

ZÁPADOČESKÁ UNIVERZITA V PLZNI
FAKULTA APLIKOVANÝCH VĚD

DISERTAČNÍ PRÁCE

2014

Ing. Lukáš BLÁHA

ZÁPADOČESKÁ UNIVERZITA V PLZNI
FAKULTA APLIKOVANÝCH VĚD

PLÁNOVÁNÍ POHYBU PODÉL
SPECIFIKOVANÉ CESTY S UVAŽOVÁNÍM
VŠECH POHYBOVÝCH OMEZENÍ

Ing. Lukáš BLÁHA

disertační práce
k získání akademického titulu Doktor
v oboru Kybernetika

Školitel: Prof. Ing. Miloš Schlegel, CSc
Katedra: Katedra kybernetiky

Plzeň 2014

UNIVERSITY OF WEST BOHEMIA
FACULTY OF APPLIED SCIENCES

TRANSFER PLANNING ALONG A GIVEN
PATH WITH HARD MOTION
CONSTRAINTS

Ing. Lukáš BLÁHA

Dissertation to obtain
the degree of Doctor in Cybernetics

Supervisor: Prof. Ing. Miloš Schlegel, CSc
Department: Department of Cybernetics

Pilsen 2014

Čestné prohlášení

Prohlašuji, že jsem tuto disertační práci vypracoval samostatně a výhradně s použitím citovaných pramenů, literatury a dalších odborných zdrojů.

V Plzni dne 30.8.2014

Never run faster than your guardian angel can fly.
In memory of all my heroes who were too fast.

PODĚKOVÁNÍ

Nejprve bych rád poděkoval mému školiteli Miloši Schlegelovi. Miloš byl ten, kdo mi ukázal krásy teorie řízení a mechaniky v její nejčistší podobě. Jeho podpora mé práce byla pro mě vždy hnací silou. Rád bych poděkoval za všechny poskytnuté rady a připomínky, které přišly vždy v pravý čas. Nápad, myšlenky a způsoby řešení některých problémů jsou pro mě a mé kolegy vždy motivací a bránou k novému poznání světa kolem nás. Jeho kreativita a nadšení jsou nakažlivé.

Rád bych vyjádřil upřímné poděkování také mým nejbližším spolupracovníkům Martinu Švejdovi a Arnoldu Jágerovi, se kterými jsem za ta léta strávil spoustu času debatováním o aktuálně řešených problémech. Naše rozdílné znalosti nám zpravidla vždy pomohly udělat si lepší náhled na daný problém a většinou i nalézt odpovídající řešení.

Dále bych chtěl také poděkovat mým ostatním spolupracovníkům a kamarádům, především naší REX Controls skupině, která za všech okolností drží pohromadě, a umožňuje nám všem jít si za svými cíli ve vývoji a výzkumu. Za dobu mého působení na univerzitě se z nás všech stal těsný kolektiv kamarádů a spolupracovníků, který se vídá v takovém počtu opravdu zřídka. A já jsem rád, že mohu být stále jeho součástí.

Konečně bych chtěl poděkovat mým rodičům Miladě a Mírovi za jejich lásku a víru ve mně. Jejich srdečná podpora mi umožnila snadno vstoupit do mého osobního života a okolního světa.

Moje poslední poděkování patří Pětě, která stála celou dobu po mém boku, poslouchala moje nadšené vyprávění o řešených problémech a vždy se na mě uměla krásně smát.

ABSTRAKT

Tato práce se zabývá řízením pohybu mechatronických systémů. Návrh řízení lze obecně rozdělit do tří úrovní: plánování geometrické cesty stroje, plánování trajektorie pohybu podél této specifikované cesty a zpětnovazební řízení za účelem sledování naplánovaných trajektorií pohybu. Práce se soustřeďuje na druhou úroveň návrhu řízení. V úvodu je stručně představena základní úloha plánování pohybu, včetně standardního členění do výše uvedených úrovní. Úvod je dále věnován druhé úrovni plánování a jejím typickým úlohám. V druhé kapitole je představen matematický nástroj z oblasti algebraické geometrie, jmenovitě Gröbnerova báze, a jsou ukázány možnosti využití tohoto nástroje v teorii řízení. V třetí kapitole je Gröbnerova báze využita k řešení časově optimálního plánování pohybu pro úlohy s konstantními omezeními. Je nalezen analytický algoritmus výpočtu trajektorií polohy, rychlosti a zrychlení ve stavovém prostoru generátoru pohybu s respektováním všech stavových omezení. Omezení jsou v této úloze uvažována konstantní na celém rozsahu pohybu. Algoritmus generující trajektorie pohybu obsahuje původní procedury a tvoří hlavní výsledky práce. V další části práce je představena složitější úloha časově optimálního plánování pohybu. Jedná se opět o plánování pohybu podél specifikované cesty stroje, ale s obecným tvarem omezení. Oproti standardním technikám, využívajících numerického řešení pohybových rovnic systému a metodu střelby, je využito výše zmíněných původních procedur. Pro diskretizovaná omezení je uveden obecný analytický postup, který tuto úlohu řeší geometricky a umožňuje nalézt trajektorie polohy, rychlosti a zrychlení v minimálním čase.

ABSTRACT

This thesis deals with motion control of mechatronic systems. The control design can be generally divided into three levels: geometric path planning, trajectory planning along the specified path, and feedback control of planned trajectory. The paper focuses on the second level of control design. The introduction briefly shows the basic task of motion planning, including the standard division to the above mentioned levels. Further the mathematical tool named Gröbner basis is introduced. The second chapter introduces a mathematical tool from the field of algebraic geometry, namely Gröbner bases. There are shown some standard applications for use of this tool in control theory. In the third chapter, the Gröbner basis is used to solve time-optimal motion planning problem with motion constraints. It is found analytical algorithm for generating trajectories of position, velocity and acceleration in state-space of generator, respecting all motion constraints. In this task the motion constraints are considered constant in the whole range of motion. The algorithm for generating motion trajectories contains the original procedures and constitutes the main results of the thesis. The next section presents more complex task of time-optimal motion planning, where the curves of constraints can be of general shape. Compared with standard optimization techniques using nonlinear programming, or other numerical methods the presented procedure uses the procedures mentioned above. It is found a general procedure that solves this task geometrically and, for the discretized constraints, allows to find the trajectory of position, velocity and acceleration in minimum time.

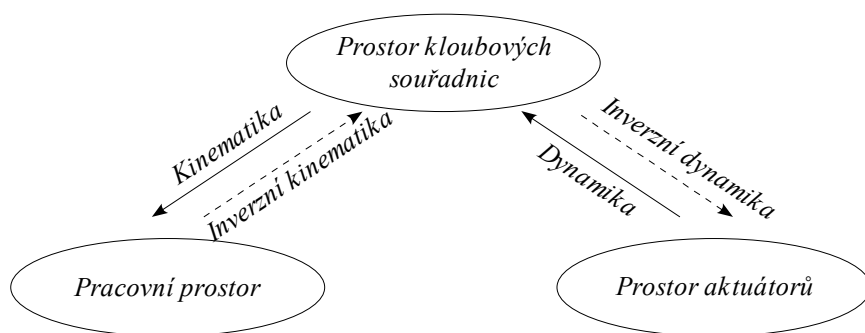
OBSAH

Poděkování.....	11
Abstrakt.....	13
Abstract.....	15
1 Úvod.....	19
1.1 Plánování a generování pohybu mechatronických systémů.....	20
1.1.1 Plánování pohybu v jedné ose.....	21
1.1.2 Plánování pohybu podél specifikované cesty.....	22
1.1.3 Časová optimalita a plánování pohybu v omezeném stavovém prostoru.....	24
1.2 Úvod do Gröbnerovy báze.....	25
1.2.1 Historické kořeny Gröbnerovy báze.....	25
1.2.2 Co je to Gröbnerova báze.....	26
1.3 Hlavní cíle a výsledky práce.....	28
2 Gröbnerova báze - Přípravná kapitola.....	31
2.1 Polynom, Varieta, Ideál.....	32
2.1.1 Afinní prostor.....	32
2.1.2 Afinní variety a jejich parametrizace.....	32
2.1.3 Ideál.....	34
2.2 Uspořádání a redukce polynomů více proměnných.....	36
2.2.1 Uspořádání.....	36
2.2.2 Redukce polynomů více proměnných.....	37
2.3 Gröbnerova báze.....	38
2.4 Algoritmus (Buchberger).....	39
2.4.1 S-polynom.....	39
2.4.2 Redukovaná Gröbnerova báze.....	41
2.4.3 Korektnost algoritmu.....	42
2.4.4 Konečnost algoritmu a Dickson's lemma.....	42
2.4.5 Specializace (Gauss, Eukleidés).....	43
2.5 Vlastnosti a aplikace Gröbnerovy báze ve výpočetní algebraické geometrii.....	43
2.5.1 Dimenze variety a faktorový okruh.....	43
2.5.2 Hlavní problém teorie ideálů (ideal membership problem).....	48
2.5.3 Rovnost ideálů.....	49
2.5.4 Hledání průniku ideálů.....	49
2.5.5 Problém eliminace.....	50
2.5.6 Implicitizace.....	50
2.6 Aplikace Gröbnerovy báze v teorii řízení.....	52
2.6.1 Gröbnerova báze a kinematická úloha.....	52
2.6.2 Přiřazení pólů výstupní zpětnou vazbou.....	56
3 Časově optimální řízení řetězce integrátorů jako generátoru pohybu.....	63
3.1 Obecný problém optimálního řízení.....	63
3.2 Obecný problém t-optimálního řízení.....	64
3.2.1 Řetězec integrátorů a t-optimální řízení.....	65
3.2.2 Problém obecného konce v t-optimálním řízení.....	68
3.3 Řetězec integrátorů s omezeními jako generátor t-optimálního pohybu.....	68
3.3.1 Standardní plánovací algoritmus BAVS.....	70
3.3.2 Základní struktura a vlastnosti.....	71
3.3.3 Tvary trajektorie.....	73
3.4 Gröbnerova báze a t-optimální řízení.....	83
3.4.1 Systém druhého řádu (2 integrátory s omezeními).....	83
3.4.2 Systém třetího řádu (3 integrátory s omezeními).....	90
3.5 Algoritmus GBAVS.....	94
3.5.1 Základní struktura algoritmu GBAVS.....	95

