlení, rychlosti, dráhy a také požadované hodnoty zrychlení, rychlosti a dráhy. Těchto hodnot by se mělo dosahovat vždy na konci trajektorií. Další důležitou součástí prvního bloku bylo nastavení maximálních povolených hodnot. Tyto hodnoty by měli být trvale udržitelné, neboť se jich bude po delší dobu dosahovat. Druhý blok obsahoval vztahy pro výpočet jednotlivých časů, ty byly popsány v samotné kapitole Odvození časů. Tyto časy jsou velice důležité pro předposlední blok, generování trajektorií, bez nich by nebylo možné přecházet mezi jednotlivými intervaly a ztratil by se pojem o aktuálním vývoji. Přepínání mezi jednotlivými intervaly bylo vyřešeno pomocí větvení, přímo závislém na aktuální hodnotě pole s časovými údaji. Například první interval z pohledu času trvá od nuly do t_r , druhý interval pak trvá od t_r do součtu časů t_r a t_a . Obdobně lze postupovat i u dalších intervalů.

Při každé simulaci bylo vygenerováno pět proměnných ve formátu pole. První pole obsahovalo časové údaje trajektorií. První prvek tohoto pole měl nulovou hodnotu a reprezentoval začátek trajektorie. Poslední prvek byl součtem hodnot všech dob jednotlivých intervalů, tedy celková doba trvání profilu. Rozdíl hodnot mezi dvěma sousedními prvky časového pole byl delta, tato hodnota reprezentovala možnosti řízení servopohonu, tedy jak rychle lze měnit žádanou dráhu, rychlost či zrychlení. Z toho vyplývá, že velikost polí není konstantní, ale je přímo závislá na hodnotě delta a časech všech intervalů. Posledním blokem je už samotné vypsání časů a vykreslení všech trajektorií, vždy se vykreslují v závislosti na čase. Výsledkem simulace jsou vždy čtyři grafy (každý pro jednu trajektorii) a vypsání hodnoty všech časů a vstupních podmínek do prostředí programu MATLAB (Kód 3).

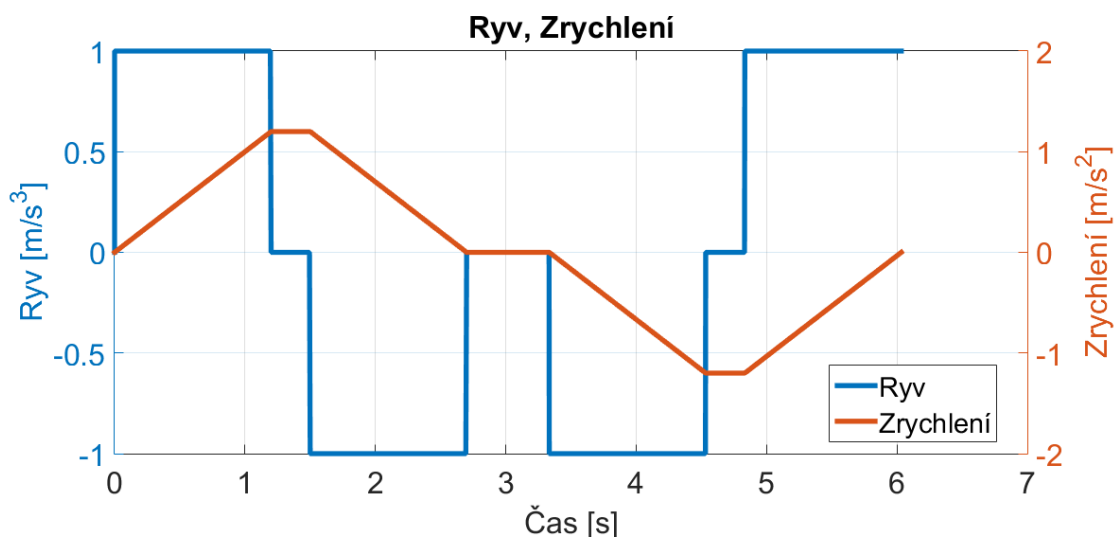
```

*****
Jerk, Acceleration, Velocity, Position
-----
R_class [m/s3]: 1.00
-----
A_max [m/s2]: 1.20
A_min [m/s2]: -1.20
A_0 [m/s2]: 0.00
A_end [m/s2]: 0.00
-----
V_max [m/s]: 1.80
V_0 [m/s]: 0.00
V_end [m/s]: 0.00
-----
S_req [m/s]: 6.00
s_0 [m/s]: 0.00
*****
Classical Profile
*****
tsum [s]: 6.033
tr [s]: 1.200
ta [s]: 0.300
tr2 [s]: 1.200
tv [s]: 0.633
tr3 [s]: 1.200

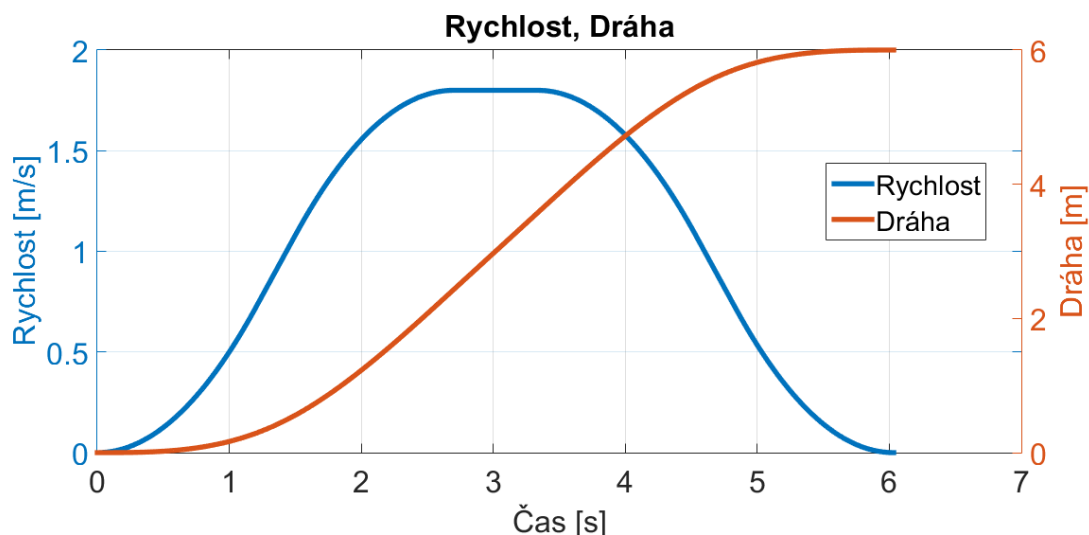
```

Kód 3 – Vypsání do prostředí

Pro první simulační odzkoušení bylo použito stejné nastavení vstupních podmínek, jako je zobrazeno v Kódu 3, výsledné trajektorie jsou zobrazeny v Grafu 2 a Grafu 3. Žádaná dráha byla nastavena dostatečně dlouhá, tak aby dosáhly trajektorie zrychlení a rychlosti svých maxim.



Graf 2 – Trajektorie ryvu a zrychlení



Graf 3 – Trajektorie rychlosti a dráhy

5.1 Kompenzace limitů

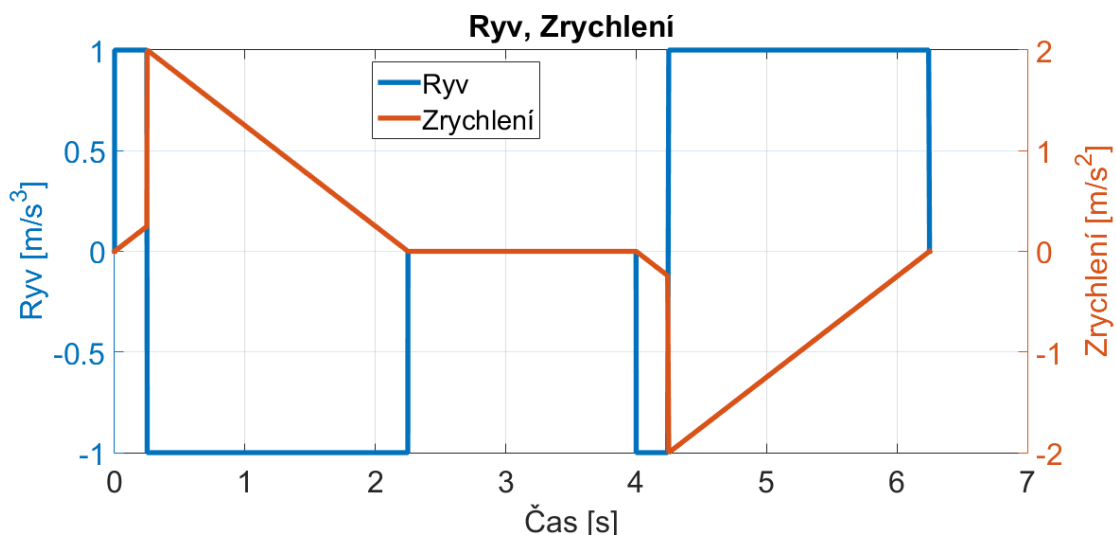
Problém nastával, když byla žádaná dráha malá nebo když byly limity nastaveny velice vysoko, tedy trajektorie jich nemusely dosáhnout, ale byly k tomu nuceny. Obdobně jako v následujícím případě. Nastavení simulace, které bylo použito, je vypsáno v Kódu 4. Už ze zobrazených časů je patrné, že tato metoda není korektní. Trajektorie simulace jsou zobrazeny na Grafu 4 a Grafu 5.

```

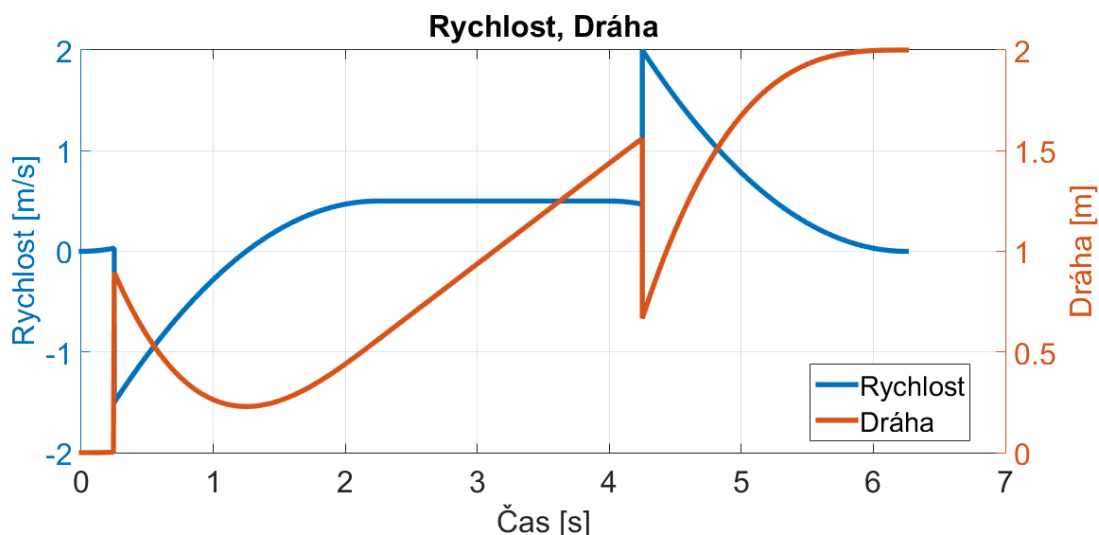
*****
Jerk, Acceleration, Velocity, Position
-----
R_class [m/s3]: 1.00
-----
A_max [m/s2]: 2.00
A_0 [m/s2]: 0.00
A_end [m/s2]: 0.00
-----
V_max [m/s]: 0.50
V_0 [m/s]: 0.00
V_end [m/s]: 0.00
-----
S_req [m/s]: 2.00
s_0 [m/s]: 0.00
*****
Classical Profile
*****
tsum [s]: 6.250
tr [s]: 2.000
ta [s]: -1.750
tr2 [s]: 2.000
tv [s]: 1.750
tr3 [s]: 2.000
ta2 [s]: -1.750
tr4 [s]: 2.000

```

Kód 4 – Kontrolní příklad 1



Graf 4 – Trajektorie ryvu a zrychlení pro kontrolní příklad 1



Graf 5 – Trajektorie rychlosti a dráhy pro kontrolní příklad 1

Maximální hodnota povoleného zrychlení byla nastavena velice vysoko a trajektorie rychlosti by dosáhla svého maxima, aniž by zrychlení muselo dosahovat hodnoty A_{max} , tedy čas t_a by musel vyjít nulový, nikoliv záporný. Z tohoto předpokladu byla odvozena rovnice (42), kde ve vzorci (35) pro čas t_a bylo použito hodnota maximálního dosaženého zrychlení A_H místo A_{max} .

$$t_a(A_{max} = A_H) = 0 \rightarrow \frac{v_0 - \frac{V_{max} + A_H^2}{2 * R_{max}} + \frac{(A_H - a_0)^2}{2 * R_{max}} + \frac{a_0 * (A_H - a_0)}{R_{max}}}{A_H} = 0 \quad (42)$$

$$A_{H1} = \frac{2^{\frac{1}{2}} * \sqrt{a_0^2 + 2 * R_{max} * V_{max} - 2 * R_{max} * v_0}}{2} \quad (43)$$

$$A_{H2} = -\frac{2^{\frac{1}{2}} * \sqrt{a_0^2 + 2 * R_{max} * V_{max} - 2 * R_{max} * v_0}}{2} \quad (44)$$

Rovnice má dvě řešení A_{H1} a A_{H2} , kde druhé řešení je pouze opačným řešením prvního. Neboť v první části trajektorie dochází ke zrychlení, tedy maximální dosažená hodnota zrychlení musí být kladná, bylo použito řešení (43).

Podobný předpoklad byl použit pro výpočet maximální hodnoty zpomalení A_{LOH} . Opět není potřeba dosahovat A_{min} proto, aby byla dosažena požadovaná koncová rychlost v_{end} . Čas t_{a2} , rovnice (39), byl položen jako nulový a rovnice dostala tvar:

$$\frac{t_a(A_{max} = A_H) = 0 \rightarrow v_{end} - \frac{V_{max} + A_{LOH}^2}{2 * R_{max}} - \frac{(A_{LOH} - a_{end})^2}{2 * R_{max}} + \frac{A_{LOH} * (A_{LOH} - a_{end})}{R_{max}}}{A_{LOH}} = 0 \quad (45)$$

$$A_{LOH1} = -\frac{2^{\frac{1}{2}} * \sqrt{a_{end}^2 + 2 * R_{max} * V_{max} - 2 * R_{max} * v_{end}}}{2} \quad (46)$$

$$A_{LOH2} = \frac{2^{\frac{1}{2}} * \sqrt{a_{end}^2 + 2 * R_{max} * V_{max} - 2 * R_{max} * v_{end}}}{2} \quad (47)$$

Stejně jako rovnice (42) má rovnice (45) dvě řešení, opět je druhé řešení opačným řešením prvního. Ve třetí fázi trajektorií dochází k celkovému zpomalení, minimální dosažené zrychlení by mělo být záporné, proto bylo použito řešení (46).

Podobný problém byl objeven i v rychlostním profilu. Opět vynucené využití rychlostního maxima mělo za následek, že doba časového intervalu t_v byla záporná. Tedy k docílení žádané hodnoty nebylo potřeba dosahovat maxima rychlosti, ale jen hodnoty V_H . Při odvozování vztahu se vycházelo ze stejného principu jako při profilu zrychlení. Rovnice pro čas t_v (41) se položila rovna nule:

$$\begin{aligned}
& t_v(V_{max} = V_H) = 0 \\
& (3 * A_{min} * a_0^4 - 3 * A_{max} * a_{end}^4 + 6 * A_{min} * A_{max}^2 * a_0^2 - 6 * A_{min}^2 \\
& * A_{max} * a_{end}^2 + 12 * A_{max} * R_{max}^2 * V_H^2 - 12 * A_{min} \\
& * R_{max}^2 * V_{max}^2 + 12 * A_{min} * R_{max}^2 * v_0^2 - 12 * A_{max} \\
& * R_{max}^2 * v_{end}^2 - 8 * A_{min} * A_{max} * a_0^3 + 8 * A_{min} \\
& * A_{max} * a_{end}^3 - 24 * A_{min} * A_{max} * R_{max}^2 * S_0 + 24 \\
& * A_{min} * A_{max} * R_{max}^2 * S_{req} - 12 * A_{min} * A_{max}^2 \\
& * R_{max} * v_0 + 12 * A_{min}^2 * A_{max} * R_{max} * v_{end} - 12 \\
& * A_{min} * R_{max} * a_0^2 * v_0 + 12 * A_{max} * R_{max} * a_{end}^2 \\
& * v_{end} + 12 * A_{min}^2 * A_{max} * R_{max} * V_H - 12 * A_{min} \\
& * A_{max}^2 * R_{max} * V_{max} + 24 * A_{min} * A_{max} * R_{max} * a_0 \\
& * v_0 - 24 * A_{min} * A_{max} * R_{max} * a_{end} * v_{end}) \\
& / (24 * A_{min} * A_{max} * R_{max}^2 * V_H) = 0
\end{aligned} \tag{48}$$

$$\begin{aligned}
V_{H1} = & \left(\frac{1}{3^{\frac{1}{2}}} * (A_{max} - A_{min}) \right. \\
& * (3 * A_{max} * a_{end}^4 - 3 * A_{min} * a_0^4 - 3 * A_{max}^2 * A_{min}^3 \\
& + 3 * A_{max}^3 * A_{min}^2 - 6 * A_{max}^2 * A_{min} * a_0^2 + 12 \\
& * A_{max} * R_{max}^2 * v_{end}^2 - 12 * A_{min} * R_{max}^2 * v_0^2 - 8 \\
& * A_{max} * A_{min} * a_{end}^3 + 8 * A_{max} * A_{min} * a_0^3 + 6 \\
& * A_{max} * A_{min}^2 * a_{end}^2 + 24 * A_{max} * A_{min} * R_{max}^2 * S_0 \\
& + 12 * A_{max}^2 * A_{min} * R_{max} * v_0 + 12 * A_{min} * R_{max} \\
& * a_0^2 * v_0 - 24 * A_{max} * A_{min} * R_{max}^2 * S_{req} - 12 * A_{max} \\
& * A_{min}^2 * R_{max} * v_{end} - 12 * A_{max} * a_{end}^2 * R_{max} * v_{end} \\
& - 24 * A_{max} * A_{min} * R_{max} * a_0 * v_0 + 24 * A_{max} \\
& * A_{min} * a_{end} * R_{max} * v_{end})^{\frac{1}{2}} - 3 * A_{max} * A_{min}^2 + 3 \\
& \left. * A_{max}^2 * A_{min} \right) / (6 * R_{max} * (A_{max} - A_{min}))
\end{aligned} \tag{49}$$

Obdobně jako v případech zrychlení i u rychlosti měla rovnice (48) dvě řešení. Nevyobrazené řešení je opět opačným řešením (49). Bylo zvoleno kladné řešení rovnice z důvodu kladné koncové dráhy. Tímto byly vyřešeny problémy s nuceným využitím maximálních omezení, ale tyto dosavadní výsledky byly řešením pro každý problém zvlášť a jejich kombinace nebyla možná. Bylo třeba vyřešit možnost jejich kombinace, v nejlepším případě zkombinovat všechny tři možnosti do jednoho výpočetního bloku. Zde ale nastal problém, kdy pro vypočítání maximální dosažené hodnoty zrychlení byla použita hodnota maximální povolené rychlosti V_{max} , ale v obecném profilu nemusí dojít k dosažení této hodnoty, v tomto případě nastala kombinace dvou více popsáných problému. Z této úvahy vyplynulo, že maximální dosažené zrychlení i zpomalení by se

mělo počítat z maximální dosažené rychlosti a vstupních podmínek. Stejný předpoklad platí i opačně pro výpočet maximální rychlosti. Toto lze matematicky zapsat následovně:

$$A_H = f(V_H, a_0, v_0, R_{max}) \quad (50)$$

$$A_{LOH} = f(V_H, a_{end}, v_{end}, R_{max}) \quad (51)$$

$$V_H = f(A_H, A_{LOH}, a_0, v_0, s_0, a_{end}, v_{end}, s_{req}, R_{max}) \quad (52)$$

5.2 Řešení nelinearity

Tímto se z rovnic stala nelineární soustava. Není možné vyjádřit hodnotu V_H pouze v závislosti na maximálním ryvu, počátečních a koncových podmínkách. Tedy nalezení přesného řešení v jednom kroku je nemožné. Přesněji nalezení přesného řešení v jednom kroku je nemožné, neboť pro výpočet hodnoty V_H jsou třeba i hodnoty A_H a A_{LOH} . Jakákoliv změna hodnoty zrychlení se pojí se změnou maximální dosažené rychlosti, ale tím se změní i samotná hodnota zrychlení.