3.5.2	Obecný počátek a konec.....	95
3.5.3	Pohyb z konstantní rychlosti do konstantní rychlosti.....	106
3.5.4	Pohyb z klidu do klidu.....	120
3.5.5	Standardní aplikace algoritmu GBAVS v plánování pohybu.....	122
4	Plánování pohybu podél specifikované cesty.....	131
4.1	Kinematika a dynamika systému podél specifikované cesty.....	132
4.1.1	Kinematika podél specifikované cesty v pracovním prostoru.....	133
4.1.2	Kinematika podél specifikované cesty v konfiguračním prostoru.....	134
4.1.3	Dynamika systému podél specifikované cesty.....	135
4.1.4	Omezení a jejich vliv na problém plánování.....	136
4.2	Parametrické plánování s vyloučením času.....	138
4.3	Plánování podél specifikované cesty systémem druhého řádu.....	139
4.3.1	Základní popis a definice úlohy.....	139
4.3.2	Plánovací algoritmus – Triangle Algorithm.....	139
4.3.3	Plánovací algoritmus GAVS(s).....	144
4.3.4	Plánovací algoritmus GAVS(s,v).....	146
4.4	Planární manipulátor - ilustrační příklad.....	150
4.4.1	Zadání úlohy.....	150
4.4.2	Kinematika manipulátoru podél specifikované cesty.....	152
4.4.3	Dynamika manipulátoru podél specifikované cesty.....	153
4.4.4	Sestrojení limitních křivek omezení.....	154
4.4.5	Výpočet přesunu systému podél cesty generátorem 2.řádu.....	155
	Závěr.....	159
	Shrnutí.....	161
	Resume.....	163
	A Nové směry v teorii Gröbnerovy báze.....	165
	B Algoritmy GAVS a GBAVS.....	172
	Literatura.....	183
	Seznam publikovaných i nepublikovaných prací autora.....	189
4.4.1	Publikace.....	189
4.4.2	Ostatní práce.....	190

1 ÚVOD

Plánování pohybu je základní problematikou mechatronických systémů. Obecně je plánování pohybu ne zcela definovaný problém, jemuž vyhovuje více řešení. Přirozeně je pak vhodné zvolit určité kritérium kvality a hledat takové řešení, které toto kritérium optimalizuje. Plánování a řízení pohybu lze tedy chápat jako variační problém. Je možné ho rozdělit do dvou větví: na kinematické a dynamické plánování. Kinematické plánování neuvažuje dynamické chování systému, pouze respektuje kinematické závislosti, omezení na pracovní prostor, rychlost a možné zrychlení systému, v literatuře známá jako geometrická, diferenciální a fázová omezení. Dynamické plánování pak hledá takové velikosti akčních sil a momentů, které jsou konzistentní s dynamickými rovnicemi systému, splňují diferenciální a fázová omezení a minimalizují kritérium kvality. Jedním z nejčastěji řešených úkolů v robotice je přesun robota nebo jeho části mezi danými body, specifikovanými v jeho pracovním prostoru. Kromě polohové závislosti je obvyklým požadavkem, aby pohyb byl v určitém smyslu efektivní nebo dokonce optimální. V případě složitého pracovního prostoru nebo existence překážek je mnohdy vhodné, či dokonce nezbytné, specifikovat trajektorii pohybu dopředu. Aktuátory robota nemohou generovat nekonečné síly, momenty a rychlosti, takže je nutné plánovat pohyb po dané cestě tak, aby nebyly porušeny limity a omezení aktuátorů. Samozřejmě je také vhodné klást určité požadavky na tvar trajektorií rychlosti a zrychlení, jejich strmost, křivost apod. Tyto požadavky většinou vyplývají ze samotné úlohy řízení pohybu, kde například nechceme nespojitostmi ve zrychlení vybudit některé vlastní frekvence stroje, které by narušily žádané chování nebo třeba i destabilizovaly pohyb stroje.



Obrázek 1.1: Prostory, ve kterých lze pozorovat a plánovat pohyb.

1.1 PLÁNOVÁNÍ A GENEROVÁNÍ POHYBU MECHATRONICKÝCH SYSTÉMŮ

V automatickém řízení, robotice a mechatronice se prakticky kdekoli setkáváme s problémem plánování a řízení pohybu. Ať už se jedná o polohování koncového efektoru manipulátoru, řízení robotického vozítka, či některé pohyblivé součásti průmyslového stroje. Všechny tyto úlohy vedou na společný problém, jakým způsobem pohyb naplánovat a jak ho následně vykonat. V praktických úlohách je nutné dodržovat určitá pravidla a omezení, která vyplývají z povahy řízeného systému. Jedná se především o konstrukční vlastnosti stroje, omezení aktuátorů a senzorů, omezení vzniklé prostředím, ve kterém se stroj nachází, pohybuje atd.

Řízení pohybu můžeme rozdělit do tří základních úrovní:

1. *Plánování cesty v pracovním prostoru (path planning) – geometrické plánování cesty pohybu vzhledem k požadavkům úlohy, překážkám, tvaru pracovního prostoru, atd. (polohové závislosti)*

Například u CNC strojů je plánování cesty pohybu klíčovou otázkou. Většinou se využívá SPLINE/NURBS křivek [80],[47], na které je možné klást další požadavky. Například pohyb v 3D prostoru je možné definovat sekvencí řídicích bodů postupně tak, aby cesta procházela každým z těchto bodů, a současně byly splněny požadavky na rychlost a zrychlení v každém bodě. Nebo například dodržena maximální křivost výsledné trajektorie pohybu.

2. *Plánování pohybu podél cesty vzhledem k omezením na pohybové veličiny (trajectory planning) – (rychlostní a zrychlující závislosti podél polohy)*

Jakmile máme k dispozici naplánovanou cestu pohybu stroje, můžeme přistoupit k plánování průjezdu po této cestě s respektováním vlastností stroje a omezení. Zde jsou ke geometrickému tvaru cesty přidány časové značky, nebo je pohyb systému vztažen k naplánované cestě. Mezi standardní úlohy patří časově optimální (t-optimální) průjezd dané cesty [13],[43],[44],[46], pohyb s minimalizací akčních zásahů [60],[84], pohyb konstantní rychlostí.

V praxi je ve všech případech nutné respektovat všechna omezení systému a úlohy. Typickým příkladem může být t-optimální pohyb kolejového vozidla, kde není možné zanedbat vliv odstředivých sil v zatáčkách a je nutné respektovat tvar omezení podél cesty. Z křivek omezení můžeme například zjistit, jaká může být maximální konstantní rychlost pohybu bez nebezpečí převrácení vozíku, nebo jaké limitní hodnoty je třeba vložit do plánovacího algoritmu pro návrh t-optimálního přesunu.

Někdy není principiálně možné oddělit úlohu plánování cesty a plánování trajektorií pohybu. Jedná se například o problémy přemísťování mezi pohyblivými cíli. Typickým problémem je plánování přistání sondy Curiosity na Mars, kde rychlost pohybu a tvar cesty jsou závislými veličinami a nelze je v problému plánování oddělit. Těmito úlohami se zabývá například Sussmann [94] a některé speciální případy byly vyřešeny, nebo alespoň částečně vyřešeny pomocí technik Lieovy algebry. V této práci se těmito úlohám věnovat nebudeme.

3. *Sledování naplánovaných křivek (trajectory tracking) – zajištění, aby to, co bylo naplánováno, bylo splněno.*

Po procesu naplánování cesty v pracovním prostoru a naplánování trajektorií pohybu jsou generované trajektorie polohy, rychlosti a zrychlení předány na nejnižší úroveň řízení. Zpětnovazební regulátor musí zajistit sledování předdefinovaných trajektorií, ať v pracovním prostoru nástroje nebo v prostoru kloubových souřadnic, a je poslední

a nejnižší úrovni řízení pohybu.

V mnoha případech se využívá kaskádní regulace [3]. Především proto, že trajektorie bývá často generována pro každou veličinu pohybu (poloha, rychlost, zrychlení) zvlášť. Pokud jsme schopni měřit přímo některou složku stavu nebo stav rekonstruovat, potom je možné využít například stavový regulátor s dopřednou vazbou.

Plánování cesty a pohybu po této cestě je zpravidla proces v otevřené smyčce na rozdíl od poslední úrovně. Může být prováděno on-line nebo off-line.

1.1.1 PLÁNOVÁNÍ POHYBU V JEDNÉ OSE

Plánování pohybu v jedné ose je základní úlohou plánování pohybu. Aplikace najdeme především ve strojovém obrábění nebo v úlohách přepravování materiálu či osob. Jedná se například o řízení otáček vrtaček, soustruhů a dalších vřeten obráběcích strojů. Dále pak řízení dopravníků, eskalátorů, výtahů apod. Všechny tyto úlohy mají jeden stupeň volnosti, nebo je možné tyto stupně řídit nezávisle, a tedy plánovat pohyb v každé ose zvlášť. Typickým příkladem jsou rychloposuvy na obráběcích strojích. Důležitým faktorem v plánování pohybu je obvykle plynulost přechodu mezi požadovanými rychlostmi a strmost náběhu na danou rychlost. Tyto požadavky vyplývají buď z technologie, v případě obrábění a přesunu materiálu, nebo z požadavků na bezpečnost a citlivost lidského organismu v případě transportu osob. Příkladem může být maximální rychlost eskalátoru a jeho maximální rozběh a doběh. Omezení v těchto úlohách jsou uvažována konstantní na celém rozsahu pohybu.

POHYB Z KLIDU DO KLIDU

Je nejjednodušší úlohou plánování pohybu systému. Při plánování v jedné ose hledáme takové trajektorie polohy, rychlosti a zrychlení, které respektují systémová omezení, počáteční a koncovou polohu stroje. Jestliže budeme uvažovat časovou optimalitu jako kritérium výběru trajektorie pohybu, potom je možné převést úlohu plánování na problém t-optimálního řízení řetězce integrátorů s omezeními na stav a vstup systému. V praxi je nejčastěji využíváno řetězce dvou integrátorů [57],[52],[95], protože vstup řetězce můžeme chápat jako zrychlení, první integraci jako rychlost a výstup jako polohu. Jedná se o generátor pohybu druhého řádu. Podrobně bude generování pohybu tímto postupem vysvětleno v kapitole 3.4.

OBECNÝ POHYB V JEDNÉ OSE

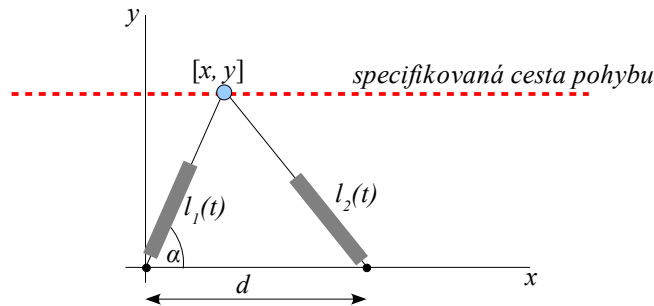
Mnoho úloh v řízení pohybu vyžaduje, aby přesun systému byl proveden po předem definované trajektorii pohybu. V praxi se většinou jedná o zadanou sekvenci poloh, rychlostí a zrychlení v čase, nebo rychlostí a zrychlení vzhledem k poloze systému. Řídící nebo plánovací algoritmus musí zajistit, aby trajektorie procházela danými body a většinou se jedná o polynomiální interpolační algoritmus. Protože požadujeme hladkost pohybu, je třeba využít polynomy vyšších řádů, nebo po částech polynomiální interpolaci spline funkcemi.

Z praktického hlediska je možné problém formulovat opět jako problém t-optimálního řízení. Jestliže je sekvence bodů vztažena k poloze systému, potom můžeme hledat takovou řídicí strategii řetězce integrátorů, která zajistí postupný průběh všemi body ve stavovém prostoru řetězce integrátorů vzhledem k omezením na jeho stav a vstup. Klíčovou vlastností generátoru pohybu je, aby uměl plánovat pohyb z libovolného stavu do libovolného stavu. Potom je možné napojovat dílčí pohyby mezi body sekvence a respektovat jednotlivá omezení na každém úseku. Samozřejmě se nejedná o triviální problém, který má vždy řešení. Podrobně viz kapitola 3.4.

1.1.2 PLÁNOVÁNÍ POHYBU PODÉL SPECIFIKOVANÉ CESTY

Plánováním pohybu podél specifikované cesty budeme, vzhledem k druhé úrovni řízení pohybu, chápat problém generování trajektorií polohy, rychlosti a zrychlení stroje s ohledem na jeho kinematické a dynamické vlastnosti, které se podél cesty mění. Ty způsobí, že omezení podél cesty nemusí nabývat konstantních hodnot a tedy radikálně zkomplikují úlohu plánování pohybu. Dokonce nelze předpokládat ani spojitost těchto omezení, natož jejich hladkost.

Pro představu uvažujme jednoduchý planární paralelní manipulátor, tvořený třemi rotačními klouby a dvěma výsuvy $l_1(t), l_2(t)$, jejichž velikost můžeme měnit, viz obrázek 1.2. Předpokládejme pro jednoduchost, že velikost výsuvu není omezena. Takový manipulátor má dva stupně volnosti, singularitu na ose x a na kružnici se středem v $[d/2, 0]$ o nekonečném poloměru. Předpokládejme, že úkolem manipulátoru je pohybovat koncovým efektem po přímce rovnoběžné s osou x . Zajímá nás maximální možná konstantní rychlost pohybu po této cestě vzhledem k omezení jednotlivých výsuvů $|\dot{l}_1(t)| \leq \dot{l}_{1max}, |\dot{l}_2(t)| \leq \dot{l}_{2max}$ a maximální možná dosažitelná rychlost.



Obrázek 1.2: Struktura manipulátoru včetně důležitých parametrů.

Předpokládejme plánování v kinematické oblasti, viz obrázek 1.1. Zajímají nás tedy pouze kinematické závislosti. Podívejme se, jakým způsobem ovlivní kinematika stroje plánovací prostor a možnosti pohybu podél cesty. Přímé kinematické zobrazení lze vyjádřit ve tvaru

$$\begin{bmatrix} x \\ y \end{bmatrix} = \mathbf{r}(\mathbf{q}) = \begin{bmatrix} l_1 \cos \alpha \\ l_1 \sin \alpha \end{bmatrix} = \begin{bmatrix} \frac{l_1^2 - l_2^2 + d^2}{2d} \\ \sqrt{l_1^2 - \left(\frac{l_1^2 - l_2^2 + d^2}{2d} \right)^2} \end{bmatrix}. \quad (1.1)$$

Vyjádříme specifikovanou cestu pomocí parametrizace přirozeným parametrem s

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} x_0 + s \\ y_0 \end{bmatrix}. \quad (1.2)$$

Dosadíme do rovnice (1.1) a vyjádříme závislosti délek výsuvů na parametru s

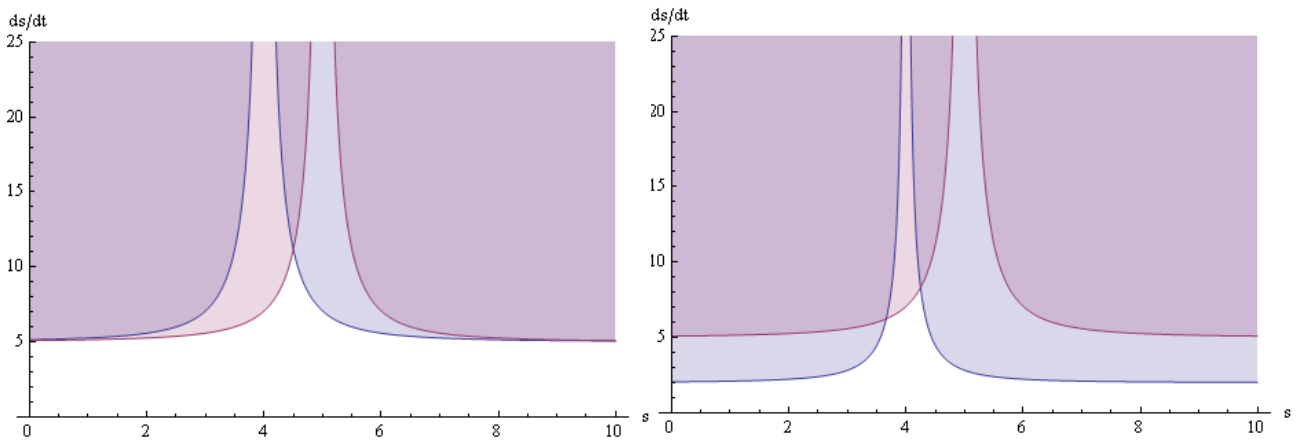
$$\begin{aligned} l_1(s) &= \sqrt{s^2 + 2s x_0 + x_0^2 + y_0^2} \\ l_2(s) &= \sqrt{d^2 - 2ds + s^2 - 2dx_0 + 2sx_0 + x_0^2 + y_0^2}. \end{aligned} \quad (1.3)$$

Derivací podle času dostaneme

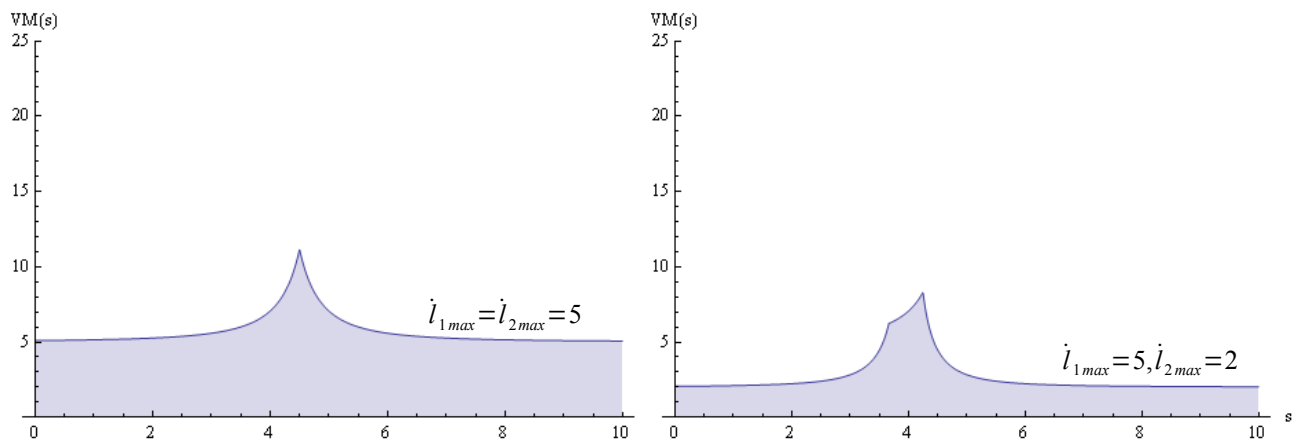
$$\begin{aligned}\frac{d l_1}{d t} &= \frac{2 s+2 x_0}{2 \sqrt{\left[s^2+2 s x_0+x_0^2+y_0^2\right]}} \frac{d s}{d t} \\ \frac{d l_2}{d t} &= \frac{-2 d+2 s+2 x_0}{2 \sqrt{\left[d^2-2 d s+s^2-2 d x_0+2 s x_0+x_0^2+y_0^2\right]}} \frac{d s}{d t}.\end{aligned}\quad (1.4)$$