Bylo potřeba vypočítávat hodnoty v cyklu, kde by se v každém kroku vypočítala aktuální maxima. K jejich vypočítání byla použita maxima z předchozího kroku. Cyklus byl ukončen tehdy, kdy už nedocházelo ke změnám hodnot, tedy předchozí a aktuální maximum byly stejné. Pro využití v cyklu byly vztahy (50), (51) a (52) přepsány na:

$$A_H(k) = f(V_H(k-1), a_0, v_0, R_{max}) \quad (53)$$

$$A_{LOH}(k) = f(V_H(k-1), a_{end}, v_{end}, R_{max}) \quad (54)$$

$$V_H(k) = f(A_H(k-1), A_{LOH}(k-1), a_0, v_0, s_0, a_{end}, v_{end}, s_{req}, R_{max}) \quad (55)$$

Pro urychlení výpočtu byla použita startovací metoda taková, že počáteční hodnoty A_H , A_{LOH} a V_H se vypočítali z povolených maxim, tedy:

$$A_H = f(V_{max}, a_0, v_0, R_{max}) \quad (56)$$

$$A_{LOH} = f(V_{max}, a_{end}, v_{end}, R_{max}) \quad (57)$$

$$V_H = f(A_{max}, A_{min}, a_0, v_0, s_0, a_{end}, v_{end}, s_{req}, R_{max}) \quad (58)$$

Při každém výpočtu dosažených maxim bylo nutné kontrolovat, zda jejich hodnota nepřesahuje povolená maxima. V případě, že hodnota přesahovala toto maximum, tak byla pro další výpočty použita hodnota povoleného maxima. Například, když vypočtená hodnota A_H vyšla $6,046 \text{ ms}^{-2}$, ale povolené maximum bylo 5 ms^{-2} , tak bylo použito pro další výpočty právě to maximum A_{max} s hodnotou 5 ms^{-2} . Jako kontrola správné funkčnosti cyklu mohly sloužit vypočítané časy intervalů t_a , t_{a2} a t_v . Pokud nedocházelo k využití omezení, tak jejich hodnota byla velice blízko nule (dosahovaly hodnot 10^{-6}).

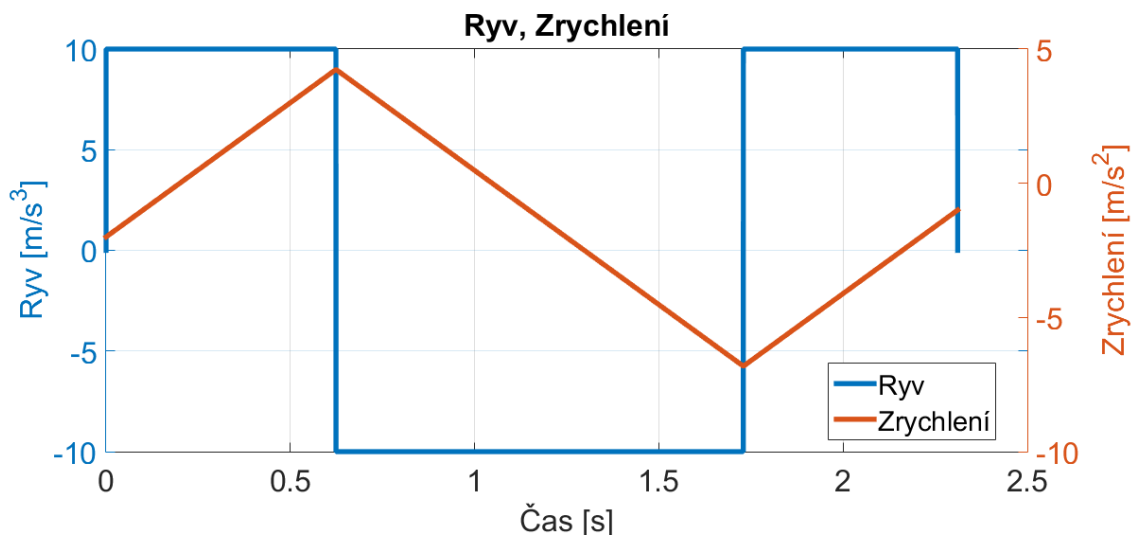
Výše popsaný iterační cyklus byl vložen do simulačního skriptu jako vlastní blok a stal se jeho nejdůležitější součástí, ale postupem času také nejnáročnější. V následujícím příkladu byla demonstrována funkčnost tohoto cyklu a jeho správná implementace do skriptu. Dráha i omezení byly voleny tak, aby nedocházelo k využití plného rozsahu povolených hodnot.

```

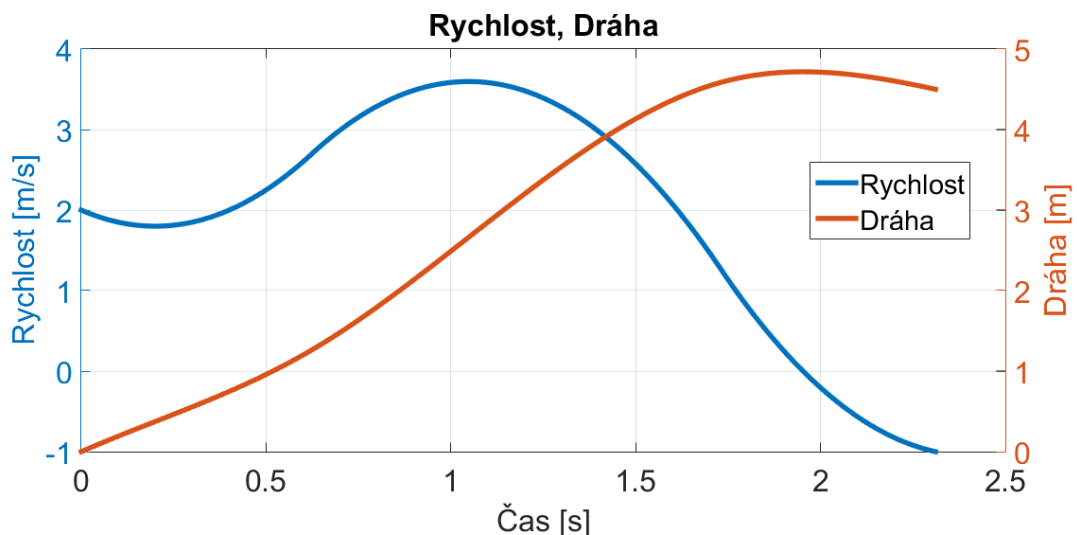
*****
Jerk, Acceleration, Velocity, Position
-----
R_class [m/s3]: 10.00
-----
A_max [m/s2]: 8.00
A_H [m/s2]: 4.24
A_LO_H [m/s2]: -6.82
A_0 [m/s2]: -2.00
A_end [m/s2]: -1.00
-----
V_max [m/s]: 8.00
V_H [m/s]: 3.60
V_0 [m/s]: 2.00
V_end [m/s]: -1.00
-----
S_req [m/s]: 4.50
s_0 [m/s]: 0.00
*****
Classical Profile
*****
tsum [s]: 2.311
tr [s]: 0.624
tr2 [s]: 0.424
tr3 [s]: 0.682
tr4 [s]: 0.582
ta [s]: 0.000
ta2 [s]: 0.000
tv [s]: -0.000

```

Kód 5 – Kontrolní příklad 2



Graf 6 – Trajektorie rychlosti a zrychlení pro kontrolní příklad 2



Graf 7 – Trajektorie rychlosti a dráhy pro kontrolní příklad 2

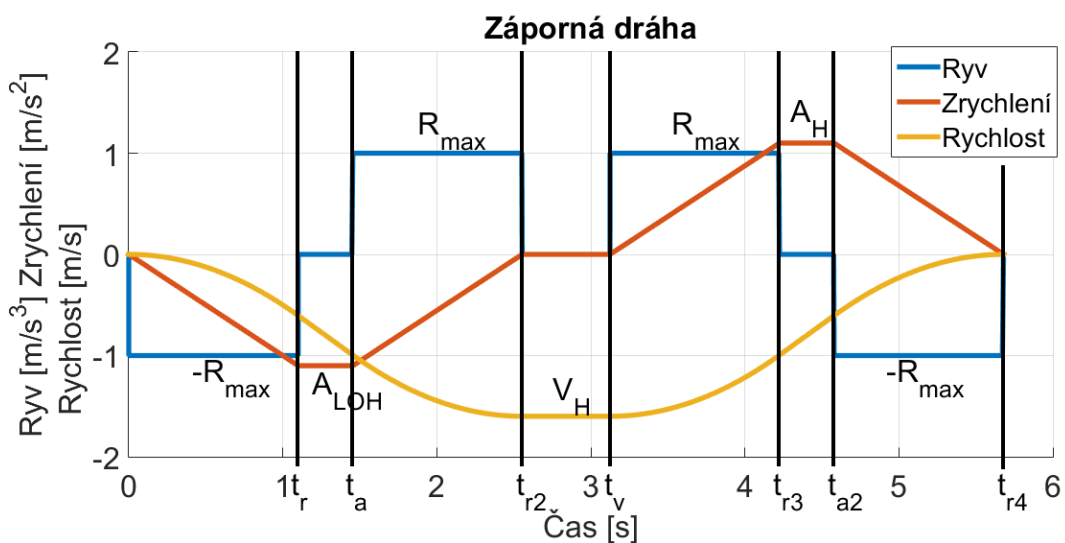
Z kontrolního příkladu 2 je dobře patrné, že blok pro výpočet maximálních hodnot fungoval správně, vypočítané časy t_a , t_{a2} a t_v měly hodnotu menší než jedna nanosekunda. Tato jejich velikost byla způsobena zaokrouhlováním během výpočetního cyklu, ale nijak neovlivňuje správnou funkci skriptu. V Grafu 6 je dobře zřejmé jak zrychlení plynule přechází na zpomalení a opačně. To způsobilo, že trajektorie rychlosti a dráhy byly hladší než v předchozím případě. Vynecháním druhého, čtvrtého a šestého intervalu způsobilo, že došlo za celou dobu trvání profilu pouze ke čtyřem změnám rychlosti oproti předchozím osmi, což v praxi znamenalo menší rázy celkového systému. Toto vynechání způsobilo,

že profil rychlosti byl po celou dobu simulace polynom druhého řádu a profil dráhy byl třetího.

V kontrolním příkladu 2 byla taky odzkoušena možnost nenulových počátečních a koncových podmínek. Z Grafu 6 a Grafu 7 je patrné, že se profily dobře přizpůsobily nenulovým podmínkám. Kladná hodnota počáteční rychlosti způsobila, že profil zrychlení musel dosáhnout menší maximální hodnoty oproti případu s nulovými počátečními podmínkami. V případě, že by byly oboje podmínky nulové, tak maximální hodnota zrychlení by byla $6,08 \text{ ms}^{-2}$, to je o $1,84 \text{ ms}^{-2}$ více než v zobrazeném příkladu. Oproti tomu záporná koncová hodnota rychlosti způsobila, že bylo třeba dosáhnout vyššího zpomalení, v tomto případě byl rozdíl $0,74 \text{ ms}^{-2}$. Zajímavostí bylo, že v průběhu trajektorie dráhy došlo k překročení žádané dráhy, to bylo způsobeno požadavkem na koncové hodnoty zrychlení a rychlosti. Bez zmiňovaného překmitu by nebyla možnost jak dosáhnout žádané dráhy s požadovanými koncovými podmínkami.

6 Záporná dráha

Jedním z nedostatků dosavadního skriptu byla nemožnost simulace profilu se zápornou koncovou dráhou. To bylo způsobeno základním tvarem profilů, který byl použit při odvozování vztahů, ten umožňuje užití pro zápornou dráhu. Bylo tedy potřeba navrhnout nový základní tvar pro zápornou koncovou dráhu. V tomto profilu by rychlost dosahovala záporných hodnot po většinu své trajektorie. Profily ryvu a zrychlení by měly mít obrácený tvar než v základním tvaru. Při využití těchto znalostí byl vytvořen druhý základní tvar zobrazený v Grafu 8.



Graf 8 – Základní tvar 2

Pro zjištění matematických rovnic popisující jednotlivé intervaly bylo postupováno stejně jako v dřívějším odvození, to bylo popsáno v kapitole: Matematické odvození. Jediný rozdíl byl v tom, že zde již nebylo uvažováno o maximálních povolených hodnotách (V_{max} , A_{max} a A_{min}), ale bylo již počítáno s maximálními dosaženými hodnotami: V_H , A_H a A_{LOH} . Tento fakt pouze mění název proměnných ve výsledném popisu. Opět byly pro jednoduchou představu zobrazeny počáteční a koncové hodnoty nulové.

6.1 Matematické odvození pro zápornou dráhu

V prvním intervalu dochází k zápornému růstu zrychlení, tedy hodnota ryvu je na svém záporném maximu. Zrychlení mění svoji hodnotu lineárně se směrnicí odpovídající $-R_{max}$,

klesá od hodnoty a_0 až do svého minima A_{LOH} , toho dosáhne na konci prvního intervalu, tedy v čase t_r

$$r(t_1) = -R_{max} \quad (59)$$

$$a(t_1) = \int r(t_1) * dt_1 + a_0 = a_0 - R_{max} * t_1 \quad (60)$$

$$v(t_1) = \int a(t_1) * dt_1 + v_0 = v_0 + a_0 * t_1 - \frac{R_{max} * t_1^2}{2} \quad (61)$$

$$s(t_1) = \int v(t_1) * dt_1 + s_0 = -\frac{R_{max} * t_1^3}{6} + \frac{a_0 * t_1^2}{2} + v_0 * t_1 + s_0 \quad (62)$$

Zrychlení by v tomto intervalu již mělo být na svém maximu, a proto ryv na začátku intervalu skokově mění svoji hodnotu na nulu. Tím je zajištěno, že zrychlení neporoste. Obdobně jako druhý interval v profilech pro kladnou dráhu má tento interval za cíl pomoci dosáhnout minima rychlosti.

$$r(t_2) = 0 \quad (63)$$

$$a(t_2) = \int r(t_2) * dt_2 + a(t_1 = t_r) = A_{LOH} \quad (64)$$

$$v(t_2) = \int a(t_2) * dt_2 + v(t_1 = t_r) \quad (65)$$

$$= v_0 + A_{LOH} * t_2 + a_0 * t_r - \frac{R_{max} * t_r^2}{2}$$

$$s(t_2) = \int v(t_2) * dt_2 + s(t_1 = t_r) \quad (66)$$

$$= s_0 + t_r * v_0 + t_2 * \left(v_0 + a_0 * t_r - \frac{R_{max} * t_r^2}{2} \right)$$

$$+ \frac{A_{LOH} * t_2^2}{2} - \frac{R_{max} * t_r^3}{6} + \frac{a_0 * t_r^2}{2}$$

V následujícím intervalu dojde ke kladnému růstu zrychlení na nulovou hodnotu. Tím bude zajištěno, že rychlost nesestoupí níže než je potřebná hodnota V_H .

$$r(t_3) = R_{max} \quad (67)$$

$$a(t_3) = \int r(t_3) * dt_3 + a(t_2 = t_a) = A_{LOH} + R_{max} * t_3 \quad (68)$$

$$\begin{aligned}
v(t_3) &= \int a(t_3) * dt_3 + v(t_2 = t_a) \\
&= \frac{R_{max} * t_3^2}{2} + A_{LOH} * t_3 + v_0 + A_{LOH} * t_a + a_0 * t_r \\
&\quad - \frac{R_{max} * t_r^2}{2}
\end{aligned} \tag{69}$$

$$\begin{aligned}
s(t_3) &= \int v(t_3) * dt_3 + s(t_2 = t_a) \\
&= s_0 + t_r * v_0 + t_a * \left(v_0 + a_0 * t_r - \frac{R_{max} * t_r^2}{2} \right) \\
&\quad + \frac{A_{LOH} * t_3^2}{2} + \frac{A_{LOH} * t_a^2}{2} + t_3 \\
&\quad * \left(v_0 + A_{LOH} * t_a + a_0 * t_r - \frac{R_{max} * t_r^2}{2} \right) + \frac{R_{max} * t_3^3}{6} \\
&\quad - \frac{R_{max} * t_r^3}{6} + \frac{a_0 * t_r^2}{2}
\end{aligned} \tag{70}$$