Vyjádřením časové změny parametru s z (1.4) a dosažením limitních hodnot omezení $|l_1(t)| \leq \dot{l}_{1 \max}$, $|l_2(t)| \leq \dot{l}_{2 \max}$ dostaneme limitní křivky rychlosti jednotlivých výsuvů podél cesty, obrázek 1.3,

$$\begin{aligned}\frac{d s}{d t} &= \frac{2 s+2 x_0}{2 \sqrt{\left[s^2+2 s x_0+x_0^2+y_0^2\right]}} \dot{l}_{1 \max} \\ \frac{d s}{d t} &= \frac{2 \sqrt{\left[d^2-2 d s+s^2-2 d x_0+2 s x_0+x_0^2+y_0^2\right]}}{-2 d+2 s+2 x_0} \dot{l}_{2 \max}\end{aligned}\quad (1.5)$$



Obrázek 1.3: Limitní křivky rychlosti výsuvů v absolutní hodnotě pro dvě nastavení limitních rychlostí $\dot{l}_{1 \max}$, $\dot{l}_{2 \max}$. Vykresleno pro parametry $x_0=-4$, $y_0=1$, $d=1$.



Obrázek 1.4: Limitní křivka rychlosti pro dvě nastavení limitních hodnot výsuvů (její kladná část).

Průnikem výše uvedených křivek (1.5) je pak limitní křivka rychlosti podél specifikované cesty, $VM(s)$. Z obrázku 1.4 vidíme, že maximální konstantní rychlost podél cesty je v obou případech rovna minimu z omezení výsvů. Pokud bychom se chtěli pohybovat t-optimálně, by bylo možné měnit rychlost podle aktuální hodnoty omezení a využít celý rozsah rychlosti daný kinematikou manipulátoru. Maximální rychlost by v tomto případě mohla v úzkém rozsahu polohy narůst i několikanásobně. Z výše uvedeného je také zřejmé, že limitní křivka rychlosti není ani hladkou ani spojitou křivkou, neboť je výsledkem průniku dílčích limitních křivek jednotlivých omezení.

1.1.3 ČASOVÁ OPTIMALITA A PLÁNOVÁNÍ POHYBU V OMEZENÉM STAVOVÉM PROSTORU

Časová optimalita je jedním ze standardních kritérií v teorii řízení. Setkáme se s ní ve výrobním průmyslu, ekonomice, robotice, vojenství a dalších odvětvích, kde je kladen důraz na rychlost výroby, produkce, rychlost přesunu, atd. V terminologii optimalizace se jedná o úlohu s volným časem a pevným koncem. V případě časově optimálního řízení (t-optimálního řízení) bez stavových omezení musí platit a lze ukázat, že alespoň jeden z aktuátorů robota musí pracovat na hranici svého rozsahu, saturován v maximu nebo minimu. Strategie řízení je tedy známa a v literatuře nazývána jako *bang-bang* řízení [4],[17]. Celý problém návrhu řídicí strategie je redukován na nalezení časových intervalů, na jejichž hranicích bude docházet k přepnutí řízení. Tato vlastnost velmi zjednodušuje hledání řešení optimalizační úlohy, a proto bývá využívána pro generování pohybu v průmyslových algoritmech a knihovnách motion control [81].

Pokud uvažujeme t-optimální řízení s omezením na stavové proměnné, potom kromě předchozího může ještě nastat případ, kdy ani jeden aktuátor nemusí pracovat na hranici svého rozsahu. To může nastat v případě omezení rychlosti systému takovou funkcí, jejíž strmost je menší než maximální možná, kterou lze systémem vykonat. Potom aktuátory budou pracovat tak, aby se stav systému pohyboval po této omezující křivce rychlosti, dokud ji nebude třeba opustit směrem k žádanému koncovému stavu nebo za účelem respektování dalších omezení. Stav systému pohybující se na nějaké omezující křivce je v odborné literatuře nazýván dynamickou singularitou [42],[112]. Daný bod na omezující křivce je singulárním bodem. Posloupnost více bodů je singulárním obloukem. V tomto případě totiž neplatí vlastnosti *bang-bang* řízení. Řízení je zcela specifikováno strmostí omezující křivky. Jestliže omezující křivka vyplývá pouze z omezení na síly a momenty aktuátorů (a neuvažujeme omezení na rychlost), potom lze tato omezení modifikovat tak, že množina těchto omezení je takzvanou striktně konvexní množinou¹. Pro t-optimální úlohu s těmito vlastnostmi lze dokázat, že řízení bude opět pouze typu *bang-bang* a nebude obsahovat žádnou dynamickou singularitu [91].

Nalezení optimálního řešení pro úlohy s omezeným stavovým prostorem je dodnes jednou z nejproblematictějších oblastí teorie řízení. Dokonce i nalezení jakéhokoli řešení, splňující omezující podmínky, je většinou náročné. Nicméně v praktických aplikacích se setkáváme právě s těmito typy úloh. Zajímavý přístup uvedl S.S.L.Chang v roce 1963,[26]. Poukázal na své zkušenosti z výzkumu v U.S. Air Force a zdůraznil, že v praktických aplikacích není vhodné uvažovat zrychlení za neomezenou funkci a také, že její strmost je významnou veličinou. Zabýval se lineárními systémy jejichž řídicí veličina má nejen omezenou absolutní hodnotu, ale i strmost. Fakticky tak převedl problém časově optimálního pohybu mezi dvěma body s omezením vstupní veličiny na problém řízení systému třetího řádu s omezením jedné složky stavu. Nalezl řešení t-optimálního řízení pro lineární systémy s omezenou strmostí vstupních veličin. Odvození provedl bez využití Pontryaginova principu maxima. Za zmínku stojí i přístup pomocí lineárního programování [111], do něhož je možné zahrnout stavová omezení. Bohužel výpočetní náročnost

¹ Standardně je množina omezení uvažována jako n-dimenzionální polyhedron, který je pouze konvexní množinou, nikoli striktně konvexní.

prozatím zamezuje použití v real-time aplikacích. Poznamenejme ale, že jeho možnosti z hlediska změn omezení v průběhu pohybu a napojování jsou velmi zajímavé.

Do současnosti vzniklo mnoho dalších zajímavých přístupů, jejichž základní myšlenky budou nastíněny v kapitole 3.

OPTIMÁLNÍ ŘÍZENÍ A DIFERENCIÁLNÍ GEOMETRIE

V optimálním řízení obecně sehrává velkou roli diferenciální geometrie a Riemannův prostor [12]. Ukázalo se, že nutné podmínky optimality lze převést do geometrických podmínek pro jisté definované množiny. Systematická studie křivek (geodetik) v Riemannově prostoru, které minimalizují svoji délku, vedla až k zobecnění Pontryaginova principu maxima [103]. Další z podnětů vyšel od R. Brocketta, který poukázal na to, že systém řízení můžeme chápat jako systém vektorových polí. To vedlo k modifikaci Pontryaginova principu do tvaru, který není závislý na souřadném systému [102]. Problém omezení je i v této oblasti velkou překážkou a není dořešen. I tak je tato oblast optimálního řízení v poslední době nejvíce rozvíjena. K práci s těmito přístupy je zapotřebí mnoho znalostí z odvětví Riemannovské geometrie, Lieovy algebry, teorie konexí, geodetik a dalších.

1.2 ÚVOD DO GRÖBNEROVY BÁZE

Teorie Gröbnerovy báze (*GB*) byla původně vytvořena jako nástroj v oblasti algebraické geometrie. Nicméně postupem času se ukázalo, jak obrovskou oblast působení tato teorie pokrývá. Kromě oblastí matematiky ji nalezneme ve výpočetních oborech, statistice, bio-informatice, v teorii systémů, atd. Je až s podivem, kolik problémů z tolika odlišných oborů je možné převést na jeden a ten samý matematický problém, teorii Gröbnerových bází. Jakmile je možné transformovat původní úlohu do oblasti polynomů více proměnných, definovaných na polynomiálním okruhu, potom je tato úloha kandidátem na využití *GB*.

Teorie Gröbnerovy báze je širokou oblastí pokrývající pole lineární a diferenciální algebry. Jsou jí věnovány dlouhé statě v článcích, odborných publikacích, a v poslední době je součástí každého pokročilejšího matematického softwaru. My se pro naše účely soustředíme pouze na základní teorii a její důsledky, které jsou podstatné pro pochopení dalších kapitol a využití v praktických aplikacích. V následujícím textu bude stručně uvedena historie *GB* a nastíněny základní vlastnosti a možnosti jejího využití. Detailnější představení *GB* bude provedeno v kapitole 2.

1.2.1 HISTORICKÉ KOŘENY GRÖBNEROVY BÁZE

Gröbnerova báze je relativně mladou oblastí matematiky. Za jejím objevem je možné nalézt dvě významné tváře.

Wolfgang Gröbner (1899 – 1980), rakouský matematik.

Začal studovat strojírenství, ale v roce 1929 přešel na studium matematiky. Svoji dizertační práci nazvanou „*Ein Beitrag zum Problem der Minimalbasen*“ napsal v roce 1932 na vídeňské Univerzitě. Po promoci pokračoval ve studiích pod vedením Emmy Noetherové (komutativní algebra). Pod jeho vedením vznikl v roce 1965 pojem Gröbnerova báze.

Bruno Buchberger (1942), rakouský matematik.

Profesor výpočetní matematiky na Univerzitě Johannese Keplera v Linzi. V roce 1965 zavedl teorii Gröbnerovy báze, pojmenovanou po svém školiteli, a rozvíjí ji dodnes. V roce 1985 začal vydávat časopis *Journal of Symbolic Computation*, který se stal nejváženějším periodikem na poli výpočetní algebry. V roce 1987 byl oceněn a posazen do čela výzkumného institutu pro symbolické výpočty (RISC). Za teorii Gröbnerovy báze získal v roce 2007 ocenění *ACM Paris Kanellakis Theory and Practice Award*. Byl také oceněn zlatým řádem *Golden Medal of Honor*. Získal čestný doktorát od University of Nijmegen (1993) a West University of Timisoara (2000).

Analogický koncept pro *lokální okruhy* vymyslel nezávisle na Brunovi Buchbergerovi Heisuke Hironaka v roce 1964, který jej pojmenoval *standardní báze*. Analogická teorie pro Lieovu algebru byla vymyšlena A.I. Shirshovem v roce 1962, ale jeho práce zůstala dlouho ukryta pod pláštěm tehdejšího Sovětského svazu.

1.2.2 CO JE TO GRÖBNEROVA BÁZE

Z laického pohledu, bez uvažování znalosti potřebného matematického aparátu, můžeme říct, že pro danou soustavu polynomiálních rovnic je Gröbnerova báze speciální bází, obsahující všechna řešení původní soustavy. Navíc se tato báze vyznačuje některými užitečnými vlastnostmi.

Můžeme ji vnímat jako nelineární zobecnění:

- Euklidovského algoritmu pro výpočet největšího společného dělitele
- Gaussovy eliminace
- lineárního programování

a to vše pro polynomy ve více proměnných.

ELEMENTÁRNÍ PŘÍKLAD

Pro lepší představu se podívejme na následující soustavu rovnic jedné proměnné $F = \{f_1, f_2\} \subset \mathbb{R}[x]$ a pokusme se najít řešení této soustavy

$$\begin{aligned} f_1: & -2x + x^3 - x^2 = 0 \\ f_2: & -1 + x^2 = 0 \end{aligned}$$

Hledáme takovou množinu reálných čísel, která je tvořena společnými kořeny obou polynomů, neboli hledáme kořeny největšího společného dělitele (n.s.d.) obou polynomů. Víme, že n.s.d. bude generátorem řešení soustavy rovnic F ve tvaru polynomu minimálního stupně. Rozhodujícím faktorem při dělení polynomů je uspořádání jednotlivých členů v polynomu. V případě polynomů jedné proměnné, existuje pouze jedno možné uspořádání, a to podle stupně členů polynomu.

Soustavu F před procesem dělení uspořádáme

$$\begin{aligned} f_1: & x^3 - x^2 - 2x = 0 \\ f_2: & x^2 - 1 = 0 \end{aligned}$$

Protože pracujeme s polynomy v jedné proměnné, můžeme pro nalezení n.s.d. využít známý Euklidův algoritmus:

vstup: a, b : $st(a) \geq st(b)$

výstup: největší společný dělitel - *nsd*

krok 1: vypočti zbytek po dělení $mod := a \setminus b$

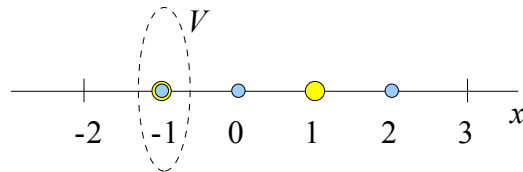
krok 2: pokud $mod \neq 0$, potom $a := b, b := mod$, a jdi zpět na krok 1.

potom $mod = 0$ potom $nsd = b$, konec algoritmu.

V našem příkladu dostáváme

$$\begin{array}{l} a: x^3 - x^2 - 2x \\ b: \sqrt{x^2 - 1} \\ \text{mod: } x+1 \end{array} \rightarrow \begin{array}{l} a: x^2 - 1 \\ b: \sqrt{x+1} \\ \text{mod: } 0 \end{array} \Rightarrow \text{nsd} := b = x+1$$

Kořen výsledného polynomu tvoří tzv. algebraickou varietu $V(f_1, f_2)$.



Obrázek 1.5: Algebraická varieta soustavy F .

Gröbnerova báze popisuje podobnou strukturu pro polynomy ve více proměnných. Generátory GB generují množinu polynomů, jejichž společnými kořeny je varieta V . Pokud bychom aplikovali teorii GB na tento jednoduchý příklad dvou polynomů jedné proměnné, dostali bychom samozřejmě totožný výsledek, $G = \{x+1\}$. Avšak na rozdíl od Euklidovského algoritmu není Gröbnerova báze teoreticky omezena ani počtem proměnných, ani počtem rovnic. Navíc v oblasti polynomů více proměnných přináší další užitečné vlastnosti.

APLIKAČNÍ PŮSOBNOST GRÖBNEROVY BÁZE

Hlavní silou GB je možnost řešit (a to i obecně) soustavy polynomiálních rovnic o více proměnných. Definovat, zda řešení existují, určit jejich počet a usnadnit jejich vyčíslení. Najít bázi generátorů těchto řešení.

Mezi problémy, řešitelné pomocí GB , patří:

- lineární diofantické rovnice s polynomiálními koeficienty ("syzigies")
- Hilbertovy funkce
- algebraické vztahy mezi polynomy
- implicitizace
- problém eliminace proměnných
- inverzní polynomiální zobrazení
- určení zda f je prvkem daného ideálu I

Na poli teorie systémů se GB uplatňuje zejména pro:

- návrh FIR/IIR filtrů
- faktorizace polynomiálních matic více proměnných
- test na stabilitu, říditelnost, pozorovatelnost, detekovatelnost, dosažitelnost
- optimální řízení
- řešení Cauchyova problému pro diskrétní systémy

POSTUP APLIKACE GRÖBNEROVY BÁZE

Necht' je dána soustava polynomiálních rovnic F v $k[x_1, \dots, x_n]$ popisující daný problém.

Obecná strategie je dána následujícím postupem:

- Transformujeme F na jinou soustavu G , nazvanou *Gröbnerova báze*, která se vyznačuje určitými užitečnými vlastnostmi.
- F a G jsou ekvivalentní, tzn. generují stejnou varietu, stejné řešení.
- Díky „užitečným“ vlastnostem G , je mnoho problémů těžko řešitelných na množině F „snadno“ řešitelných na množině G .
- Nalezené řešení v G převedeme zpět do F .

NALEZENÍ GRÖBNEROVY BÁZE (VÝPOČETNÍ A ČASOVÉ NÁROKY)

Nalézt GB není vždy snadné. Výpočetní náročnost a z ní vyplývající časová náročnost závisí na mnoha aspektech, se kterými se seznámíme dále. Nicméně algoritmy pro její nalezení jsou neustále zdokonalovány [36],[48], příloha A. Zde pouze poznamenejme, že od prvního algoritmu pro nalezení GB se čas výpočtu pomocí nových algoritmů zkrátil mnohonásobně.

GRÖBNEROVA BÁZE A TEORIE ŘÍZENÍ

Mnoho problémů v teorii řízení je založeno na popisu pomocí polynomiálních rovnic a jejich řešení. Samozřejmě mnoho těchto problémů je možné řešit i dalšími postupy. Ale v některých případech je řešení na původní množině F tak složité nebo téměř neuchopitelné a GB je jednou z mála metod, která umožňuje nalézt odpovědi na otázky typu zda řešení vůbec existuje, kolik jich je a usnadnit jejich vyčíslení [7],[90].

1.3 HLAVNÍ CÍLE A VÝSLEDKY PRÁCE

Tato práce se soustřeďuje na druhou úroveň plánování pohybu, tedy plánování pohybu podél geometrické cesty. V práci se omezíme na jednu z hlavních úloh robotiky a produkčních systémů, plánování pohybu se striktním respektováním všech pohybových omezení. Některé výsledky práce vykazují vlastnost časové optimality, jejíž podmínky jsou využívány jako kritérium kvality pro generování pohybu. Časová optimalita/sub-optimalita je sekundárním výsledkem plánovacího algoritmu. V kapitole 3 bude vysvětleno, proč bylo zvoleno právě toto kritérium kvality.

Pro tuto práci byly vytyčeny dva základní cíle.

Prvním cílem bylo seznámit se s relativně novou teorií z oblasti algebraické geometrie, teorií Gröbnerovy báze. Ta umí pracovat s polynomy více proměnných a umožňuje řešit polynomiální soustavy rovnic. Velkou předností je algoritmizovatelnost úloh, které se dají na problém řešení polynomiálních soustav převést.

První část práce se věnuje úvodu do této problematiky. Některé novější výsledky, využitelné pro hlubší pochopení teorie, jsou uvedeny v apendixu A. Algoritmy pro práci s Gröbnerovou bází je dnes možné nalézt a využít ve všech matematických nástrojích jako jsou Maple, Mathematica, Macsyma, Magma a Derive. Sofistikovanějšími nástroji v oblasti výpočetní algebry jsou například

CoCoA, Singular, Macaulay. Tento výčet pouze dokazuje, že GB je významným výsledkem v oblasti výpočetní algebry. V některých výše zmiňovaných nástrojích je při hledání řešení soustav polynomiálních rovnic automaticky využíváno Gröbnerovy báze, aniž by to uživatel věděl.

Druhým cílem bylo vytvořit generátor řídicích trajektorií pro úlohy řízení pohybu. Hlavní požadavky na algoritmus byly:

- generování trajektorií polohy, rychlosti, zrychlení
- respektování všech pohybových omezení, včetně omezení na strmost zrychlení
- možnosti napojování dílčích trajektorií
- možnost spojitého přechodu mezi libovolnými stavy v prostoru $[s(t), v(t), a(t)]$
- nízká výpočetní náročnost
- nezávislost na vzorkovací frekvenci
- optimalizace některého kritéria kvality

V druhé části práce je proveden detailní rozbor řešení úlohy plánování pohybu s výše zmíněnými požadavky. Pro kritérium kvality byla zvolena časová optimalita, ze které apriori vyplývá struktura řízení. To umožňuje zjednodušit úlohu ve smyslu lepší algoritmickeovatelnosti s využitím algebraických technik GB .