Čtvrtý interval má za cíl využít minimální rychlosti, viz rychlostní rovnice (73), k nejrychlejšímu růstu dráhy a tím přiblížení žádané hodnoty.

$$r(t_4) = 0 \tag{71}$$

$$a(t_4) = \int r(t_4) * dt_4 + a(t_3 = t_{r2}) = 0 \tag{72}$$

$$v(t_4) = \int a(t_4) * dt_4 + v(t_3 = t_{r2}) = V_H \tag{73}$$

$$\begin{aligned}
s(t_4) &= \int v(t_4) * dt_4 + s(t_3 = t_{r2}) \\
&= s_0 + V_H * t_4 + t_r * v_0 + t_a \\
&\quad * \left(v_0 + a_0 * t_r - \frac{R_{max} * t_r^2}{2} \right) + \frac{A_{LOH} * t_a^2}{2} + \frac{A_{LOH} * t_{r2}^2}{2} \\
&\quad + t_{r2} * \left(v_0 + A_{LOH} * t_a + a_0 * t_r - \frac{R_{max} * t_r^2}{2} \right) \\
&\quad - \frac{R_{max} * t_r^3}{6} + \frac{R_{max} * t_{r2}^3}{6} + \frac{a_0 * t_r^2}{2}
\end{aligned} \tag{74}$$

Aby byla možnost splnění koncových podmínek, je třeba pomalu směřovat s rychlostní trajektorií ke koncové hodnotě, tedy je nutno zrychlit, proto zde dochází ke skokové změně ryvu. Zrychlení z nulové hodnoty vyroste na své maximum A_H v čase t_{r3} .

$$r(t_5) = R_{max} \tag{75}$$

$$a(t_5) = \int r(t_5) * dt_5 + a(t_4 = t_v) = R_{max} * t_5 \quad (76)$$

$$v(t_5) = \int a(t_5) * dt_5 + v(t_4 = t_v) = \frac{R_{max} * t_5^2}{2} + V_H \quad (77)$$

$$\begin{aligned} s(t_5) &= \int v(t_5) * dt_5 + v(t_4 = t_v) \\ &= s_0 + V_H * t_5 + V_H * t_v + t_r * v_0 + t_a \\ &\quad * \left(v_0 + a_0 * t_r - \frac{R_{max} * t_r^2}{2} \right) + \frac{A_{LOH} * t_a^2}{2} + \frac{A_{LOH} * t_{r2}^2}{2} \\ &\quad + t_{r2} * \left(v_0 + A_{LOH} * t_a + a_0 * t_r - \frac{R_{max} * t_r^2}{2} \right) \\ &\quad + \frac{R_{max} * t_5^3}{6} - \frac{R_{max} * t_r^3}{6} + \frac{R_{max} * t_{r2}^3}{6} + \frac{a_0 * t_r^2}{2} \end{aligned} \quad (78)$$

V tomto intervalu zrychlení dosáhlo svého maxima, a proto dochází ke změně hodnoty ryvu, není třeba nadále zvyšovat zrychlení. Cílem šestého intervalu je dosáhnout koncové rychlosti v_{end} , pátý a sedmý interval by pro splnění této podmínky nemusel stačit.

$$r(t_6) = 0 \quad (79)$$

$$a(t_6) = \int r(t_6) * dt_6 + a(t_5 = t_{r3}) = A_H \quad (80)$$

$$v(t_6) = \int a(t_6) * dt_6 + v(t_5 = t_{r3}) = \frac{R_{max} * t_{r3}^2}{2} + V_H + A_H * t_6 \quad (81)$$

$$\begin{aligned} s(t_6) &= \int v(t_6) * dt_6 + s(t_5 = t_{r3}) \\ &= s_0 + t_6 * \left(\frac{R_{max} * t_{r3}^2}{2} + V_H \right) + V_H * t_{r3} + V_H * t_v \\ &\quad + t_r * v_0 + t_a * \left(v_0 + a_0 * t_r - \frac{R_{max} * t_r^2}{2} \right) + \frac{A_H * t_6^2}{2} \\ &\quad + \frac{A_{LOH} * t_a^2}{2} + \frac{A_{LOH} * t_{r2}^2}{2} + t_{r2} \\ &\quad * \left(v_0 + A_{LOH} * t_a + a_0 * t_r - \frac{R_{max} * t_r^2}{2} \right) - \frac{R_{max} * t_r^3}{6} \\ &\quad + \frac{R_{max} * t_{r2}^3}{6} + \frac{R_{max} * t_{r3}^3}{6} + \frac{a_0 * t_r^2}{2} \end{aligned} \quad (82)$$

V posledním intervalu dochází ke snížení zrychlení na hodnotu a_{end} , proto záporná hodnota ryvu. Na konci tohoto intervalu by měli být splněny všechny koncové podmínky a trajektorie ukončena.

$$r(t_7) = -R_{\max} \quad (83)$$

$$a(t_7) = \int r(t_7) * dt_7 + a(t_6 = t_{a2}) = A_H - R_{\max} * t_7 \quad (84)$$

$$v(t_7) = \int a(t_7) * dt_7 + v(t_6 = t_{a2}) = \frac{R_{\max} * t_{r3}^2}{2} + V_H + A_H * t_7 + A_H * t_{a2} - \frac{R_{\max} * t_7^2}{2} \quad (85)$$

$$s(t_7) = \int v(t_7) * dt_7 + s(t_6 = t_{a2}) = s_0 + t_{a2} * \left(\frac{R_{\max} * t_{r3}^2}{2} + V_H \right) + t_7 * \left(\frac{R_{\max} * t_{r3}^2}{2} + V_H + A_H * t_{a2} \right) + V_H * t_{r3} + V_H * t_v + t_r * v_0 + t_a * \left(v_0 + a_0 * t_r - \frac{R_{\max} * t_r^2}{2} \right) + \frac{A_H * t_7^2}{2} + \frac{A_H * t_{a2}^2}{2} + \frac{A_{LOH} * t_a^2}{2} + \frac{A_{LOH} * t_{r2}^2}{2} + t_{r2} * \left(v_0 + A_{LOH} * t_a + a_0 * t_r - \frac{R_{\max} * t_r^2}{2} \right) - \frac{R_{\max} * t_7^3}{6} - \frac{R_{\max} * t_r^3}{6} + \frac{R_{\max} * t_{r2}^3}{6} + \frac{R_{\max} * t_{r3}^3}{6} + \frac{a_0 * t_r^2}{2} \quad (86)$$

6.2 Odvození časů pro zápornou dráhu

Jako v předchozím odvození bylo nutné zjistit vztahy pro výpočet jednotlivých intervalů, což je nutné udělat i pro druhý základní tvar. Použité postupy a principy byly obdobné jako v kapitole Odvození časů. Výpočet dob trvání jednotlivých intervalů je zásadní pro vytvoření programu, který dokáže generovat trajektorii pro servopohon.

Čas t_r udává, za jakou dobu dokáže trajektorie zrychlení z počáteční hodnoty a_0 dosáhnout svého minima A_{LOH} . Tento vztah lze odvodit z profilu zrychlení pro první interval (rovnice (60)).

$$a(t_1 = t_r) = A_{LOH} \rightarrow t_r = - \frac{A_{LOH} - a_0}{R_{\max}} \quad (87)$$

Čas t_{r2} odpovídá době potřebné ke zrychlení z minima A_{LOH} na nulovou hodnotu zrychlení. Rovnice popisující zrychlení ve třetím intervalu (rovnice (68)) byla položena rovna nule a byl vyjádřen vztah pro t_{r2}

$$a(t_3 = t_{r2}) = 0 \rightarrow \boxed{t_{r2} = -\frac{A_{LOH}}{R_{max}}} \quad (88)$$

Doba trvání druhého intervalu vyjadřuje potřebnou dobu k dosažení maximální rychlosti V_H . Samotná kombinace prvního a třetího intervalu by nemusela být dostačující pro dosažení. Proto byla rovnice rychlosti ve třetím intervalu (rovnice (69)) položena nule a byl z ní vyjádřen vztah pro výpočet času t_a . Do tohoto vztahu byly vloženy rovnice popisující časy t_r a t_{r2} (rovnice (87) a (88)).

$$v(t_3 = t_{r2}) = V_H \rightarrow \boxed{t_a = \frac{2 * A_{LOH}^2 - a_0^2 + 2 * R_{max} * V_H - 2 * R_{max} * v_0}{2 * A_{LOH} * R_{max}}} \quad (89)$$

Obdobně bylo postupováno i při odvozování následujících vztahů. Čas t_{r3} udává potřebnou dobu ke zrychlení z nuly na hodnotu A_H .

$$a(t_5 = t_{r3}) = A_H \rightarrow \boxed{t_{r3} = \frac{A_H}{R_{max}}} \quad (90)$$

Pro zjištění vztahu pro čas t_{r4} lze postupovat velice podobně jako v prvním intervalu. Tento čas udává, za jakou dobu klesne zrychlení ze svého maxima na konečnou hodnotu a_{end} .

$$a(t_7 = t_{r4}) = a_{end} \rightarrow \boxed{t_{r4} = \frac{A_H - a_{end}}{R_{max}}} \quad (91)$$

Předposledním vztahem, který bylo třeba odvodit, byl vztah pro čas šestého intervalu. Tento interval má za cíl pomoci se zvýšením rychlosti na koncovou podmínku v_{end} . Opět byly do odvozeného vztahu vloženy vzorce pro ostatní časy.

$$v(t_7 = t_{r4}) = v_{end} \rightarrow$$

$$t_{a2} = -\frac{2 * A_H^2 - a_{end}^2 + 2 * R_{max} * V_H - 2 * R_{max} * v_{end}}{2 * A_H * R_{max}} \quad (92)$$

Posledním časem, který chyběl odvodit, byl čas t_v . Tento čas je dobou trvání čtvrtého intervalu. Vztah pro tento čas byl odvozen z rovnice celkové dráhy (rovnice (86)), ta se rovná žádané dráze s_{req} . Opět byly dosazeny vztahy jednotlivých intervalů, tedy finální vztah byl vyjádřen v závislosti na maximálních hodnotách, počátečních a koncových podmínkách.

$$s(t_7 = t_{r4}) = s_{req} \rightarrow$$

$$t_v = \left(3 * A_H * a_0^4 - 3 * A_{LOH} * a_{end}^4 + 6 * A_H * A_{LOH}^2 * a_0^2 - 6 * A_H^2 * A_{LOH} * a_{end}^2 - 12 * A_H * R_{max}^2 * V_H^2 + 12 * A_{LOH} * R_{max}^2 * V_H^2 + 12 * A_H * R_{max}^2 * v_0^2 - 12 * A_{LOH} * R_{max}^2 * v_{end}^2 - 8 * A_H * A_{LOH} * a_0^3 + 8 * A_H * A_{LOH} * a_{end}^3 - 24 * A_H * A_{LOH} * R_{max}^2 * s_0 + 24 * A_H * A_{LOH} * R_{max}^2 * s_{req} + 12 * A_H * A_{LOH}^2 * R_{max} * v_0 - 12 * A_H^2 * A_{LOH} * R_{max} * v_{end} + 12 * A_H * R_{max} * a_0^2 * v_0 - 12 * A_{LOH} * R_{max} * a_{end}^2 * v_{end} + 12 * A_H * A_{LOH}^2 * R_{max} * V_H - 12 * A_H^2 * A_{LOH} * R_{max} * V_H - 24 * A_H * A_{LOH} * R_{max} * a_0 * v_0 + 24 * A_H * A_{LOH} * R_{max} * a_{end} * v_{end} \right) / (24 * A_H * A_{LOH} * R_{max}^2 * V_H) \quad (93)$$

6.3 Kompenzace limitů v záporné dráze

S takto upravenými vztahy pro jednotlivé intervaly, bylo dalším krokem odvození vztahů pro výpočet dosažitelných maxim, tedy hodnoty V_H , A_H a A_{LOH} . Postup byl zvolen stejný jako v případě vztahů (43),(46) a (49).

Minimální hodnota zrychlení, bez využití povolených maxim, byla dosažena mezi prvním a druhým intervalem, tedy čas t_a musel být nulový. Z toho lze vyjádřit rovnici (94) a její řešení (95):

$$t_a = 0 \rightarrow \frac{2 * A_{LOH}^2 - a_0^2 + 2 * R_{max} * V_H - 2 * R_{max} * v_0}{2 * A_{LOH} * R_{max}} = 0 \quad (94)$$

$$A_{LOH1} = -\frac{\frac{1}{2^2} * \sqrt{a_0^2 + 2 * R_{max} * V_H + 2 * R_{max} * v_0}}{2} \quad (95)$$

Řešením rovnice jsou dva výsledky navzájem opačné. Bylo použité takové řešení, které dokáže generovat záporné hodnoty.

Maximální využitá hodnota zrychlení, bez dosažení limitů, by byla dosažena mezi pátým a sedmým intervalem. Tedy doba šestého intervalu by musela být nulová. Z toho předpokladu lze odvodit rovnici takovou, že čas t_{a2} bude rovný nule.

$$t_{a2} = 0 \rightarrow -\frac{2 * A_H^2 - a_{end}^2 + 2 * R_{max} * V_H - 2 * R_{max} * v_{end}}{2 * A_H * R_{max}} = 0 \quad (96)$$

$$A_{H1} = \frac{\frac{1}{2^2} * \sqrt{a_{end}^2 + 2 * R_{max} * V_H + 2 * R_{max} * v_{end}}}{2} \quad (97)$$

Výsledkem rovnice jsou opět dva kořeny navzájem opačné. Byl použit takový kořen, který dokáže generovat kladné hodnoty zrychlení.

Maximální hodnota rychlosti by byla dosažena v průběhu zrychlení, v bodě kdy zrychlení dosahuje nulové hodnoty, tedy mezi třetím a pátým intervalem. Stejně jako v případě rovnice (48), i pro druhý základní tvar profilu, byl čas t_v (rovnice (93)) položen rovný nule.

$$t_v(V_{max} = V_H) = 0 \quad (98)$$

$$V_{H1} = -\left(3 * A_H^2 * A_{LOH} - 3 * A_H * A_{LOH}^2 + 3^{\frac{1}{2}} * ((A_H * A_{LOH}) * (3 * A_H * a_0^4 - 3 * A_{LOH} * a_{end}^4 - 3 * A_H^2 * A_{LOH}^3 + 3 * A_H^3 * A_{LOH}^2 + 6 * A_H * A_{LOH}^2 * a_0^2 - 6 * A_H^2 * A_{LOH} * a_{end}^2 + 12 * A_H * R_{max}^2 * v_0^2 - 12 * A_{LOH} * R_{max}^2 * v_{end}^2 - 8 * A_H * A_{LOH} * a_0^3 + 8 * A_H * A_{LOH} * a_{end}^3 - 24 * A_H * A_{LOH} * R_{max}^2 * s_0 + 24 * A_H * A_{LOH} * R_{max}^2 * s_{req} + 12 * A_H * A_{LOH}^2 * R_{max} * v_0 - 12 * A_H^2 * A_{LOH} * R_{max} * v_{end} + 12 * A_H * R_{max} * a_0^2 * v_0 - 12 * A_{LOH} * R_{max} * a_{end}^2 * v_{end} - 24 * A_H * A_{LOH} * R_{max} * a_0 * v_0 + 24 * A_H * A_{LOH} * R_{max} * a_{end} * v_{end})\right)^{\frac{1}{2}} / (6 * R_{max} * (A_H - A_{LOH})) \quad (99)$$

Opět bylo řešením rovnice dvojice opačných kořenů. Bylo zvoleno takové řešení, které dokáže vyprodukovat zápornou hodnotu maximální rychlosti.