Hlavním výsledkem práce je analytický algoritmus, pojmenovaný $GBAVS$ (*General/Gröbner BAVS*²), který pro konstantní hodnoty všech omezení generuje spojitě trajektorie polohy, rychlosti a zrychlení pro pohyb mezi libovolnými stavy. Algoritmus pracuje vzhledem ke konstantním omezením i v takzvaných singulárních bodech nebo obloucích, kde některé standardní algoritmy selhávají [112]. Dnes je tento algoritmus již implementován do řídicího softwaru REX jako jeden z algoritmů pro *Open loop motion control*, vyvíjeném na našem pracovišti.

Hlavní přednosti algoritmu $GBAVS$:

- analytické řešení problému v explicitním tvaru
- nezávislost na vzorkovací frekvenci a malá paměťová náročnost
- možnost spojitého přechodu mezi libovolnými stavy v prostoru $[s(t), v(t), a(t)]$
- respektování omezení $|\dot{a}(t)| \leq BM$ (*Jerk*), $-DM \leq a(t) \leq AM$, $|v(t)| \leq VM$
- možnost hladkého napojování řídicích trajektorií v poloze $s(t)$ i rychlosti $v(t)$
- možnost přeplánování trajektorie během pohybu systému
- časová optimalita přechodu mezi stavy vzhledem k uvažovaným omezením
- algoritmus neobsahuje žádné cykly ani jiné výpočetní smyčky, jejichž výpočetní doba závisí na přesnosti hledaného výsledku (například metody monte-carlo nebo metody střelby).

Dalším výsledkem je nalezení rozdělovací variety pro plánování pohybu z konstantní do konstantní rychlosti. Ta umožňuje apriori rozhodnout, která ze dvou přípustných strategií řízení obsahuje hledané řešení, a tedy dvojnásobně zkrátit čas výpočtu algoritmu $GBAVS$.

Poslední část práce se věnuje úloze plánování pohybu s proměnnými omezeními podél specifikované cesty. Ukazuje se, že tento problém nelze obecně řešit analytickými prostředky a je třeba využít numerických metod. Cílem bylo nalézt výpočetně nenáročné procedury, jejichž doba výpočtu nebude závislá na přesnosti výsledku. Pro generátor respektující omezení na rychlost a zrychlení byly nalezeny jednoduché algoritmy využívající analytických postupů, které je možné využít i v real-time aplikacích. Doba výpočtu je závislá pouze na množství vzorků limitních křivek. Algoritmy opět neobsahují žádné cykly.

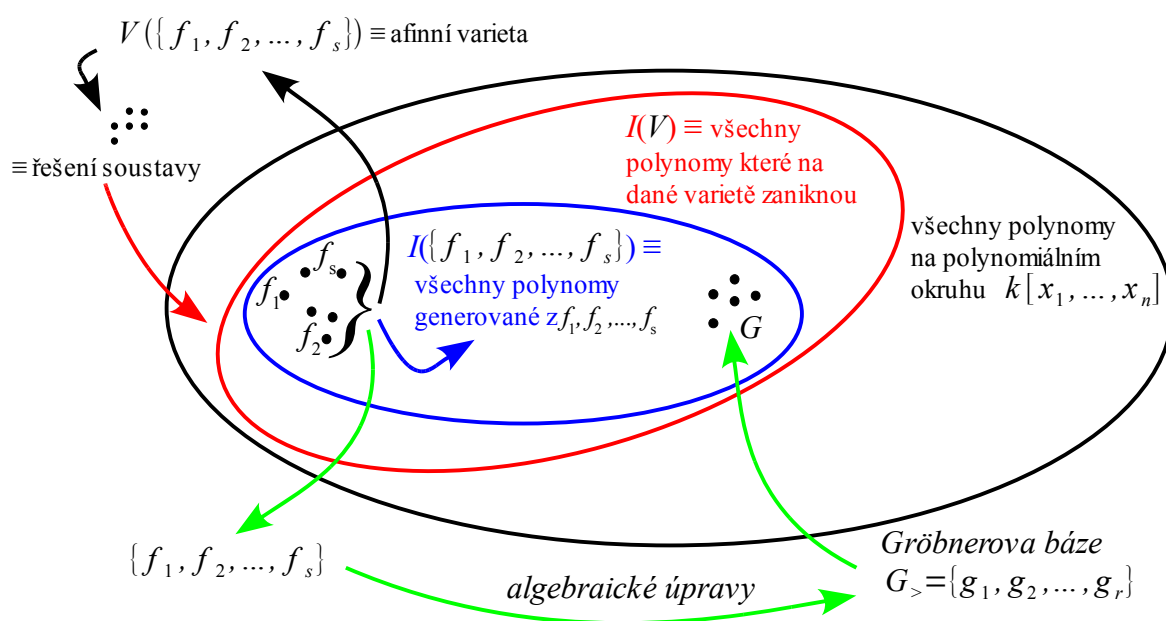
² BAVS je označení standardního generátoru pohybu, využívajícího řetězec integrátorů a t-optimální strategii řízení.

There is nothing more practical than a good theory.

L. Boltzmann

2 GRÖBNEROVA BÁZE - PŘÍPRAVNÁ KAPITOLA

Tato kapitola by měla tvořit ucelený úvod do teorie Gröbnerovy báze, jakožto součást výpočetní algebraické geometrie a komutativní algebry. Na důkladné porozumění budeme potřebovat něco málo z teorie komutativní algebry. Zaměříme se na ideály na polynomiálním okruhu a afinní variety (křivky, plochy a objekty vyšší dimenze). Zavedeme pojmy jako *uspořádání monomů*, *S-polynom*, *faktorový okruh*. Zobecníme dělení polynomů jedné proměnné na polynomy ve více proměnných. Vysvětlíme pojem Gröbnerova báze a redukovaná Gröbnerova báze. Představíme *Buchbergerův algoritmus* pro její nalezení a seznámíme se s některými vylepšeními, jejich klady a zápory. Ukážeme na některé aplikace v oblasti výpočetní algebraické geometrie. Také se podíváme na vztah Gröbnerovy báze s dimenzí variet. Úvod do této problematiky bude doprovázen příklady a odkazy na publikace s detailnějším zpracováním. Vzhledem k obsáhlosti této problematiky je nutné uvažovat mnoho základních pojmů za známé. Pro úplné porozumění textu je tedy nutná základní znalost z lineární algebry a teorie funkcí jedné proměnné. Úvod do této oblasti lze nalézt například v [82].



Obrázek 2.1: Vztah ideálu, variety a Gröbnerovy báze, převzato z [74].

2.1 POLYNOM, VARIETA, IDEÁL

Nejprve je třeba zavést základní elementy algebraické geometrie, jako je polynom, ideál a varieta nad komutativním tělesem komplexních čísel. Pro zjednodušení zápisu budeme komutativní těleso dále nazývat pouze tělesem.

2.1.1 AFINNÍ PROSTOR

Definice 2.1. *Nechť je dáno těleso k a kladné celé číslo n . **Afinním prostorem** nad k je množina definována jako $k^n = \{(c_1, \dots, c_n) : c_1, \dots, c_n \in k\}$.*

Příkladem může být *afinní plocha* \mathbb{R}^2 .

Definice 2.2. *Monom \mathbf{x}^a v okruhu $k[x_1, \dots, x_n]$ je definován jako součin ve tvaru $\mathbf{x}^a = x_1^{a_1} \cdot x_2^{a_2} \cdot \dots \cdot x_n^{a_n}$, kde všechny exponenty jsou nezáporná celá čísla. **Celkový stupeň** monomu je dán součtem exponentů $\sum a_i, i=1, 2, \dots, n$.*

Definice 2.3. *Polynom f v proměnných x_1, \dots, x_n je dán konečnou lineární kombinací monomů, zapsán jako $f = \sum_a c_a \mathbf{x}^a, c_a \in k$.*

Příkladem takové lineární kombinace více proměnných může být polynom v $\mathbb{C}[x, y, z]$ ve tvaru

$$f = x^3 y z^2 - 3 y^2 z + 6 y^3 z^3 \quad (2.1)$$

Pro úplnost ještě připomeneme pojem *algebraicky uzavřené* těleso. Podívejme se na následující rovnici

$$x^2 + 1 = 0, \quad (2.2)$$

definovanou nad tělesem reálných čísel \mathbb{R} . Vyřešením rovnice zjistíme, že kořeny již neleží v tělese \mathbb{R} , ale v tělese komplexních čísel \mathbb{C} . Je tedy možné nad tělesem reálných čísel definovat takovou rovnici, jejíž řešení není v tělese obsaženo. Nad tělesem \mathbb{C} toto již možné není, a proto je těleso \mathbb{C} nazýváno *algebraicky uzavřené*. Pro těleso $k = \mathbb{R} \subset \mathbb{C}$ je těleso \mathbb{C} tzv. *algebraickým uzávěrem* \bar{k} .

2.1.2 AFINNÍ VARIETY A JEJICH PARAMETRIZACE

Definice 2.4. *Nechť je množina polynomů $F = \{f_1, \dots, f_r\}$ definována v $k[x_1, \dots, x_n]$. Potom *afinní varietou* $V(F)$ nazveme množinu všech řešení*

$$V(F) = \{(c_1, \dots, c_n) \in \bar{k}^n : f_i(c_1, \dots, c_n) = 0 \text{ pro všechna } f_i \in F\}. \quad (2.3)$$

Příkladem variety může být varieta definovaná soustavou polynomiálních rovnic, obrázek 2.3a:

$$V(F) = (f_1, f_2) = (y - x^2, z - x^3) \subset \mathbb{R}[x, y, z]. \quad (2.4)$$

Zde je na místě položit si několik základních otázek:

- Lze určit, zda je varieta neprázdná, tj. soustava má řešení?
- Lze určit, zda je varieta konečným objektem a když ano, lze nalézt všechna řešení?
- Lze určit dimenzi variety?

Odpověď na všechny předchozí otázky je kladná a nástrojem, který nám tuto odpověď poskytne, je Gröbnerova báze, což uvidíme později.

PARAMETRIZACE VARIETY

Parametrizace variety, neboli převedení geometrického (implicitního) popisu do „více algebraického“ je vhodné, pokud chceme nějakým způsobem popsat (parametrizovat) všechna řešení. Pokud má varieta nekonečně mnoho řešení, nelze je jednoduše vypsat a je nutné využít parametrizaci. To bohužel není možné vždy, viz [29].

Parametrizaci variety (2.4) provedeme zavedením parametru $x=t$ ve tvaru

$$V(x-t, y-t^2, z-t^3) \subset \mathbb{R}[t, x, y, z] \quad (2.5)$$

Dostali jsme vektor $r(t) = (t, t^2, t^3)$. Nyní fixováním parametru t na konkrétní hodnotu dostaneme konkrétní bod na varietě V . Můžeme také parametrizovat plochu tečen příslušnou dané varietě V . Víme, že směr tečného vektoru ke křivce v daném bodě $r(t)$ je dán derivací $r'(t) = (1, 2t, 3t^2)$. Parametrizaci plochy tečen je tedy vektor

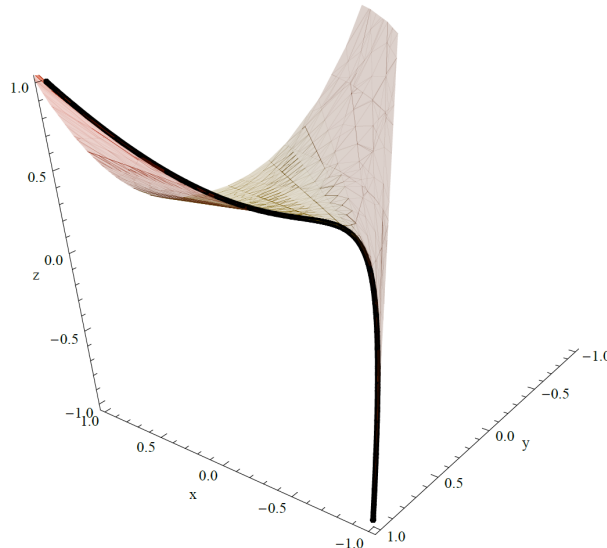
$$r(t) + ur'(t) = (t+u, t^2+2tu, t^3+3t^2u), \quad (2.6)$$

kde u je další parametr, kterým se pohybujeme po ploše, obrázek 2.2.

Vyloučením parametrů t a u dostaneme implicitní popis plochy tečen ve tvaru

$$-4x^3z + 3x^2y^2 - 4y^3 + 6xyz - z^2 = 0. \quad (2.7)$$

Parametrizace variet a jejich tečných ploch (nadploch) je přínosná zejména v popisu křivek v počítačové technice. Takzvané Bezierovy nebo B-spline křivky [80] využívající parametrizaci jsou neoddelitelnou součástí vývojových a designových nástrojů typu CAD. Také jsou velmi často využívány pro plánování pohybu.



Obrázek 2.2: Varieta $V(F)$ a její tečný prostor.

Nyní se znovu podíváme na varietu (2.4). Víme, že všechny společné kořeny funkcí $\{y-x^2, z-x^3\}$, které náleží varietě V , jsou předmětem našeho zájmu. Množina bodů tvořící varietu není ale jednoznačně definována soustavou polynomiálních rovnic $F = \{f_1, \dots, f_r\}$. Stejnou varietu V generuje například i soustava rovnic

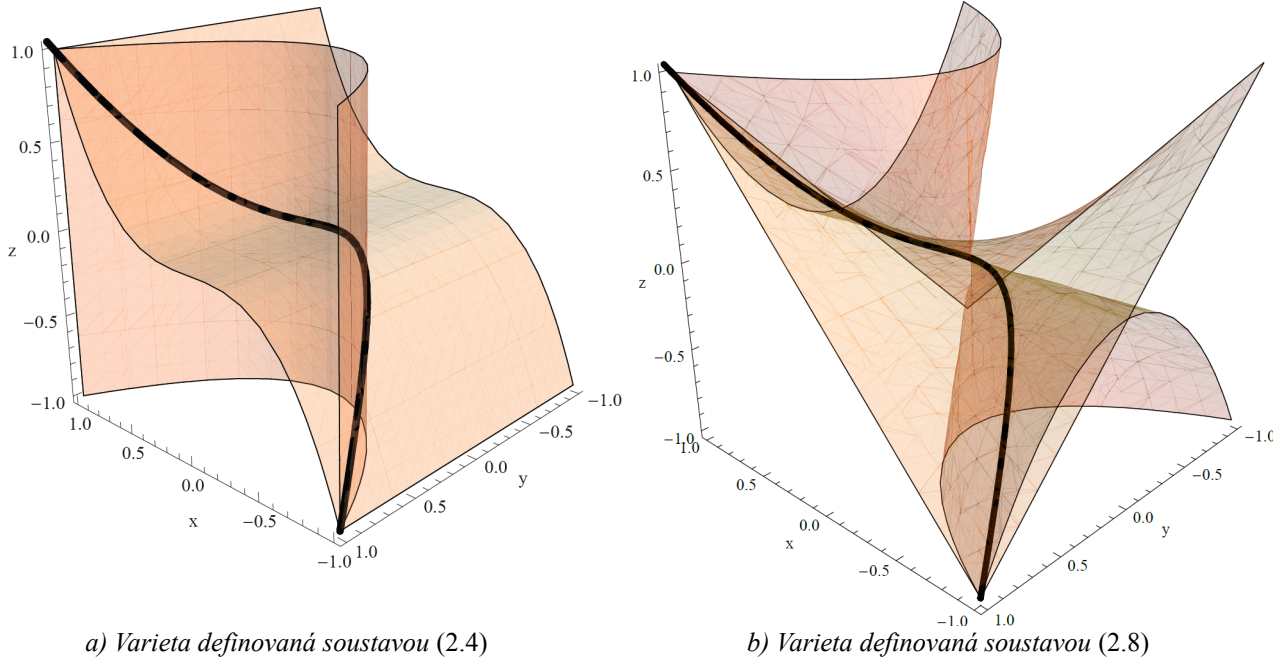
$$F' = \{z - xy, y^2 - xz\}, \quad (2.8)$$

viz totožná parametrizace, obrázek 2.3.

Okamžitě vyvstanou otázky:

- Jsou zde nulové i další funkce?
- Jak je můžeme najít?
- Existuje nějaká báze těchto funkcí?

Pro odpověď na výše uvedené otázky budeme potřebovat další pojem algebry, který nyní zavedeme.



Obrázek 2.3: Totožné variety generované průnikem odlišných afinních ploch.

2.1.3 IDEÁL

Definujeme základní algebraický objekt.

Definice 2.5. Podmnožina $I \subset k[x_1, \dots, x_n]$ je **ideálem** pokud splňuje následující vlastnosti:

1. $0 \in I$.
2. Jestliže $f, g \in I$, pak $f + g \in I$.
3. Jestliže $f \in I$ a $p \in k[x_1, \dots, x_n]$, pak $p \cdot f \in I$.

Neformálně tedy, jestliže F je nějaká množina polynomů $\{f_1, \dots, f_r\}$, potom ideál, generovaný touto množinou, je množina $\langle F \rangle$ obsahující všechny lineární kombinace

$$\langle F \rangle = \{p_1 f_1 + \dots + p_r f_r : f_1, \dots, f_r \in F \text{ a } p_1, \dots, p_r \in k[x_1, \dots, x_n]\}. \quad (2.9)$$

Klasickým příkladem ideálu je ideál $\langle f_1, \dots, f_r \rangle \subset k[x_1, \dots, x_n]$ generovaný konečnou množinou polynomů $f_1, \dots, f_r \in k[x_1, \dots, x_n]$. Pokud nahradíme množinu F (2.4) jinou množinou F' (2.8), která generuje stejný ideál, varieta $V(F)$ se nezmění. To je velmi významný fakt potvrzující, že varieta je definována ideálem, nikoli soustavou rovnic. Můžeme tedy přecházet mezi generátory ideálu, aniž bychom ovlivnili jeho varietu.

Věta 2.1 (Hilbert's Basis). Každý ideál $I \subset k[x_1, \dots, x_n]$ má konečnou množinu generátorů, neboli $I = \langle g_1, \dots, g_r \rangle$, kde $g_1, \dots, g_r \in I$.

Věta je nepostradatelnou součástí teorie Gröbnerovy báze, která je speciálním případem báze množiny generátorů. Důkaz lze nalézt v [29].

Definice 2.6. *Nechť $V \subset k^n$ je afinní varietou. Potom zavedeme objekt $I(V) = \{f \in k[x_1, \dots, x_n] : f(c_1, \dots, c_n) = 0 \text{ pro všechny } (c_1, \dots, c_n) \in V\}$.*

Lemma. *Jestliže $V \subset k^n$ je afinní varietou, potom $I(V) \subset k[x_1, \dots, x_n]$ je ideál a budeme jej nazývat ideál variety V .*

Poznámka. Ideál variety $I(V)$ je největší podmnožina $k[x_1, \dots, x_n]$ taková, že všechny polynomy této podmnožiny zaniknou na varietě V .