6.4 Řešení nelinearity

Bylo nutné sestavit výpočetní blok pro druhý základní tvar, který se v principu nelišil od bloku pro první tvar. Rozdílné jsou pouze funkce pro výpočet jednotlivých maxim. Opět bylo třeba vypočítávat hodnoty v cyklu, aktuální hodnota se vypočítávala z předchozích hodnot. Tento cyklus se ukončí v momentě, kdy už nedocházelo ke změnám, tedy aktuální hodnota je rovná minulé. Toto lze matematicky zapsat jako:

$$A_{LOH}(k) = f(V_H(k-1), a_0, v_0, R_{max}) \quad (100)$$

$$A_H(k) = f(V_H(k-1), a_{end}, v_{end}, R_{max}) \quad (101)$$

$$V_H(k) = f(A_{LOH}(k-1), A_H(k-1), a_0, v_0, s_0, a_{end}, v_{end}, s_{req}, R_{max}) \quad (102)$$

Pro urychlení výpočtu bylo nutné sestavit startovací metodu. Ta využívá k prvnímu výpočtu maximální povolené meze.

$$A_{LOH} = f(V_{max}, a_0, v_0, R_{max}) \quad (103)$$

$$A_H = f(V_{max}, a_{end}, v_{end}, R_{max}) \quad (104)$$

$$V_H = f(A_{min}, A_{max}, a_0, v_0, s_0, a_{end}, v_{end}, s_{req}, R_{max}) \quad (105)$$

Při každém iteračním výpočtu byla provedena kontrola, zda vypočtená hodnota nepřesahuje povolené maximum. V případech, že tato hodnota byla překročena, došlo k přepsání za maximální povolenou hodnotu a s tou bylo nadále počítáno. Pro správnou implementaci toho bloku do skriptu bylo třeba vyřešit přepínání mezi jednotlivými variantami. Nalézt vztah pro určení, která varianta se má použít, už z počátečních a koncových podmínek se nalézt nepodařilo. Přepnutí mezi variantami bylo provedeno za pomoci kontroly správnosti vypočtených dosažených maxim. Pokud maxima nevyšly v plánovaném intervalu nebo obsahovaly imaginární složku, byla použita druhá varianta profilu. Byla tedy přidána podmínka ke konci iteračního cyklu, která kontrolovala hodnoty vypočítaných maxim a pokud nějaká obsahovala chybu, došlo k přerušení této

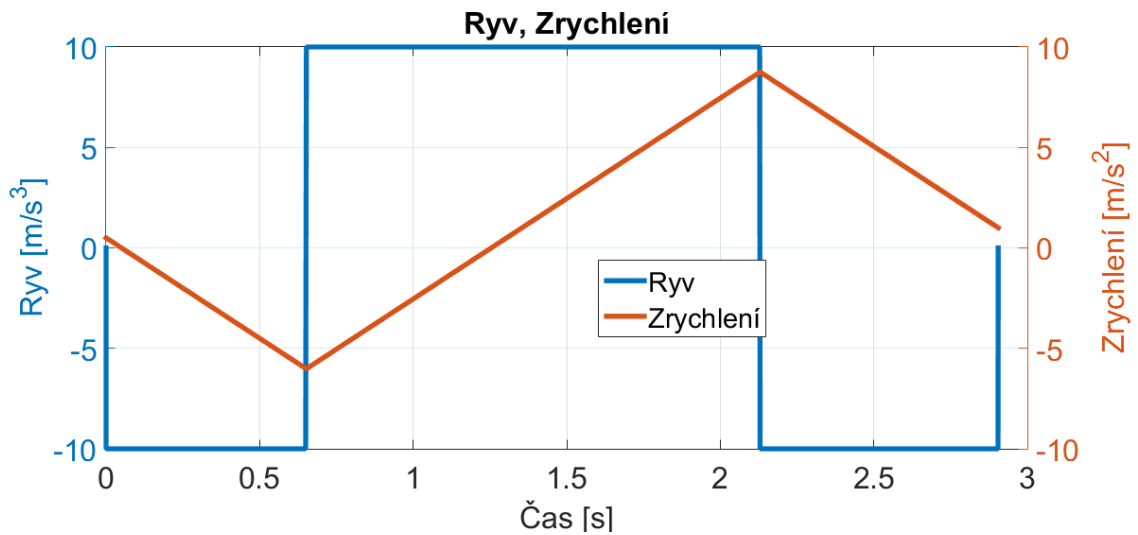
varianty a následoval výpočet od začátku, ale za použití druhé varianty. S takto upraveným skriptem je nyní možné odsimulovat profil i se zápornou dráhou.

```

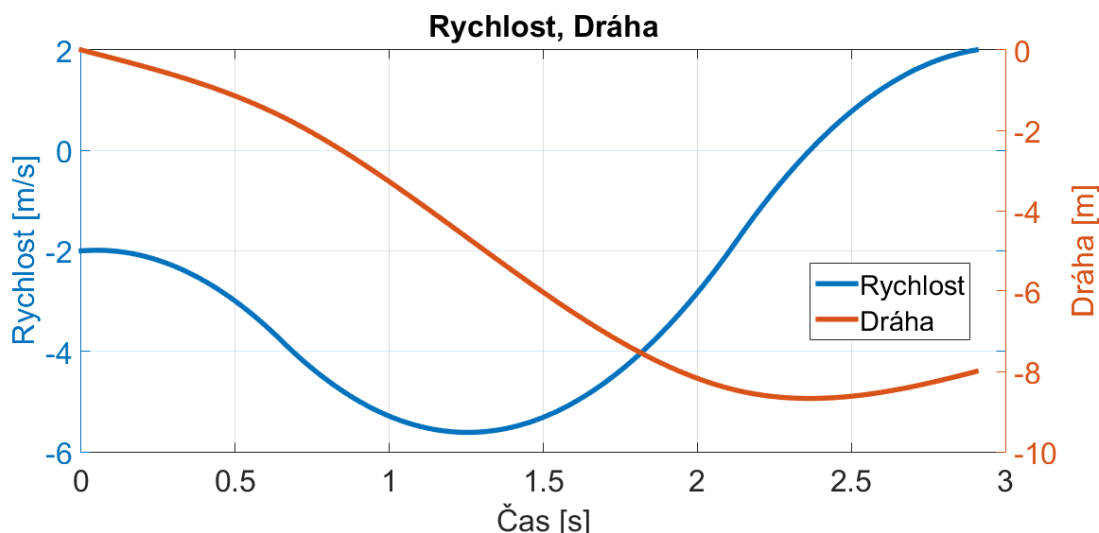
*****
Ryv, Zrychlení, Rychlost, Dráha
-----
R_class [m/s3] : 10.00
-----
A_max [m/s2] : 10.00
A_H [m/s2] : 8.75
A_LO_H [m/s2] : -6.02
A_0 [m/s2] : 0.50
A_end [m/s2] : 1.00
-----
V_max [m/s] : 10.00
V_H [m/s] : 5.61
V_0 [m/s] : -2.00
V_end [m/s] : 2.00
-----
S_req [m/s] : -8.00
s_0 [m/s] : 0.00
*****
Casove konstanty:
*****
tsum [s] : 2.905
tr [s] : 0.652
tr2 [s] : 0.602
tr3 [s] : 0.875
tr4 [s] : 0.775
ta [s] : 0.000
ta2 [s] : -0.000
tv [s] : 0.000

```

Kód 6 – Kontrolní příklad 3, záporná dráha



Graf 9 – Trajektorie ryvu a zrychlení pro kontrolní příklad 3



Graf 10 – Trajektorie rychlosti a dráhy pro kontrolní příklad 3

7 Omezení maxim

Použitá kombinace výpočetního cyklu spolu se startovací metodou obsahovala nedostatek, který se vyskytoval jen při určitých kombinacích počáteční a koncové rychlosti. Výsledkem tohoto nedostatku byla imaginární hodnota na A_H nebo A_{LOH} , v jakém zrychlení bylo dáno poměrem v_0 a v_{end} . Startovací metoda způsobí, že vypočítaná hodnota V_H vyjde menší než je hodnota v_0 nebo v_{end} . To způsobí, že výsledek pod odmocninou, ve vzorcích pro výpočet A_H a A_{LOH} (rovnice (43), (46), (95) a (97)), má zápornou hodnotu, tedy výsledná hodnota odmocniny je imaginární. Například:

```

*****
Ryv, Zrychlení, Rychlost, Dráha
-----
R_class [m/s3]: 15.00
-----
A_max [m/s2]: 6.00
A_0 [m/s2]: 0.00
A_end [m/s2]: 0.00
-----
V_max [m/s]: 10.00
V_0 [m/s]: 2.00
V_end [m/s]: 2.00
-----
s_req [m/s]: 0.50
s_0 [m/s]: 0.00
*****
Startovací podmínka:
A_H = 6
A_LO_H = -6
V_H = 0.7079

Hodnoty po první iteraci:
A_H = 0.0000 + 4.4025i
A_LO_H = 0.0000 - 4.4025i
V_H = -2.0317 - 0.4110i

```

Kód 7 – Špatně vypočítané hodnoty

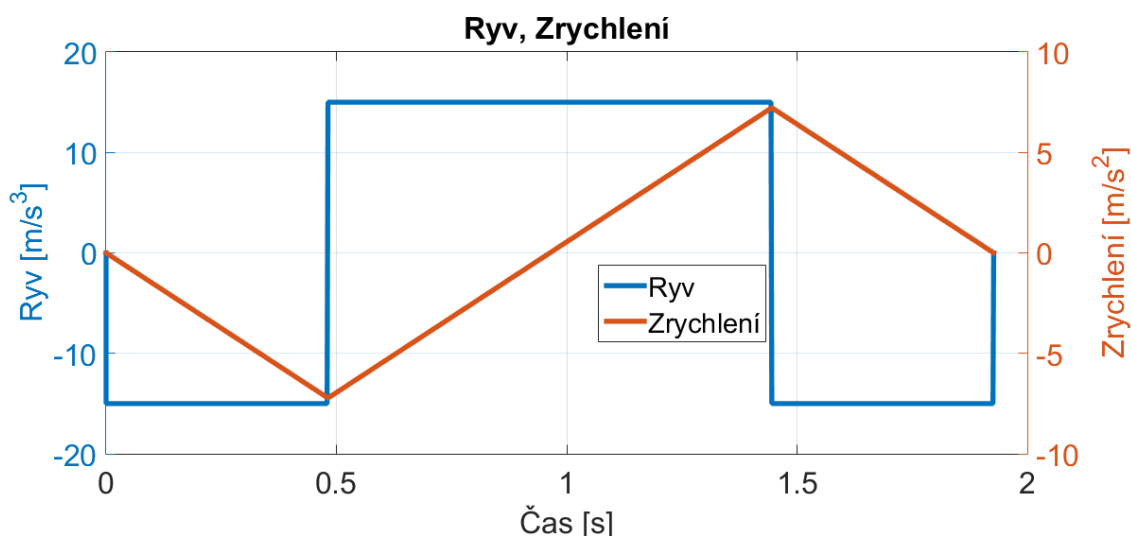
Znázorněná hodnota startovací metody způsobuje, že hodnota pod odmocninou ve vztazích pro A_H a A_{LOH} (rovnice (42) a (45)) má zápornou hodnotu a díky tomu má výsledek imaginární složku. Toto způsobí ukončení iterační metody a zvolí se druhý základní profil. To ale v tomto případě způsobí zdržení, neboť lze v tomto případě určit, že by první varianta byla dostačující.

```

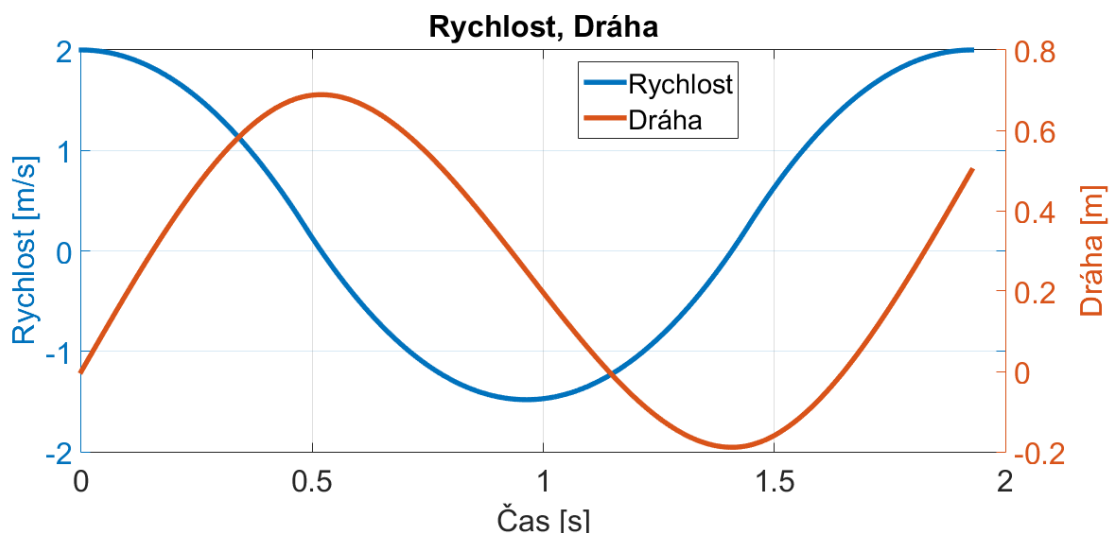
*****
Ryv, Zrychlení, Rychlost, Dráha
-----
R_class [m/s3]: 15.00
-----
A_max [m/s2]: 6.00
A_H [m/s2]: 6.00
A_LO_H [m/s2]: -6.00
A_0 [m/s2]: 0.00
A_end [m/s2]: 0.00
-----
V_max [m/s]: 10.00
V_H [m/s]: 1.49
V_0 [m/s]: 2.00
V_end [m/s]: 2.00
-----
s_req [m/s]: 0.50
s_0 [m/s]: 0.00
*****
Casove konstanty:
*****
tsum [s]: 1.964
tr [s]: 0.400
tr2 [s]: 0.400
tr3 [s]: 0.400
tr4 [s]: 0.400
ta [s]: 0.182
ta2 [s]: 0.182
tv [s]: -0.000

```

Kód 8 – Kontrolní příklad 4



Graf 11 – Trajektorie ryvu a zrychlení pro kontrolní příklad 4



Graf 12 – Trajektorie rychlosti a dráhy pro kontrolní příklad 4

Hodnota V_H v kladné verzi profily musí vyjít větší než v_0 a v_{end} , aby nedocházelo k přerušení iterace. Byl zvolen postup, kdy se polynom pod odmocninou položil rovný jedné desetinně, za proměnou V_H byla vložena startovací metoda (rovnice (58)) a byl odvozen vztah pro výpočet maxima zrychlení. Tím by bylo zajištěno, že se iterační metoda nachází v regionu konvergence. Následující odvození platí pro variantu kladného profilu a v_0 je větší než v_{end} . Před odvozováním byla dána podmínka, že A_{min} je rovno záporné hodnotě A_{max} .

$$a_0^2 + 2 * R_{max} * f(A_{max}, A_{min}, a_0, v_0, s_0, a_{end}, v_{end}, s_{req}, R_{max}) - 2 * R_{max} * v_0 = 0,1 \rightarrow A_{max} = \dots \quad (106)$$

Výsledek této rovnice je v příloze A.

Pro variantu kde v_{end} dosahuje větší nebo stejné hodnoty jako v_0 platí:

$$a_{end}^2 + 2 * R_{max} * f(A_{max}, A_{min}, a_0, v_0, s_0, a_{end}, v_{end}, s_{req}, R_{max}) - 2 * R_{max} * v_{end} = 0,1 \rightarrow A_{max} = \dots \quad (107)$$

Výsledek této rovnice je v příloze B.

Stejným postupem bylo nutné odvodit vztahy i pro zápornou variantu profilu, zde musí vyjít hodnota V_H menší než v_{end} a v_0 . Do částí polynomu pod odmocninou byla místo proměnné V_H vložena startovací metoda (rovnice (105)). Před odvozováním byla dána

podmínka, že A_{\min} je rovno záporné hodnotě A_{\max} . Následující odvození platí pro v_0 menší než v_{end}

$$a_{\text{end}}^2 + 2 * R_{\text{max}} * f(A_{\text{max}}, A_{\text{min}}, a_0, v_0, s_0, a_{\text{end}}, v_{\text{end}}, s_{\text{req}}, R_{\text{max}}) - 2 * R_{\text{max}} * v_{\text{end}} = 0,1 \rightarrow A_{\text{max}} = \dots \quad (108)$$

Výsledek této rovnice je v příloze C.

Poslední potřebný vztah platí pro variantu kde v_0 je větší nebo rovna v_{end}

$$a_0^2 + 2 * R_{\text{max}} * f(A_{\text{max}}, A_{\text{min}}, a_0, v_0, s_0, a_{\text{end}}, v_{\text{end}}, s_{\text{req}}, R_{\text{max}}) - 2 * R_{\text{max}} * v_0 = 0,1 \rightarrow A_{\text{max}} = \dots \quad (109)$$

Výsledek této rovnice je v příloze D.

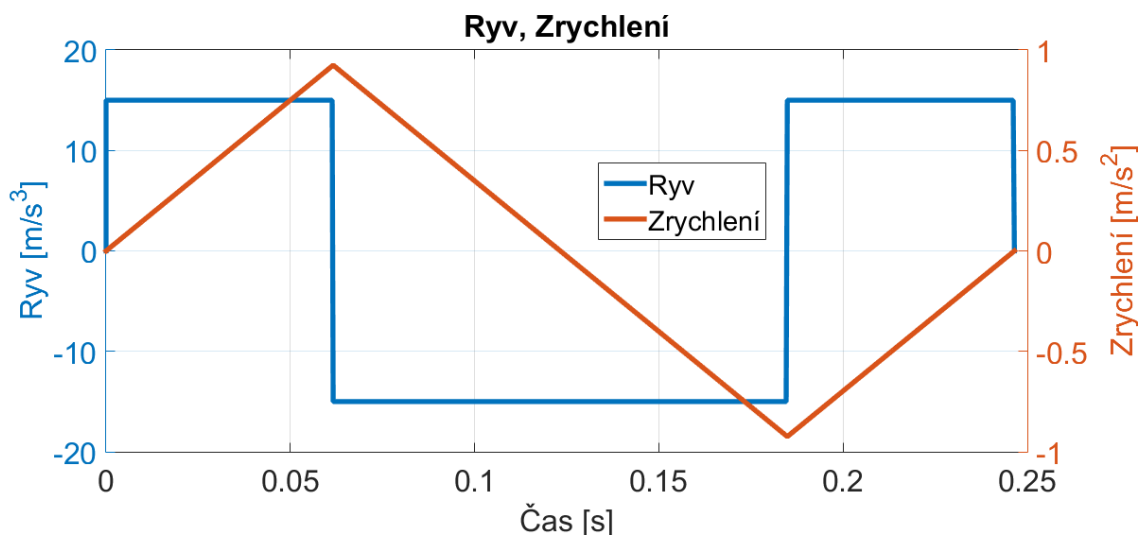
Tyto podmínky byly do skriptu implementovány tak, že jsou spuštěny až po první chybně vypočtené iteraci. Když dojde k jejich spuštění, tak se celý výpočetní blok resetuje a výpočet začne znovu. Pro demonstraci správné funkčnosti byly použity stejné podmínky jako v předchozím příkladu. Nově vygenerovaný profil je daleko rychlejší, časový rozdíl je přes jeden a půl vteřiny.

```

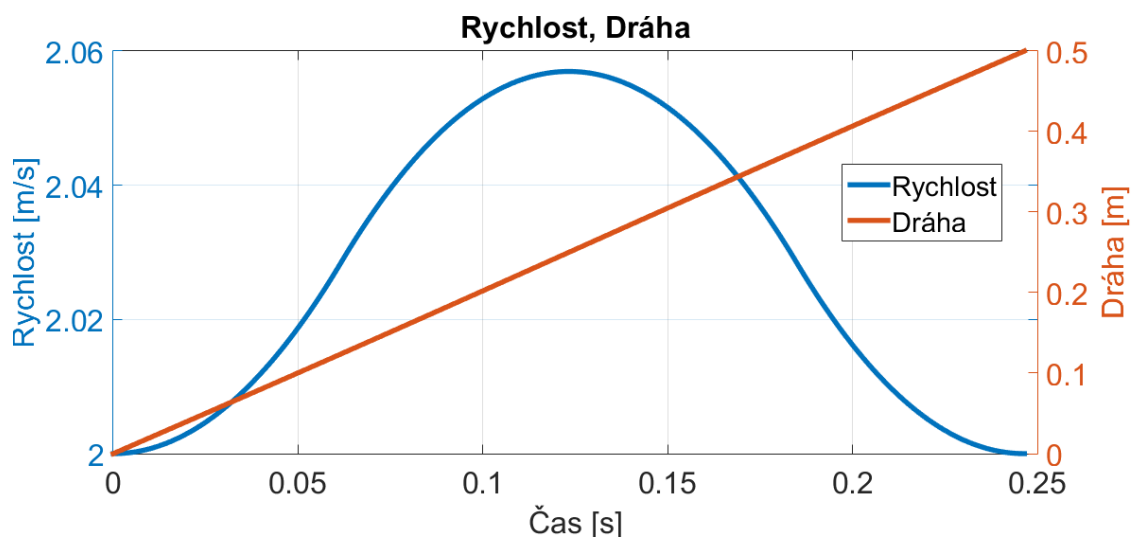
*****
Ryv, Zrychlení, Rychlost, Dráha
-----
R_class [m/s3]: 15.00
-----
A_max [m/s2]: 1.86
A_H [m/s2]: 0.92
A_LO_H [m/s2]: -0.92
A_0 [m/s2]: 0.00
A_end [m/s2]: 0.00
-----
V_max [m/s]: 10.00
V_H [m/s]: 2.06
V_0 [m/s]: 2.00
V_end [m/s]: 2.00
-----
S_req [m/s]: 0.50
s_0 [m/s]: 0.00
*****
Casove konstanty:
*****
tsum [s]: 0.246
tr [s]: 0.062
tr2 [s]: 0.062
tr3 [s]: 0.062
tr4 [s]: 0.062
ta [s]: 0.000
ta2 [s]: 0.000
tv [s]: -0.000

```

Kód 9 – Kontrolní příklad 5



Graf 13 – Trajektorie ryvu a zrychlení pro kontrolní příklad 5



Graf 14 – Trajektorie rychlosti a dráhy pro kontrolní příklad 5

8 Sledování žádané dráhy

Zadavatel diplomové práce požadoval otestování pro případ, že sledovaný objekt se bude pohybovat, tedy že koncové podmínky se budou vyvíjet v čase. Toto lze dobře demonstrovat pro objekt s konstantní rychlostí 1 ms^{-1} . Pokud by aktuální pozice objektu byla 5 metrů, tak by byly naplánovány trajektorie tak, aby se z pozice nula dosáhlo dráhy 5 metrů s konečnou rychlostí 1 ms^{-1} . Tento profil by trval 2 sekundy, ale během profilu by se objekt posunul na pozici 7 metrů, tedy je třeba vytvořit nový profil, který by tuto ztrátu dvou metrů dohnal, ale mezi tím se objekt znovu posune. Takto bude třeba objekt dohánět, až do doby kdy rozdíl bude minimální.

Dalším požadavkem zadavatele byla možnost stanovení doby, po kterou poběží profil. To by znamenalo, že trajektorie může být přerušena a začala by se počítat znovu. Jako počáteční podmínky by byly použity hodnoty zrychlení, rychlosti a dráhy v čase přerušení. Koncová hodnoty by se při každém přerušení aktualizovaly a tím by mělo dojít ke zrychlení dosažení sledovaného bodu. Toto přerušení se týkalo pouze bloku s generováním trajektorie, vždy by bylo nutné provést veškeré výpočty, i když se hodnoty nepoužijí. Bylo tedy nutno generovat profil vždy od počátku do doby přerušení, která byla dána předem. Pro simulační účely bylo nutné vyřešit aktualizaci koncových podmínek, jejich nová hodnota se byla počítána z předchozích, ty se dodaly do rovnice pro rovnoměrně zrychlený pohyb (rovnice (110) a (111)) kde t_p je časem přerušení.

$$s_{req}(k) = s_{req}(k-1) + v_{end}(k-1) * t_p + \frac{1}{2} * a_{end} * t_p^2 \quad (110)$$

$$v_{end}(k) = v_{end}(k-1) + a_{end} * t_p \quad (111)$$

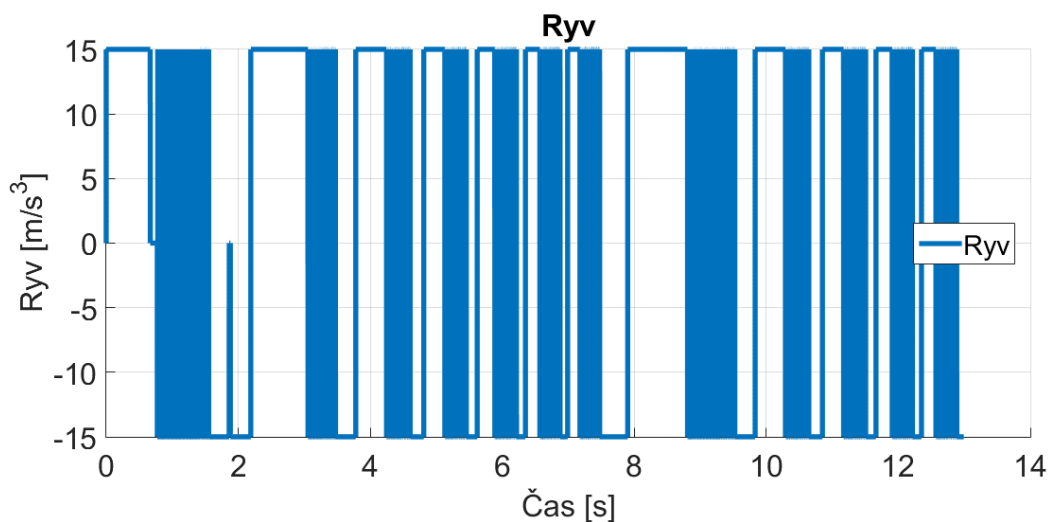
Do simulačního skriptu bylo třeba dodat maximální počet přerušení (počet opakování simulace). Po dosažení počtu opakování dojde k přerušení simulace a vykreslení složených trajektorií. Kdyby simulace nebyla omezena, mohlo by dojít k poškození simulačních dat nebo k celkovému zacyklení simulace. Ukázka funkčnosti v následujícím příkladu.

```

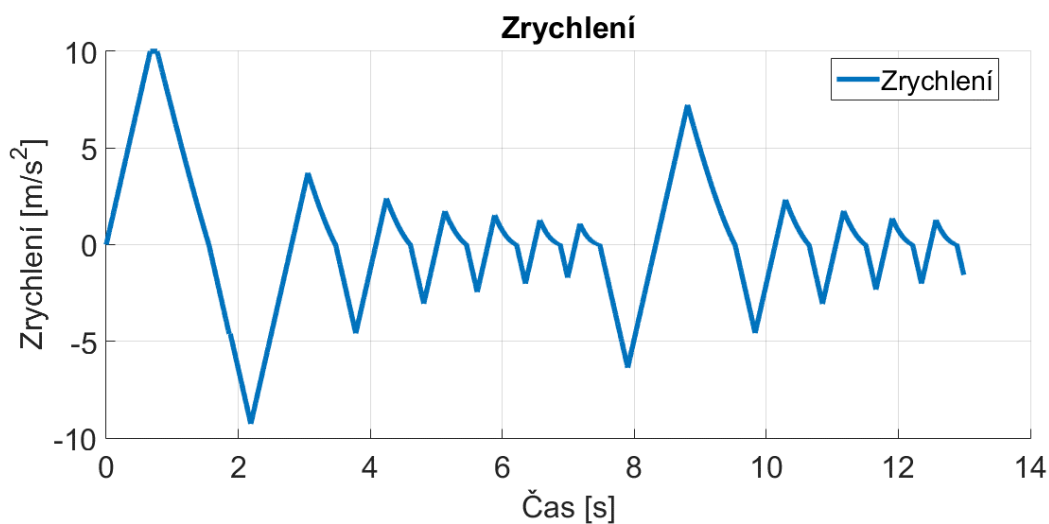
*****
Ryv, Zrychlení, Rychlost, Dráha
-----
R_class [m/s3] : 15.00
-----
A_max [m/s2] : 10.00
A_0 [m/s2] : 0.00
A_end [m/s2] : 0.00
-----
V_max [m/s] : 10.00
V_0 [m/s] : 0.00
V_end [m/s] : 2.00
-----
s_req [m/s] : 10.00
s_0 [m/s] : 0.00
*****
tp [s] : 0.0100
počet iterací : 1200

```

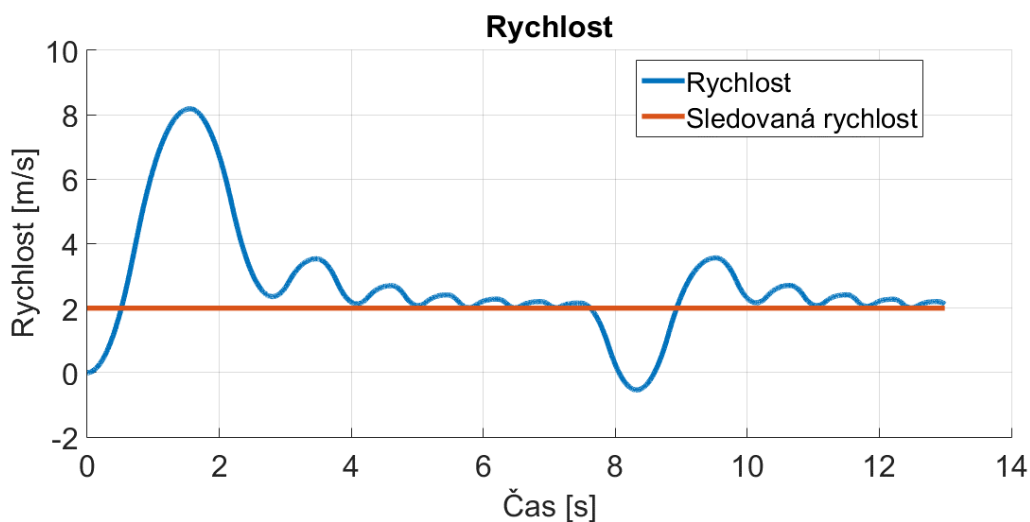
Kód 10 – Vstupní podmínky skriptu pro sledování, příklad 1



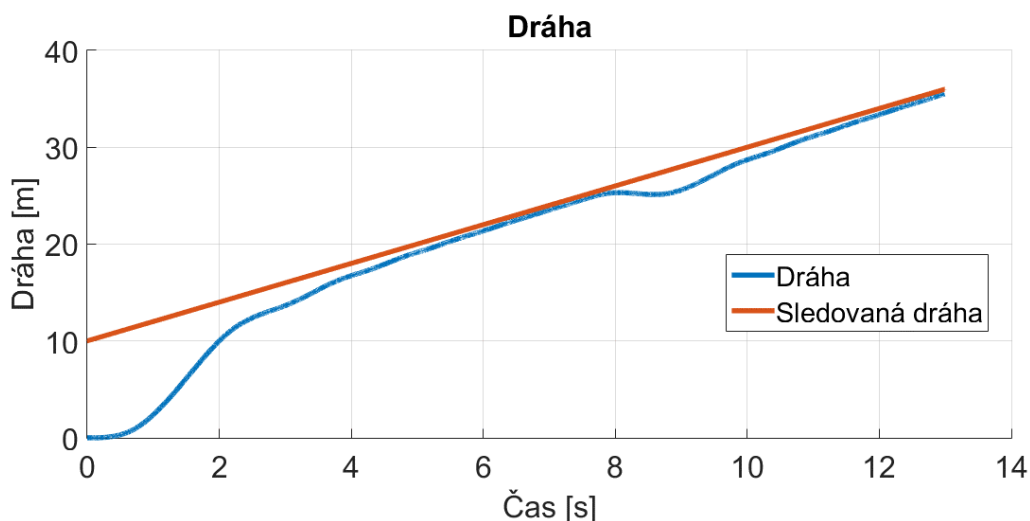
Graf 15 – Trajektorie ryvu pro sledování dráhy, příklad 1



Graf 16 – Trajektorie zrychlení pro sledování dráhy, příklad 1



Graf 17 – Trajektorie rychlosti pro sledování dráhy, příklad 1



Graf 18 – Trajektorie dráhy pro sledování dráhy, příklad 1

Tato metoda sledování žádané dráhy není ideální. Na všech trajektoriích si lze všimnout silného propadu v čase 7,5 s. Tento propad byl způsoben kombinací počátečních a koncových podmínek. Aby bylo možné dosáhnout koncových hodnot profilu bylo nutné zpomalit, protože neexistovala možnost jak propojit počáteční a koncový bod s využitím jednoho z popsaných intervalů. Jedinou možností byl profil, který nutí dráhu vrátit se a pak následně opět dohnat ztrátu. Tento propad nelze v dosavadní metodě vyrušit. Toto a zvlnění všech trajektorií jsou hlavní nevýhody této varianty. Jediná výhoda je její možné využití pro profily, které mají koncová zrychlení proměnné, a tím pádem nelze jakýmkoli způsobem předpovědět možný vývoj trajektorií.

Jako možný ekvivalent přechází varianty byla sestrojena variace, která využívá zmíněné predikce trajektorií, umožňuje přesně odhadnout, kde se bude sledovaný objekt nacházet a jaká bude jeho rychlost a zrychlení. Aby bylo možné předpovědět, kde se bude objekt nacházet, bylo by nutné znát dobu trvání profilu, tedy:

$$t_{sum} = t_r + t_a + t_{r2} + t_v + t_{r3} + t_{a2} + t_{r4} \quad (112)$$

Pak by bylo možné napsat vztah pro konečnou dráhu a rychlost jako:

$$s_{req} = s_{req0} + v_{end0} * t_{sum} + \frac{1}{2} * a_{end} * t_{sum}^2 \quad (113)$$

$$v_{end} = v_{end0} + a_{end} * t_{sum} \quad (114)$$

Tedy konečné hodnoty s_{req} a v_{end} jsou závislé na celkovém čase t_{sum} , ale ten je závislý na koncových hodnotách. Nastal stejný problém jako v případě dosažených maxim a to je nelineární soustava rovnic.

$$t_{sum} = f(a_0, v_0, s_0, a_{end}, v_{end}, s_{req}, A_{max}, A_{min}, V_{max}, R_{max}) \quad (115)$$

$$s_{req} = f(a_{end}, v_{end0}, s_{req0}, t_{sum}) \quad (116)$$

$$v_{end} = f(a_{end}, v_{end0}, t_{sum}) \quad (117)$$

Byla sestrojena iterační metoda, která sloužila právě k vyřešení popsané soustavy. V každém kroku iterace se vždy provedl blok pro výpočet dosažených maxim, ty se následně dosadily do bloku pro výpočet jednotlivých časů. Z celkového součtu všech dob intervalů vyšel čas celkového profilu t_{sum} . Tento čas slouží k odvození nových koncových podmínek, které se v následném kroku iterace použijí jako nové hodnoty pro výpočet nových časů. Jak je každou iterací zpřesňován odhad o konečné poloze a rychlosti, tak je i zpřesňován výsledný čas profilu. Iterační metoda bude zastavena, až se přesnost dostane pod zadanou hodnotu. Čím větší bude kladen důraz na přesnost, tím bude potřeba provést více iteračních cyklů a tím se bude prodlužovat doba výpočtu. Přesnost se v každém iteračním cyklu počítala jako rozdíl žádané dráhy a koncové dráhy profilu. Pokud byl

tento rozdíl menší než hodnotou parametru, tak se další iterační cyklus již neprováděl. V této iterační metodě nebyla použita startovací podmínka. Kvůli tomu byl první rozdíl roven přírůstku dráhy za dobu trvání profilu.

Vztahy pro iteraci lze zapsat následovně:

$$t_{sum}(k) = f(a_0, v_0, s_0, a_{end}, v_{end}(k-1), s_{req}(k-1), A_{max}, A_{min}, V_{max}, R_{max}) \quad (118)$$

$$s_{req}(k) = f(a_{end}, v_{end}(0), s_{req}(0), t_{sum}(k)) \quad (119)$$

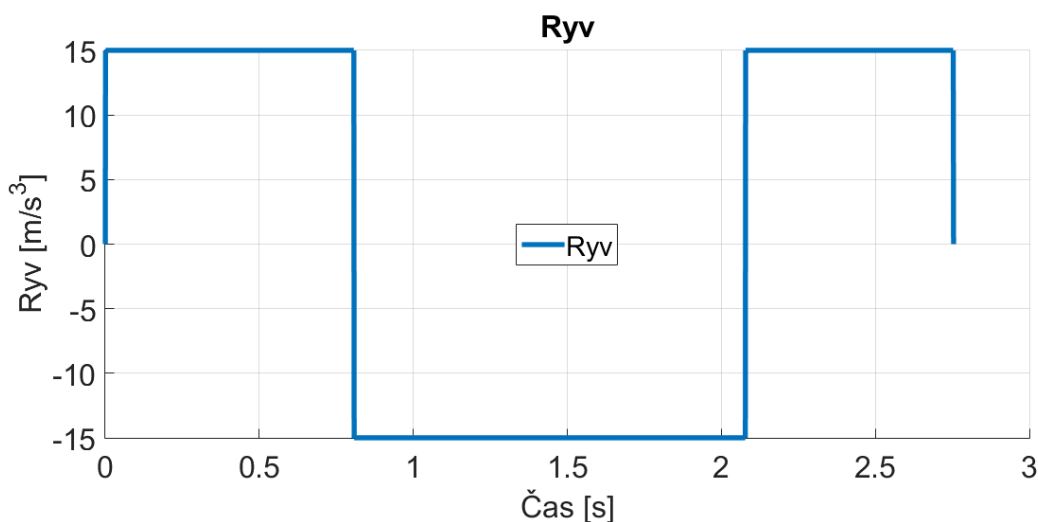
$$v_{end}(k) = f(a_{end}, v_{end}(0), t_{sum}(k)) \quad (120)$$

Podmínku pro ukončení iterace lze zapsat následovně:

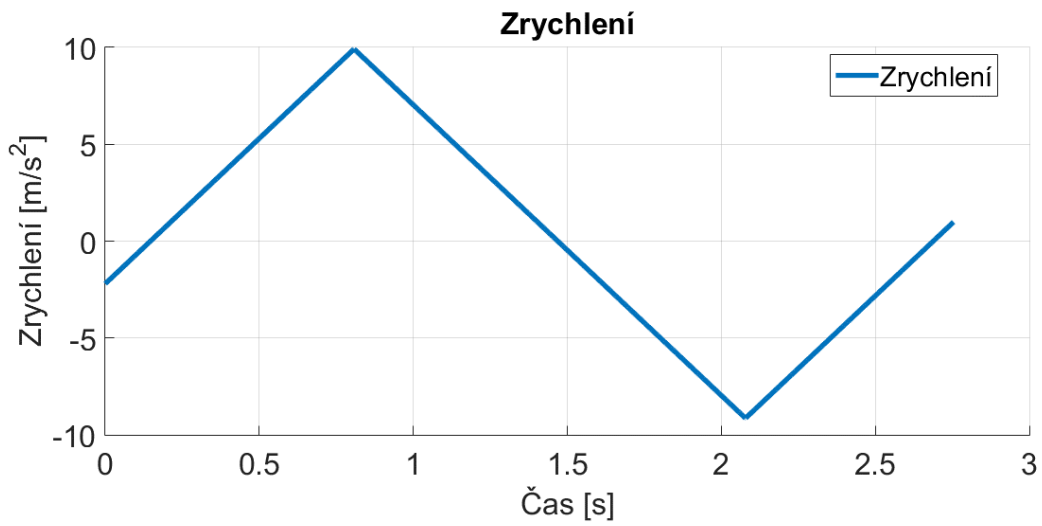
$$presnost > |s_{req} - s(t_7 = t_{r4})| \quad (121)$$

```
*****
Ryv, Zrychlení, Rychlost, Dráha
-----
R_class [m/s3]: 15.00
-----
A_max [m/s2]: 10.00
A_0 [m/s2]: -2.20
A_end [m/s2]: 1.00
-----
V_max [m/s]: 10.00
V_0 [m/s]: 2.00
V_end [m/s]: 0.10
-----
S_req [m/s]: 10.00
s_0 [m/s]: 0.00
*****
presnost=1e-8;
```

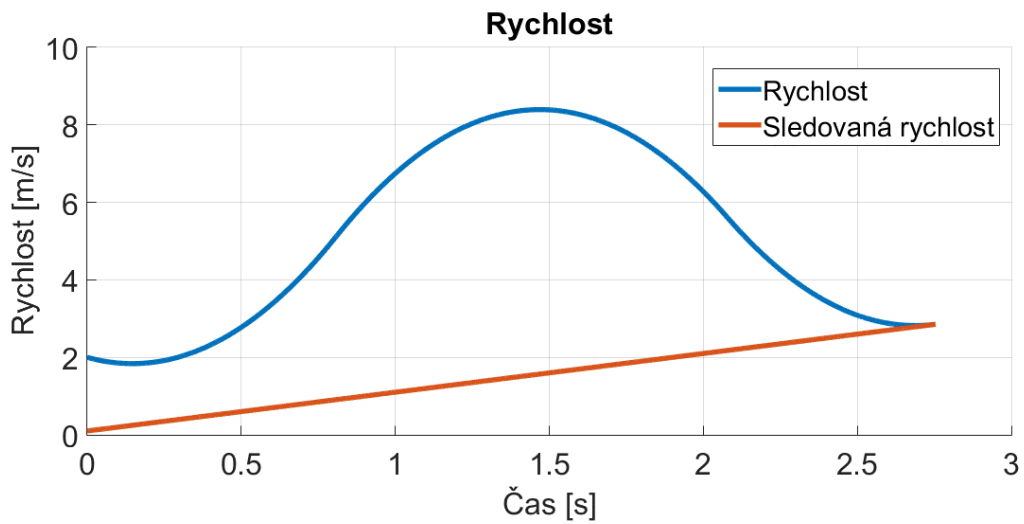
Kód 11 – Vstupní podmínky skriptu pro sledování, příklad 2



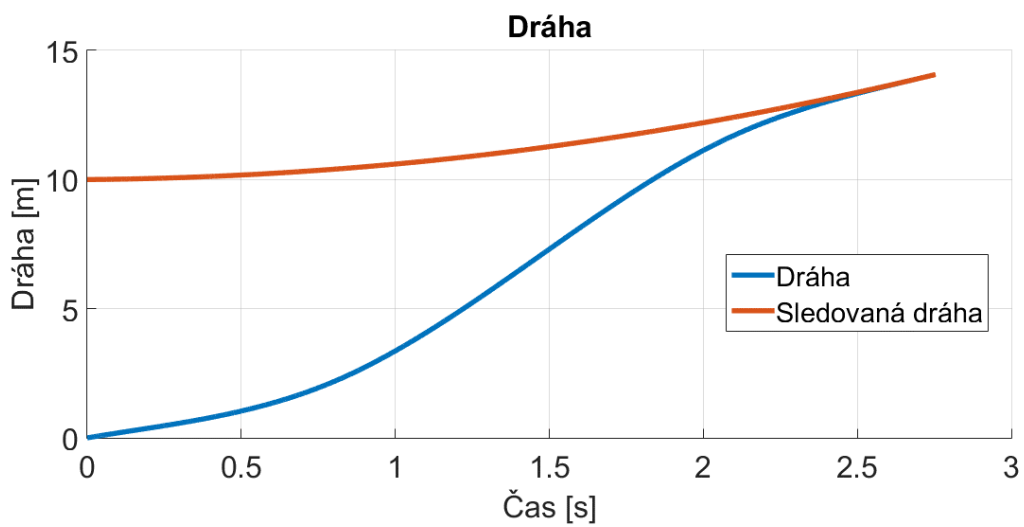
Graf 19 – Trajektorie ryvu pro sledování dráhy, příklad 2



Graf 20 – Trajektorie zrychlení pro sledování dráhy, příklad 2



Graf 21 – Trajektorie rychlosti pro sledování dráhy, příklad 2



Graf 22 – Trajektorie dráhy pro sledování dráhy, příklad 2

Hlavní výhody této varianty jsou přesnost, se kterou dokáže konvergovat ke sledované dráze a také rychlost, kdy využívá nejrychlejší možný profil. Velikou nevýhodou dané varianty je popisovaná predikce, ta omezuje využití pouze na objekty s konstantním zrychlením v čase. Jakákoliv změna zrychlení způsobí obrovskou nepřesnost v odhadu a v následném sledování. Také je daleko náročnější na výpočet než předchozí varianta, to způsobuje zmiňovaný iterační postup.

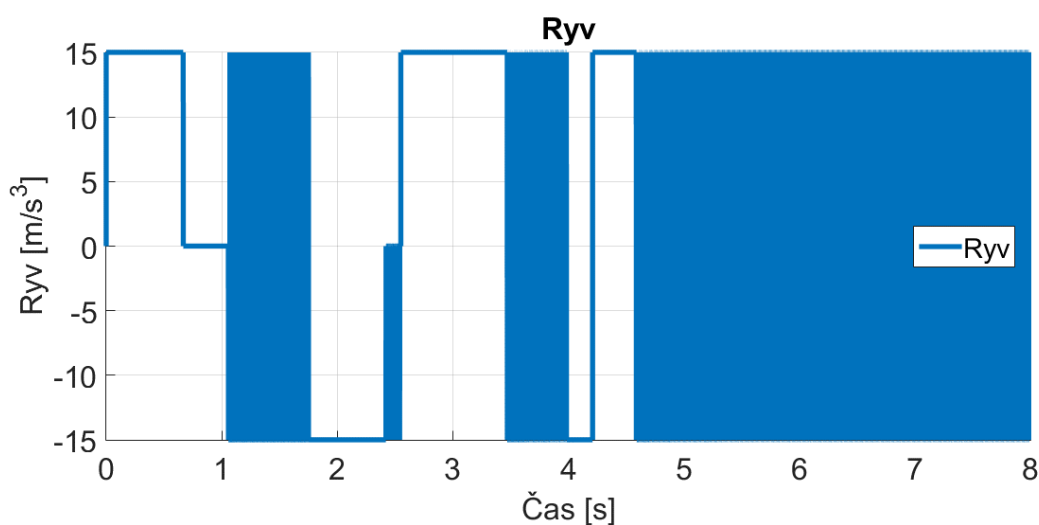
V následujícím příkladu budou použity obě varianty se stejným nastavením maxim, počátečních i koncových podmínek.

```

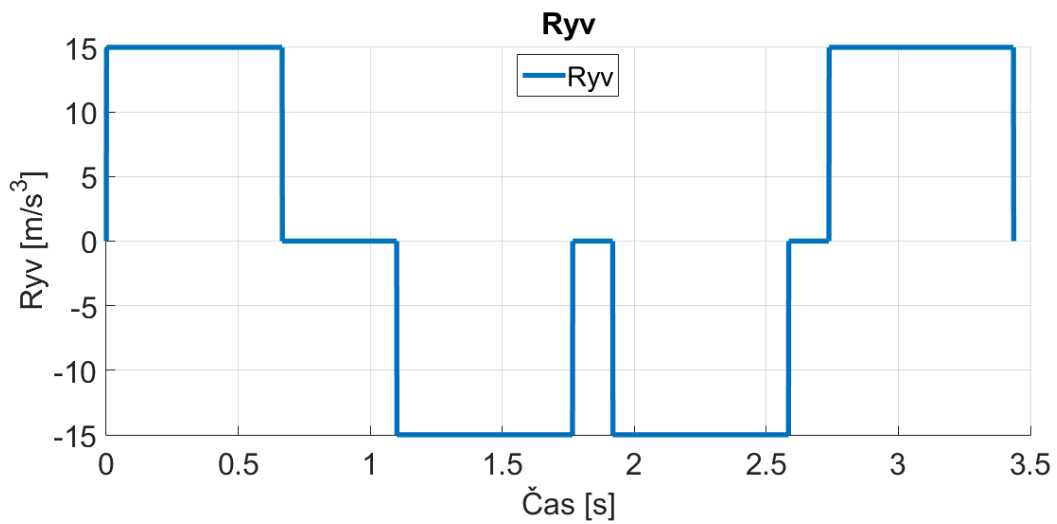
*****
Ryv, Zrychlení, Rychlost, Dráha
-----
R_class [m/s3]: 15.00
-----
A_max [m/s2]: 10.00
A_0 [m/s2]: 0.00
A_end [m/s2]: 0.50
-----
V_max [m/s]: 10.00
V_0 [m/s]: -1.00
V_end [m/s]: 0.10
-----
S_req [m/s]: 10.00
s_0 [m/s]: 0.00
*****
Varianta 1:
tp [s]: 0.0100
počet iterací: 800
Varianta 2:
presnost=1e-8;

```

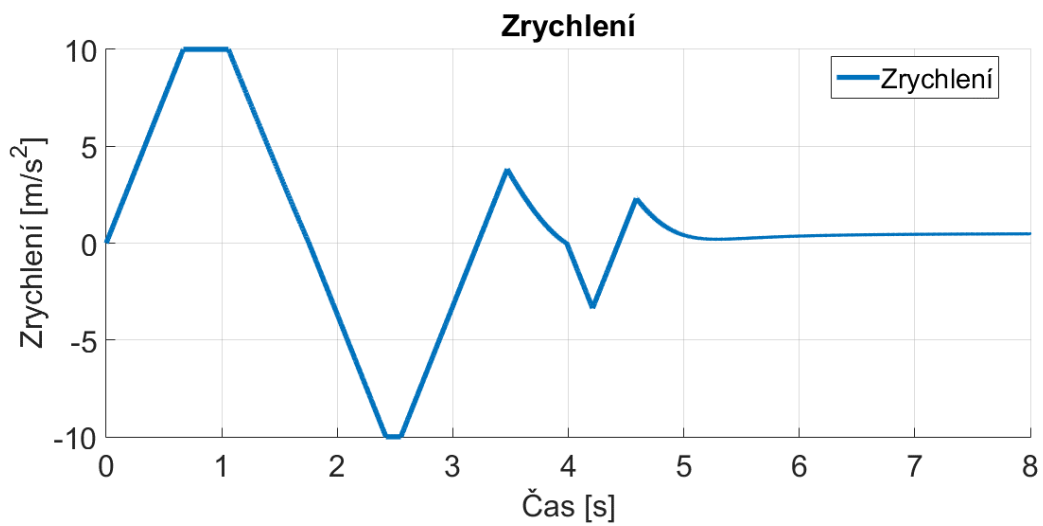
Kód 12 – Vstupní podmínky skriptu pro porovnání variant



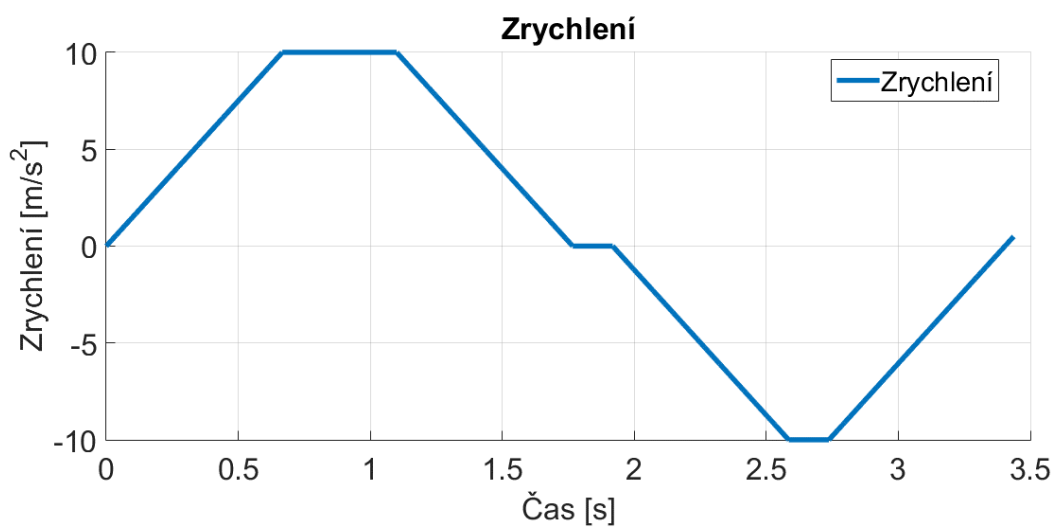
Graf 23 – Trajektorie ryvu pro první variantu



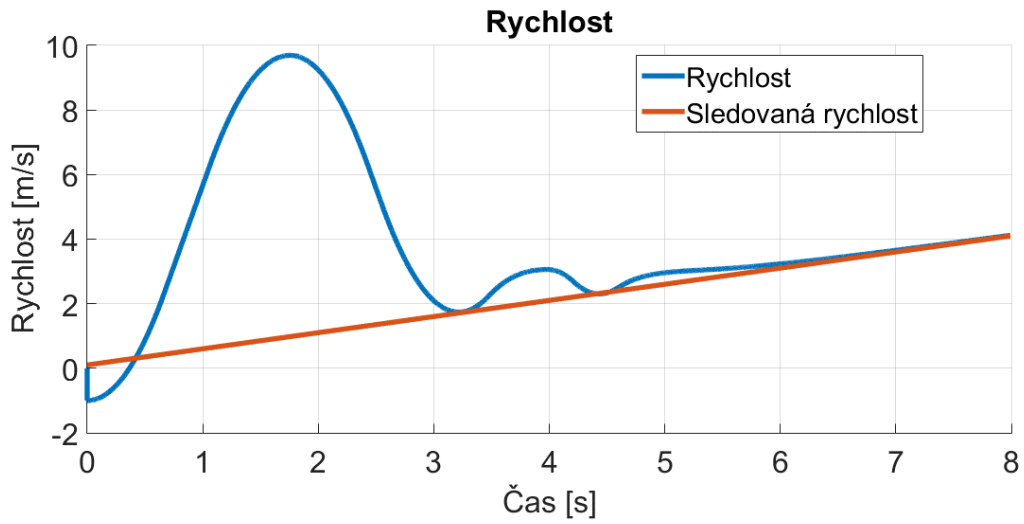
Graf 24 – Trajektorie ryvu pro druhou variantu



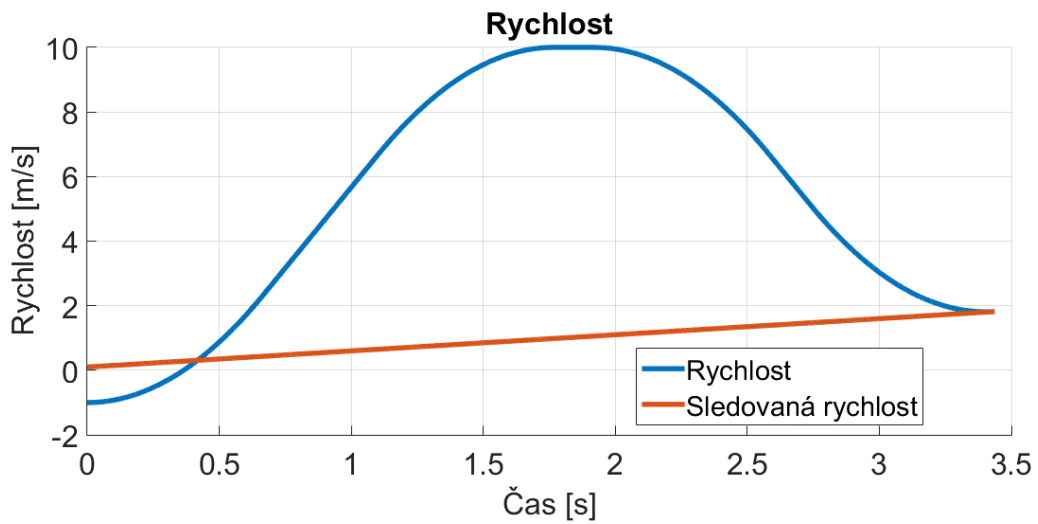
Graf 25 – Trajektorie zrychlení pro první variantu



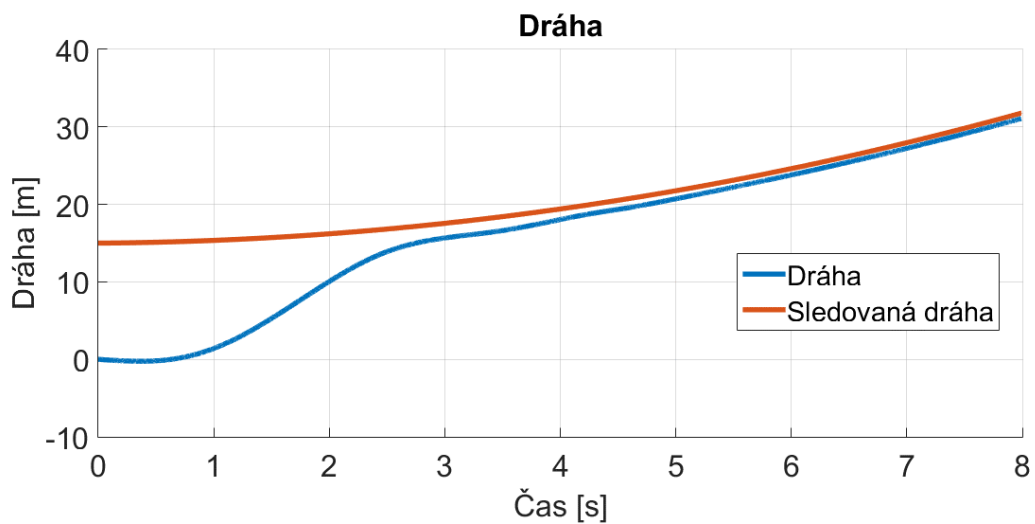
Graf 26 – Trajektorie zrychlení pro druhou variantu



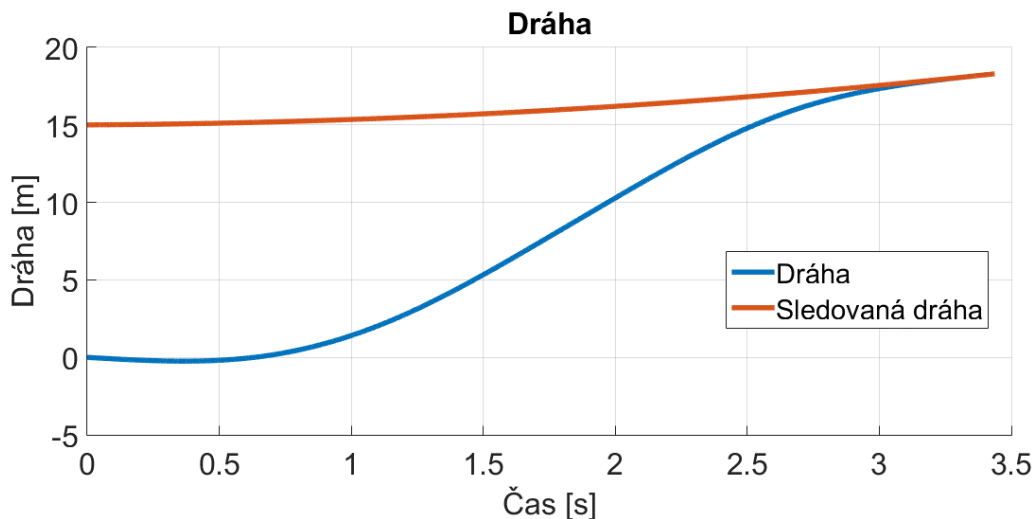
Graf 27 – Trajektorie rychlosti pro první variantu



Graf 28 – Trajektorie rychlosti pro druhou variantu



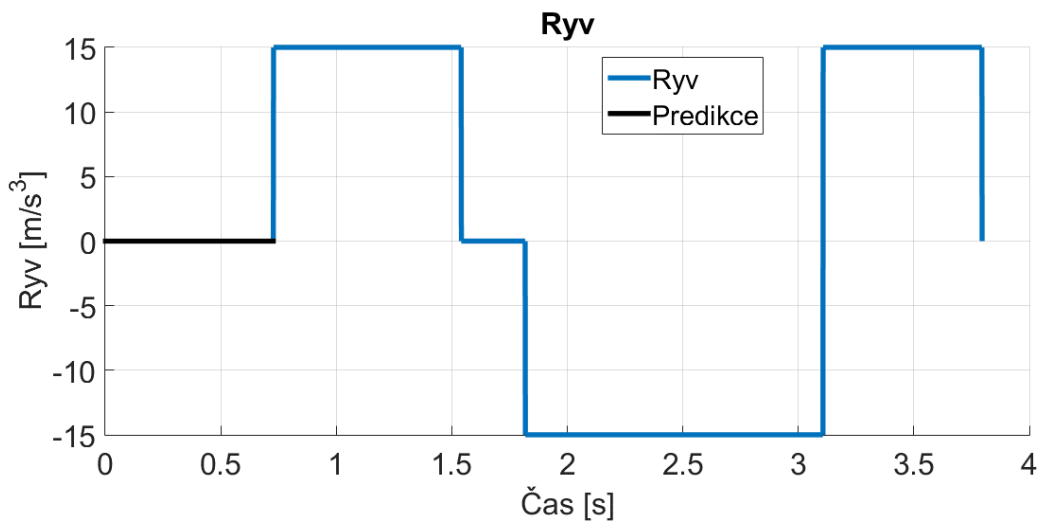
Graf 29 – Trajektorie dráhy pro první variantu



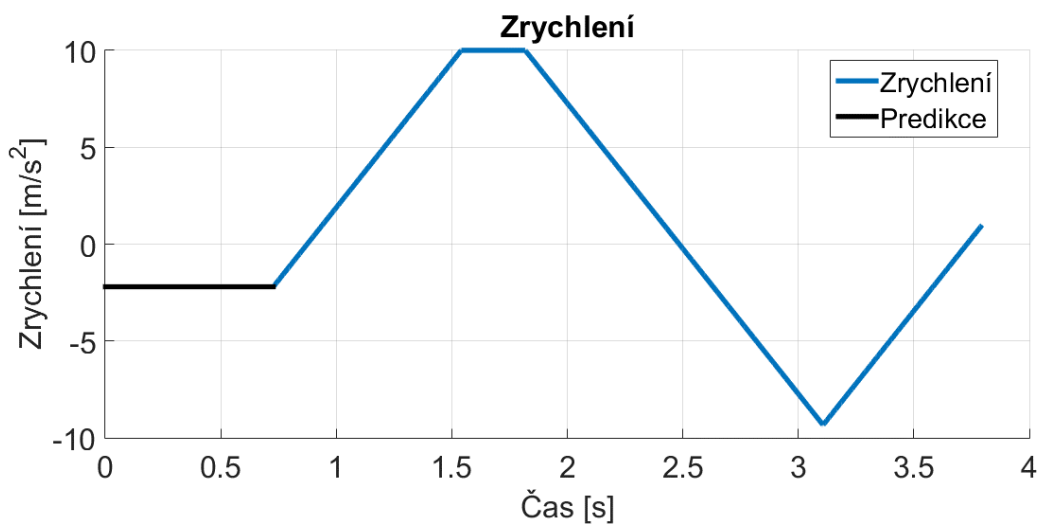
Graf 30 – Trajektorie dráhy pro druhou variantu

Z trajektorií jednotlivých variant je dobře patrné, že pro sledování objektů s konstantním zrychlením je přijatelnější varianta dvě. Ta rychleji konverguje k dráze objektu a umožňuje přesnější pozorování oproti první variantě, té se nikdy nepodaří dosáhnout přesnosti druhé varianty. U první varianty ke konci trajektorie zrychlení by mohlo být milně myšleno, že hodnota je konstantní, ale ona kolem této hodnoty osciluje. Má trojúhelníkový signál s vrcholy v $0,5158 \text{ ms}^{-2}$ a $0,4409 \text{ ms}^{-2}$, tyto časté změny hodnot jsou dobře patrné z trajektorie ryvu pro první variantu. Největším kladem první varianty je možnost periodicky kontrolovat parametry sledovaného objektu, nemusí tedy docházet k odhadům a zrychlení se může jakkoliv měnit.

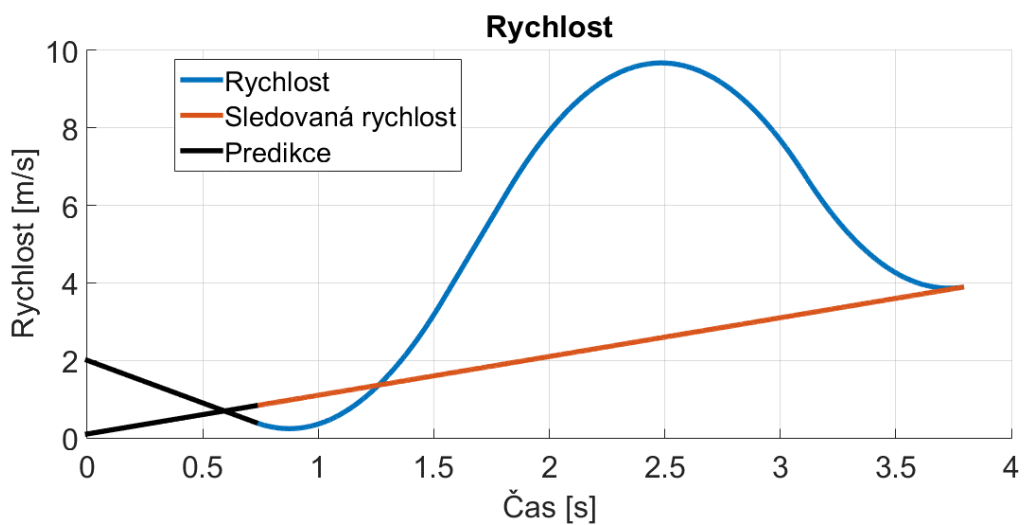
Posledním krokem bylo odsimulování skutečného chování druhé varianty, bylo nutné započítat i čas potřebný k výpočtu predikce, pro tuto činnost byly použity vnitřní časovače programu MATLAB, které počítají čas procesoru. Při každém kroku iterace se čas potřebný pro výpočet zaznamenával a díky tomuto bylo možné vypočítat nové hodnoty pro počáteční i koncové podmínky. Tímto bylo simulováno reálné chování celé úlohy. Nastavení simulace bylo použito stejné jako v příkladu dva v této kapitole (Kód 11). Celkový čas potřebný k vypočítání pouze predikce byl $0,7273 \text{ s}$. Tento čas byl změřen na počítači, potřebný čas pro výpočet predikce v PLC by byl mnohonásobně menší.



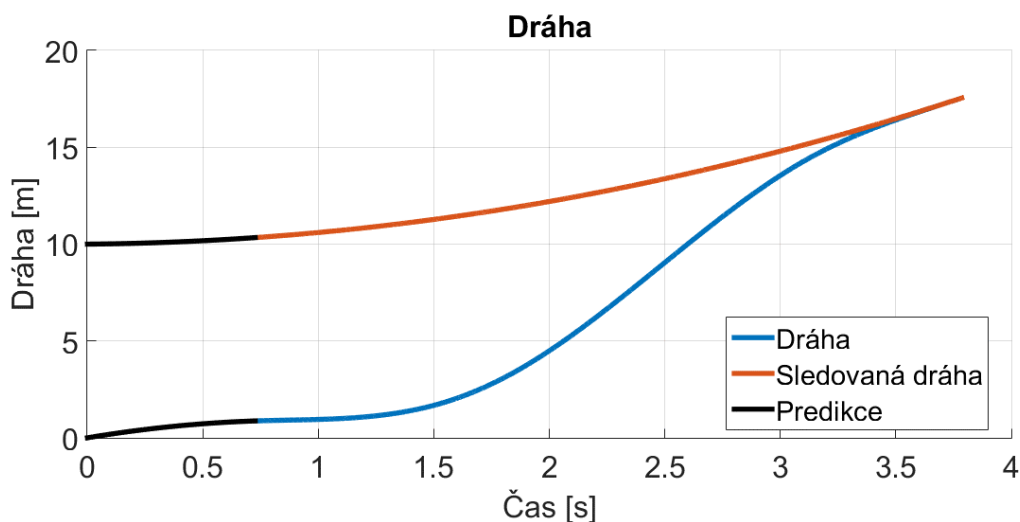
Graf 31 – Trajektorie ryvu pro sledování dráhy, příklad 2 s dobou predikce



Graf 32 – Trajektorie zrychlení pro sledování dráhy, příklad 2 s dobou predikce



Graf 33 – Trajektorie rychlosti pro sledování dráhy, příklad 2 s dobou predikce



Graf 34 – Trajektorie dráhy pro sledování dráhy, příklad 2 s dobou predikce

9 Testování

Pro otestování všech možných variant a kombinací limitů, vstupních a výstupních podmínek byl sestaven skript, který generuje pole náhodných čísel v požadovaných intervalech a následně je cyklicky testuje v připravené funkci. Ta se skládá ze dvou částí simulačního skriptu, výpočetního bloku a generování časových konstant. Ty se v každém cyklu porovnávají, jestli jsou v intervalu od mínus jedné deseti-tisíciny do deseti. Spodní mez tolerance byla definována tak, aby akceptovala možné chyby způsobené zaokrouhlováním. Horní mez byla pouze odhadnuta v závislosti na volbě intervalu pro generování žádané dráhy.

Samotné intervaly, ze kterých byly náhodně vybírány hodnoty pro jednotlivé proměnné, byly zvoleny následovně. Pro žádanou dráhu byl zvolen interval od -20 do 20 metrů. Maximální povolené limity byly zvoleny z intervalu od 10 do 20. Interval pro počáteční a koncové podmínky byl po konzultaci se zadavatelem zvolen tak, aby maximálně dosahoval 10% povolených maxim.

i	Sreq	s0	v0	a0	vend	aend	Vmax	Amax	Rmax
1	12.5889	0.5248	0.9795	-0.4816	-0.2918	-1.3245	12.3225	17.7076	16.7912
2	16.2317	-0.5797	1.5691	-0.7237	1.8215	0.9807	17.3983	16.8580	14.8550
3	-14.9205	1.9880	-1.0296	1.9442	0.8970	-0.0915	18.8899	18.6767	10.3472
4	16.5350	-1.1033	-1.4816	0.8727	0.3236	0.6138	18.5981	15.8051	11.1444
5	5.2944	0.6098	-1.0997	-0.3473	0.1610	1.8663	15.9706	16.2610	10.7663
6	-16.0984	0.4200	-0.5999	-1.6055	0.8218	-0.7479	16.5475	11.1115	16.0782
7	-8.8601	-0.4510	-0.8517	0.9382	-1.9799	-1.6942	19.1501	12.1369	12.3347
8	1.8753	-1.4313	1.7100	0.5492	1.1301	1.1657	14.3318	10.3637	11.4201
9	18.3003	-1.8995	-1.7947	-1.7046	1.7074	-0.5385	12.8976	14.4454	15.3017
10	18.5955	-0.3156	0.3707	-1.5180	-1.9668	0.3404	16.3188	13.2658	19.3030

Kód 13 – Ukázka náhodných parametrů

i	tr	ta	tr2	tv	tr3	ta2	tr4
1.0000	0.6930	-0.0000	0.6643	0.0000	0.7209	-0.0000	0.6420
2.0000	0.7875	-0.0000	0.7388	0.0000	0.7279	-0.0000	0.7939
3.0000	1.0771	0.0000	0.8892	0.0000	0.9794	-0.0000	0.9883
4.0000	0.9266	-0.0000	1.0049	0.0000	0.9199	-0.0000	0.9750
5.0000	0.7312	-0.0000	0.6989	-0.0000	0.6212	0	0.7946
6.0000	0.5912	0.1926	0.6911	-0.0000	0.6911	0.3149	0.7376
7.0000	0.6983	-0.0000	0.6222	-0.0000	0.5497	0.0000	0.6871
8.0000	0.2910	0.0000	0.3391	0	0.4121	0.0000	0.5141
9.0000	1.0554	0.0069	0.9440	0.0000	0.8144	-0.0000	0.7792
10.0000	0.7659	0.2100	0.6872	0.0000	0.6872	0.3820	0.7049

Kód 14 – Ukázka výsledných časů

Byly vytvořeny dvě testovací funkce, první umožnila otestovat funkčnost generování profilů bez změn konečných podmínek, ty se v čase nevyvíjeli, byly statické. Druhá funkce testovala funkčnost varianty skriptu s možností predikce. Celkem bylo provedeno přes 10000 simulací pro každou variantu testovací funkce. Skript se statickými koncovými hodnotami vytvořil chybu v 9 simulacích, tedy chybovat byla pod 0,1%. Varianta s predikcí koncových hodnot vykazovala chybovat v 19 případek z 10000, to odpovídá chybovosti 0,19%. U vzniklých chyb nebyla nalezena žádná spojitost, tedy jejich možnost opravy je minimální.

Závěr

Diplomová práce byla vytvořena na základě žádosti firmy ProjeSoft s požadavky na vytvoření metodiky pro generování bezrázových polohových profilů, které by byly využity pro řízení astronomického teleskopu v reálném provozu. Bezrázovost byla začleněna kvůli podmínce na co nejhladšího průběhu bez skokové změny zrychlení, která způsobují nežádoucí rázy v mechanismu. Během práce byl kladen důraz na splnění všech požadavků zadavatele.

Tato práce byla především zaměřena na seznámení se metodikou odvozování polohových profilů s nenulovými počátečními podmínkami a s omezením hodnot ryvu, tzv. S-curve. Během činnosti na této práci bylo odvozeno velké množství vztahů, které byly použity v devatenácti simulačních skriptech, které byly postupně vytvořeny v prostředí programu MATLAB.

Celkem byly v hlavní části práce odvozeny dva základní profily, pro kladnou a zápornou dráhu. Pro oba profily bylo celkem odvozeno 56 rovnic popisující jednotlivé trajektorie ryvu, zrychlení, rychlosti a dráhy v sedmi časových intervalech. Dále bylo odvozeno 14 vztahů pro výpočet časů jednotlivých intervalů. Ke každému profilu bylo nutné sestavit nelineární soustavu rovnic a iterační metodu pro její vyřešení, díky tomu byl možný výpočet dosažitelných maxim. To dovolilo možnost snížit počet časových intervalů v každém profilu, tím se metodika stala rychlejší a robustnější. Výsledná kombinace obou profilů do jedné funkce byla úspěšně otestována v simulačním prostředí, chybovost v krajním případě dosahovala 0,19%. Bylo otestováno přes deset tisíc možných kombinací.

Podle přání zadavatele byla u vyvinuté metody testována možnost, že se počáteční i koncové podmínky budou vyvíjet v čase. Pro tuto možnost byly vyvinuty dvě variace metody, kde obě mají různé možnosti využití. U druhé varianty dochází k predikci koncových hodnot, díky tomu je výsledný polohový profil této varianty velice přesný a rychlý.

Během této práce nedošlo k implementaci metody do reálného PLC, nicméně reálné možnosti metody byly odsimulovány. Při této simulaci bylo předpokládáno, že během

výpočtu predikce dojde ke změně počátečních i koncových podmínek, tak jakby se stalo v reálné aplikaci. Simulovaná funkce se s i přes tyto podmínka dokázala vygenerovat polohový profil, který k žádané dráze konvergoval s velkou přesností.

Použitá literatura

- [1] J. Lin; K. K. Shyu; C. H. Lin, “Incremental motion control of linear synchronous motor”, Aerospace and Electronic Systems, IEEE Transactions on, Volume 38, Issue 3, July 2002, Page(s):1011 – 1022.
- [2] P. H. Meckl, P. B. Arestides, and M. C. Woods, “Optimized scurve motion profiles for minimum residual vibration”, Proc. Amer. Contr. Conf., PA, June 1998, pp. 2627–2631.
- [3] D. M. Tsay, C. F. Lin, “Asymmetrical inputs for minimizing residual response”, IEEE International Conference on Mechatronics 2005, Taipei, Taiwan, July 2005, pp. 235- 240.
- [4] ZAJONC, Ivo. 2008. Teleskopie. Česká astronomická společnost [online]. [cit. 2015-05-14]. Dostupné z: <http://www.astro.cz/clanky/ostatni/teleskopie-dil-prvni-jaky-dalekohled-je-vhodny-pro-astronoma-amatera.html?hledat=Reflektor>
- [5] ŠULC, Miroslav. 2011. Jak funguje astronomický dalekohled. Česká astronomická společnost [online]. [cit. 2015-03-10]. Dostupné z: <http://www.astro.cz/clanky/ostatni/jak-funguje-astronomicky-dalekohled-dil-prvni.html?hledat=Jak%20funguje%20astronomick%C3%BD%20dalekohled>
- [6] Lift report [online]. [cit. 2015-02-15]. Dostupné z: <http://www.lift-report.de/index.php/news/176/368/Elevator-Ride-Quality>
- [7] ISO 18738-1:2012. "Measurement of ride quality -- Part 1: Lifts (elevators)". International Organization for Standardization. Retrieved 31 December 2014.
- [8] Math Works [online]. 1994. [cit. 2015-03-14]. Dostupné z: <http://www.mathworks.com/products/matlab/>Bruce R. Donald and Patrick G. Xavier, Provably
- [9] Good Approximation algorithms for dynamic Planning for Cartesian Robots and Open Chain Manipulators, Proceedings of Symposium on Computational Geometry, Berkley, California, USA pp290-300 (1990)
- [10] Nelineární rovnice. 2003. Centrum aplikovaná matematiky [online]. [cit. 2015-02-20].

Dostupné z: http://www.cam.zcu.cz/~danek/Students/2003_ZS/Materialy/nelinearni_rovnice.pdf

- [11] David Lindr, "Emin und Tmin Positionierung", Separáty výzkumné zprávy, Siemens Erlangen, Německo, 2014.
- [12] Kim Doang Nguyen; I-Ming Chen; Teck-Chew Ng, "Planning algorithms for s-curve trajectories," Advanced intelligent mechatronics, 2007 IEEE/ASME international conference on , vol., no., pp.1,6, 4-7 Sept. 2007.

A Příloha - Řešení rovnice (106)

$$\begin{aligned}
 A_{max} = & \left((1200 * R_{max}^2 * s_{req} - 1200 * R_{max}^2 * s_0 + 400 * a_{end}^3 \right. \\
 & + 10^{\frac{1}{2}} \\
 & * (-2000 * a_{end}^6 + 12000 * a_{end}^4 * R_{max} * |v_{end}| \\
 & + 108000 * a_{end}^4 * R_{max} * v_0 - 54000 * a_{end}^4 * a_0^2 \\
 & + 1800 * a_{end}^4 + 96000 * a_{end}^3 * R_{max}^2 * s_{req} \\
 & - 96000 * a_{end}^3 * R_{max}^2 * s_0 + 96000 * a_{end}^3 * R_{max} \\
 & * a_0 * v_0 - 32000 * a_{end}^3 * a_0^3 - 72000 * a_{end}^2 * R_{max}^2 \\
 & * |v_{end}|^2 - 432000 * a_{end}^2 * R_{max}^2 * |v_{end}| * v_0 \\
 & + 72000 * a_{end}^2 * R_{max}^2 * v_0^2 + 216000 * a_{end}^2 * R_{max} \\
 & * |v_{end}| * a_0^2 - 7200 * a_{end}^2 * R_{max} * |v_{end}| - 72000 \\
 & * a_{end}^2 * R_{max} * a_0^2 * v_0 + 7200 * a_{end}^2 * R_{max} * v_0 \\
 & + 18000 * a_{end}^2 * a_0^4 - 3600 * a_{end}^2 * a_0^2 + 90 * a_{end}^2 \\
 & - 288000 * a_{end} * R_{max}^3 * s_{req} * |v_{end}| + 288000 \\
 & * a_{end} * R_{max}^3 * |v_{end}| * s_0 - 288000 * a_{end} * R_{max}^2 \\
 & * |v_{end}| * a_0 * v_0 + 96000 * a_{end} * R_{max} * |v_{end}| * a_0^3 \\
 & + 144000 * R_{max}^4 * s_{req}^2 - 288000 * R_{max}^4 * s_{req} * s_0 \\
 & + 144000 * R_{max}^4 * s_0^2 + 288000 * R_{max}^3 * s_{req} * a_0 \\
 & * v_0 + 144000 * R_{max}^3 * |v_{end}|^3 + 432000 * R_{max}^3 \\
 & * |v_{end}|^2 * v_0 - 144000 * R_{max}^3 * |v_{end}| * v_0^2 \\
 & - 288000 * R_{max}^3 * a_0 * s_0 * v_0 - 432000 * R_{max}^3 \\
 & * v_0^3 - 96000 * R_{max}^2 * s_{req} * a_0^3 - 216000 * R_{max}^2 \\
 & * |v_{end}|^2 * a_0^2 + 7200 * R_{max}^2 * |v_{end}|^2 + 144000 \\
 & * R_{max}^2 * |v_{end}| * a_0^2 * v_0 - 14400 * R_{max}^2 * |v_{end}| * v_0 \\
 & + 96000 * R_{max}^2 * a_0^3 * s_0 + 792000 * R_{max}^2 * a_0^2 * v_0^2 \\
 & - 50400 * R_{max}^2 * v_0^2 - 36000 * R_{max} * |v_{end}| * a_0^4 \\
 & + 7200 * R_{max} * |v_{end}| * a_0^2 - 180 * R_{max} * |v_{end}| \\
 & - 420000 * R_{max} * a_0^4 * v_0 + 50400 * R_{max} * a_0^2 * v_0 \\
 & - 1260 * R_{max} * v_0 + 70000 * a_0^6 - 12600 * a_0^4 \\
 & + 630 * a_0^2 - 9)^{\frac{1}{2}} - 400 * a_0^3 - 1200 * a_{end} * R_{max} \\
 & * |v_{end}| + 1200 * R_{max} * a_0 * v_0) / (60 * (-10 \\
 & * a_{end}^2 - 30 * a_0^2 + 20 * R_{max} * |v_{end}| + 60 * R_{max} \\
 & * v_0 + 1)) \left. \right) \tag{122}
 \end{aligned}$$

B Příloha - Řešení rovnice (107)

$$\begin{aligned}
 A_{max} = & (1200 * R_{max}^2 * s_{req} - 1200 * R_{max}^2 * s_0 + 400 * a_{end}^3 \\
 & + 10^{\frac{1}{2}} \\
 & * (70000 * a_{end}^6 - 420000 * a_{end}^4 * R_{max} * v_{end} \\
 & - 36000 * a_{end}^4 * R_{max} * v_0 + 18000 * a_{end}^4 * a_0^2 \\
 & - 12600 * a_{end}^4 + 96000 * a_{end}^3 * R_{max}^2 * s_{req} \\
 & - 96000 * a_{end}^3 * R_{max}^2 * s_0 + 96000 * a_{end}^3 * R_{max} \\
 & * a_0 * v_0 - 32000 * a_{end}^3 * a_0^3 + 792000 * a_{end}^2 \\
 & * R_{max}^2 * v_{end}^2 + 144000 * a_{end}^2 * R_{max}^2 * v_{end} * v_0 \\
 & - 216000 * a_{end}^2 * R_{max}^2 * v_0^2 - 72000 * a_{end}^2 * R_{max} \\
 & * v_{end} * a_0^2 + 50400 * a_{end}^2 * R_{max} * v_{end} + 216000 \\
 & * a_{end}^2 * R_{max} * a_0^2 * v_0 + 7200 * a_{end}^2 * R_{max} * v_0 \\
 & - 54000 * a_{end}^2 * a_0^4 - 3600 * a_{end}^2 * a_0^2 + 630 \\
 & * a_{end}^2 - 288000 * a_{end} * R_{max}^3 * s_{req} * v_{end} \\
 & + 288000 * a_{end} * R_{max}^3 * v_{end} * s_0 - 288000 * a_{end} \\
 & * R_{max}^2 * v_{end} * a_0 * v_0 + 96000 * a_{end} * R_{max} * v_{end} \\
 & * a_0^3 + 144000 * R_{max}^4 * s_{req}^2 - 288000 * R_{max}^4 * s_{req} \\
 & * s_0 + 144000 * R_{max}^4 * s_0^2 + 288000 * R_{max}^3 * s_{req} \\
 & * a_0 * v_0 - 432000 * R_{max}^3 * v_{end}^3 - 144000 * R_{max}^3 \\
 & * v_{end}^2 * v_0 + 432000 * R_{max}^3 * v_{end} * v_0^2 - 288000 \\
 & * R_{max}^3 * a_0 * s_0 * v_0 + 144000 * R_{max}^3 * v_0^3 - 96000 \\
 & * R_{max}^2 * s_{req} * a_0^3 + 72000 * R_{max}^2 * v_{end}^2 * a_0^2 \\
 & - 50400 * R_{max}^2 * v_{end}^2 - 432000 * R_{max}^2 * v_{end} * a_0^2 \\
 & * v_0 - 14400 * R_{max}^2 * v_{end} * v_0 + 96000 * R_{max}^2 * a_0^3 \\
 & * s_0 - 72000 * R_{max}^2 * a_0^2 * v_0^2 + 7200 * R_{max}^2 * v_0^2 \\
 & + 108000 * R_{max} * v_{end} * a_0^4 + 7200 * R_{max} * v_{end} \\
 & * a_0^2 - 1260 * R_{max} * v_{end} + 12000 * R_{max} * a_0^4 * v_0 \\
 & - 7200 * R_{max} * a_0^2 * v_0 - 180 * R_{max} * v_0 - 2000 \\
 & * a_0^6 + 1800 * a_0^4 + 90 * a_0^2 - 9)^{\frac{1}{2}} - 400 * a_0^3 \\
 & - 1200 * a_{end} * R_{max} * v_{end} + 1200 * R_{max} * a_0 \\
 & * v_0)/(60 * (-30 * a_{end}^2 - 10 * a_0^2 + 60 * R_{max} \\
 & * v_{end} + 20 * R_{max} * v_0 + 1))
 \end{aligned} \tag{123}$$

C Příloha - Řešení rovnice (108)

$$\begin{aligned}
 A_{max} = & (300 * R_{max}^2 * s_{req} - 300 * R_{max}^2 * s_0 + 100 * a_{end}^3 + 5^{\frac{1}{2}} \\
 & * (8750 * a_{end}^6 + 52500 * a_{end}^4 * R_{max} * v_{end} + 4500 \\
 & * a_{end}^4 * R_{max} * v_0 + 2250 * a_{end}^4 * a_0^2 - 3150 * a_{end}^4 \\
 & + 12000 * a_{end}^3 * R_{max}^2 * s_{req} - 12000 * a_{end}^3 * R_{max}^2 \\
 & * s_0 - 12000 * a_{end}^3 * R_{max} * a_0 * v_0 - 4000 * a_{end}^3 \\
 & * a_0^3 + 99000 * a_{end}^2 * R_{max}^2 * v_{end}^2 + 18000 * a_{end}^2 \\
 & * R_{max}^2 * v_{end} * v_0 - 27000 * a_{end}^2 * R_{max}^2 * v_0^2 \\
 & + 9000 * a_{end}^2 * R_{max} * v_{end} * a_0^2 - 12600 * a_{end}^2 \\
 & * R_{max} * v_{end} - 27000 * a_{end}^2 * R_{max} * a_0^2 * v_0 \\
 & - 1800 * a_{end}^2 * R_{max} * v_0 - 6750 * a_{end}^2 * a_0^4 - 900 \\
 & * a_{end}^2 * a_0^2 + 315 * a_{end}^2 + 36000 * a_{end} * R_{max}^3 \\
 & * s_{req} * v_{end} - 36000 * a_{end} * R_{max}^3 * v_{end} * s_0 \\
 & - 36000 * a_{end} * R_{max}^2 * v_{end} * a_0 * v_0 - 12000 \\
 & * a_{end} * R_{max} * v_{end} * a_0^3 + 18000 * R_{max}^4 * s_{req}^2 \\
 & - 36000 * R_{max}^4 * s_{req} * s_0 + 18000 * R_{max}^4 * s_0^2 \\
 & - 36000 * R_{max}^3 * s_{req} * a_0 * v_0 + 54000 * R_{max}^3 \\
 & * v_{end}^3 + 18000 * R_{max}^3 * v_{end}^2 * v_0 - 54000 * R_{max}^3 \\
 & * v_{end} * v_0^2 + 36000 * R_{max}^3 * a_0 * s_0 * v_0 - 18000 \\
 & * R_{max}^3 * v_0^3 - 12000 * R_{max}^2 * s_{req} * a_0^3 + 9000 \\
 & * R_{max}^2 * v_{end}^2 * a_0^2 - 12600 * R_{max}^2 * v_{end}^2 - 54000 \\
 & * R_{max}^2 * v_{end} * a_0^2 * v_0 - 3600 * R_{max}^2 * v_{end} * v_0 \\
 & + 12000 * R_{max}^2 * a_0^3 * s_0 - 9000 * R_{max}^2 * a_0^2 * v_0^2 \\
 & + 1800 * R_{max}^2 * v_0^2 - 13500 * R_{max} * v_{end} * a_0^4 \\
 & - 1800 * R_{max} * v_{end} * a_0^2 + 630 * R_{max} * v_{end} \\
 & - 1500 * R_{max} * a_0^4 * v_0 + 1800 * R_{max} * a_0^2 * v_0 \\
 & + 90 * R_{max} * v_0 - 250 * a_0^6 + 450 * a_0^4 + 45 * a_0^2 \\
 & - 9)^{\frac{1}{2}} - 100 * a_0^3 + 300 * a_{end} * R_{max} * v_{end} - 300 \\
 & * R_{max} * a_0 * v_0)/(30 * (15 * a_{end}^2 + 5 * a_0^2 + 30 \\
 & * R_{max} * v_{end} + 10 * R_{max} * v_0 - 1))
 \end{aligned} \tag{124}$$

D Příloha - Řešení rovnice (109)

$$\begin{aligned}
 A_{max} = & (300 * R_{max}^2 * s_{req} - 300 * R_{max}^2 * s_0 + 100 * a_{end}^3 + 5^{\frac{1}{2}} \\
 & * (-250 * a_{end}^6 - 1500 * a_{end}^4 * R_{max} * v_{end} \\
 & - 13500 * a_{end}^4 * R_{max} * v_0 - 6750 * a_{end}^4 * a_0^2 \\
 & + 450 * a_{end}^4 + 12000 * a_{end}^3 * R_{max}^2 * s_{req} - 12000 \\
 & * a_{end}^3 * R_{max}^2 * s_0 - 12000 * a_{end}^3 * R_{max} * a_0 * v_0 \\
 & - 4000 * a_{end}^3 * a_0^3 - 9000 * a_{end}^2 * R_{max}^2 * v_{end}^2 \\
 & - 54000 * a_{end}^2 * R_{max}^2 * v_{end} * v_0 + 9000 * a_{end}^2 \\
 & * R_{max}^2 * v_0^2 - 27000 * a_{end}^2 * R_{max} * v_{end} * a_0^2 \\
 & + 1800 * a_{end}^2 * R_{max} * v_{end} + 9000 * a_{end}^2 * R_{max} \\
 & * a_0^2 * v_0 - 1800 * a_{end}^2 * R_{max} * v_0 + 2250 * a_{end}^2 \\
 & * a_0^4 - 900 * a_{end}^2 * a_0^2 + 45 * a_{end}^2 + 36000 * a_{end} \\
 & * R_{max}^3 * s_{req} * v_{end} - 36000 * a_{end} * R_{max}^3 * v_{end} \\
 & * s_0 - 36000 * a_{end} * R_{max}^2 * v_{end} * a_0 * v_0 - 12000 \\
 & * a_{end} * R_{max} * v_{end} * a_0^3 + 18000 * R_{max}^4 * s_{req}^2 \\
 & - 36000 * R_{max}^4 * s_{req} * s_0 + 18000 * R_{max}^4 * s_0^2 \\
 & - 36000 * R_{max}^3 * s_{req} * a_0 * v_0 - 18000 * R_{max}^3 \\
 & * v_{end}^3 - 54000 * R_{max}^3 * v_{end}^2 * v_0 + 18000 * R_{max}^3 \\
 & * v_{end} * v_0^2 + 36000 * R_{max}^3 * a_0 * s_0 * v_0 + 54000 \\
 & * R_{max}^3 * v_0^3 - 12000 * R_{max}^2 * s_{req} * a_0^3 - 27000 \\
 & * R_{max}^2 * v_{end}^2 * a_0^2 + 1800 * R_{max}^2 * v_{end}^2 + 18000 \\
 & * R_{max}^2 * v_{end} * a_0^2 * v_0 - 3600 * R_{max}^2 * v_{end} * v_0 \\
 & + 12000 * R_{max}^2 * a_0^3 * s_0 + 99000 * R_{max}^2 * a_0^2 * v_0^2 \\
 & - 12600 * R_{max}^2 * v_0^2 + 4500 * R_{max} * v_{end} * a_0^4 \\
 & - 1800 * R_{max} * v_{end} * a_0^2 + 90 * R_{max} * v_{end} \\
 & + 52500 * R_{max} * a_0^4 * v_0 - 12600 * R_{max} * a_0^2 * v_0 \\
 & + 630 * R_{max} * v_0 + 8750 * a_0^6 - 3150 * a_0^4 + 315 \\
 & * a_0^2 - 9)^{\frac{1}{2}} - 100 * a_0^3 + 300 * a_{end} * R_{max} * v_{end} \\
 & - 300 * R_{max} * a_0 * v_0) / (30 * (5 * a_{end}^2 + 15 * a_0^2 \\
 & + 10 * R_{max} * v_{end} + 30 * R_{max} * v_0 - 1))
 \end{aligned} \tag{125}$$

E CD příloha

- Text diplomové práce
 - diplomova_prace_2015_Jakub_Nosek.pdf
 - diplomova_prace_2015_Jakub_Nosek.doc
 - kopie_zadani_diplomove_prace_Jakub_Nosek.pdf
- Zdrojové kódy
 - Simulační skripty
 - Testovací skripty
 - Skripty pro odvození