Vraťme se ještě jednou k varietě (2.4) a podívejme se na vztah mezi ideálem generovaným polynomy $\langle F \rangle$ a ideálem generovaným varietou $I(V)$. Dokažme, že $I(V) \equiv \langle y - x^2, z - x^3 \rangle$, neboli, že množina lineárních kombinací obou polynomů popisuje stejnou množinu jako ideál jejich variety, a je tedy největší podmnožinou báze funkcí.

Nejprve dokažme, že $\langle F \rangle \subset I(V)$. Předpokládejme nějaký polynom $f(x, y, z)$, který je lineární kombinací f_1, f_2 , a tedy $f \in \langle F \rangle$. Na varietě $V(f_1, f_2)$ zaniknou oba polynomy f_1, f_2 , a tedy i jejich lineární kombinace, tzn. $f \in I(V(f_1, \dots, f_r))$. Z variety (2.4) víme, že ačkoli polynomy nemusí být báze funkcí variety, generují varietu, a tedy i její ideál. Protože oba polynomy generují také ideál $\langle F \rangle$, musí platit

$$h_1(y - x^2) + h_2(z - x^3) \in I(V) \Rightarrow \langle y - x^2, z - x^3 \rangle \subset I(V).$$

Na dokázání opačné inkluze $I(V) \subset \langle F \rangle$ využijeme parametrizaci (2.5). Musíme dokázat, že množina lineárních kombinací obou polynomů generujících varietu (řešení) je prvkem ideálu $\langle F \rangle$. Musí tedy platit, že oba polynomy generující varietu dělí polynom $f(x, y, z)$ beze zbytku, neboli $\bar{f}^{(f_1, f_2)} = 0$. Využitím parametrizace (2.5) dostaneme

$$\begin{aligned} f(t, t^2, t^3) &= h_1(t^2 - t^2) + h_2(t^3 - t^3) + r \\ f(t, t^2, t^3) &= h_1 \cdot 0 + h_2 \cdot 0 + r. \end{aligned}$$

Vidíme, že polynom f bude beze zbytku redukován právě pokud bude $r = 0$. Tím je důkaz hotov.

Tímto postupem lze ověřit, zda je daný polynom f prvkem ideálu I . Tento postup bohužel závisí na parametrizaci, která, jak již bylo uvedeno, není vždy možná. Naštěstí je možné rozhodnout i bez použití parametrizace, a to právě využitím Gröbnerovy báze a zobecněného dělicího algoritmu pro polynomy více proměnných, což uvidíme dále. Závěrem je přirozené se zeptat, zda rovnost

$$I(V(f_1, \dots, f_r)) \stackrel{?}{=} \langle f_1, \dots, f_r \rangle \quad (2.10)$$

platí obecně. Odpovíme jednoduchým příkladem.

Spočtěme $I(V(x^2, y^2))$. Varieta $V(x^2, y^2) = \{(0, 0)\}$. Největší množinou generující ideál počátku $(0, 0)$ je ale $\{x, y\}$, neboli

$$I(V(x^2, y^2)) = \langle x, y \rangle. \quad (2.11)$$

Uvědomíme-li si, že například $x \notin \langle x^2, y^2 \rangle$, neboť lineární kombinace $h_1 x^2 + h_2 y^2$ nemůže být nižšího řádu než 2, je tedy zřejmé, že $I(V(x^2, y^2)) \neq \langle x^2, y^2 \rangle$.

Z výše uvedeného vyplývá, že rovnost (2.10) obecně neplatí. Hilbertův *Nullstellensatz*, se kterým se blíže seznámíme v podkapitole 2.5, dává přesnou odpověď na vztah mezi ideálem generovaným polynomy a ideálem generovaným varietou.

2.2 USPOŘÁDÁNÍ A REDUKCE POLYNOMŮ VÍCE PROMĚNNÝCH

Z předchozí podkapitoly vyplývá, že reprezentace ideálu ve smyslu generátorů není jednoznačná. Pro nalezení báze s určitými algoritmickými vlastnostmi je zapotřebí sestavit obecný algoritmus pro dělení polynomů více proměnných. Nejprve je ale třeba teorie o uspořádání členů v polynomech více proměnných, podobně jako je tomu u polynomů jedné proměnné, kde je využíváno lexikografického uspořádání podle stupně

$$1 < x < x^2 < \dots < x^n < x^{n+1} < \dots$$

2.2.1 USPOŘÁDÁNÍ

U polynomů více proměnných půjde o uspořádání monomů přičemž možností uspořádání je nekonečně mnoho. Nicméně některá uspořádání jsou určitým způsobem výhodnější a vhodná v daném kontextu pro konstrukci Gröbnerovy báze. Dvě z nejčastěji používaných uspořádání jsou lexikografické uspořádání (Lex Order) a uspořádání, které uvažuje také celkový stupeň monomu (Graded Reverse Lex Order).

„ČISTÉ“ LEXIKOGRAFICKÉ USPOŘÁDÁNÍ (LEX, PLEX)

Definice 2.7. *Nechť $\mathbf{a}=(a_1, \dots, a_n)$ a $\mathbf{b}=(b_1, \dots, b_n) \in \mathbb{Z}_{\geq 0}^n$. Řekneme, že $\mathbf{a}_{lex} > \mathbf{b}$ pokud v rozdílovém vektoru $\mathbf{a}-\mathbf{b} \in \mathbb{Z}_{\geq 0}^n$ je první nenulový prvek zleva kladný. Potom zapíšeme vztah mezi monomy takto $\mathbf{x}^{\mathbf{a}}_{lex} > \mathbf{x}^{\mathbf{b}}$.*

Například: $x^3 y^3 z_{lex} > x^3 y^2 z^4$.

Možností, jak seřadit monomy, je mnoho a někdy je obtížné najít výhodné uspořádání pro snazší výpočet Gröbnerovy báze. Charakteristickým znakem tohoto uspořádání ve výpočtu GB je, že dochází k eliminaci proměnných v polynomech GB . Neboli, řešení soustavy je možno nalézt postupnou substitucí, stejně tak jako v Gaussově eliminaci. GB garantuje nalezení všech kořenů. „Čisté“ lexikografické uspořádání je považováno některými autory jako neefektivní [83],[36], dávající příklady problémů kde nalezení GB je nezdolné. Může být ale použito reverzního lexikografického uspořádání podle stupně k nalezení GB a poté pomocí algoritmu *Gröbner walk*, viz příloha A, převedeno na GB danou v lexikografickém uspořádání.

REVERZNÍ LEXIKOGRAFICKÉ USPOŘÁDÁNÍ PODLE STUPNĚ (GREVLEX)

Definice 2.8. *Nechť $\mathbf{a}=(a_1, \dots, a_n)$ a $\mathbf{b}=(b_1, \dots, b_n) \in \mathbb{Z}_{\geq 0}^n$. Řekneme, že $\mathbf{a}_{grevlex} > \mathbf{b}$ pokud $|\mathbf{a}| = \sum_{i=1}^n a_i > |\mathbf{b}| = \sum_{i=1}^n b_i$, nebo $|\mathbf{a}| = |\mathbf{b}|$ a v rozdílovém vektoru $\mathbf{a}-\mathbf{b} \in \mathbb{Z}_{\geq 0}^n$ je první nenulový prvek zprava záporný. Potom zapíšeme vztah mezi monomy takto $\mathbf{x}^{\mathbf{a}}_{grevlex} > \mathbf{x}^{\mathbf{b}}$.*

Například: $x^3 y^3 z <_{grevlex} x^3 y^2 z^4$.

Výpočet GB s tímto uspořádáním je zpravidla nejrychlejší. *Grevlex* uspořádání bývá výchozím typem uspořádání všech algoritmů zabývajících se přechodem od GB s respektováním *grevlex* uspořádání k ostatním typům.

Podívejme se ještě na příklad uspořádání polynomu $f = x^3 y^2 + 3 x^3 y - 5 x y^3 - y^5 \in k[x, y]$ a seznáme se s terminologií využívanou pro uspořádání monomů v polynomech.

Definice 2.9. Necht' je dán nenulový polynom $f = \sum_a c_a \mathbf{x}^a$, $c_a \in k$ a způsob uspořádání monomů $>$. Potom definujeme následující objekty:

multistupeň f je

$$\text{multideg}(f) = \max_{\text{v.r.t. } >} (\mathbf{a} \in \mathbb{Z}_{\geq 0}^n : c_a \neq 0)$$

vedoucí koeficient f je

$$LC(f) = c_{\text{multideg}(f)} \in k$$

vedoucí (hlavní) monom f je

$$LM(f) = \mathbf{x}^{\text{multideg}(f)}$$

vedoucí (hlavní) člen f je

$$\text{in}_<(f) = LC(f) \cdot LM(f).$$

Vedoucí členy v polynomu budeme dále v textu značit podtržením. Odlišným uspořádáním tedy dostaneme:

$$f = \underline{x^3 y^2} + 3x^3 y - 5x y^3 - y^5, \quad x >_{\text{lex}} y, \quad \text{kde } \text{multideg}(f) = (3, 2), \quad LC(f) = 1 \\ LM(f) = x^3 y^2, \quad \text{in}_<(f) = x^3 y^2$$

$$f = -\underline{y^5} - 5x y^3 + x^3 y^2 + 3x^3 y, \quad y >_{\text{lex}} x, \quad \text{kde } \text{multideg}(f) = (0, 5), \quad LC(f) = -1 \\ LM(f) = y^5, \quad \text{in}_<(f) = -y^5$$

$$f = -\underline{y^5} + x^3 y^2 - 5x y^3 + 3x^3 y, \quad y >_{\text{grevlex}} x, \quad \text{kde } \text{multideg}(f) = (0, 5), \quad LC(f) = -1 \\ LM(f) = y^5, \quad \text{in}_<(f) = -y^5$$

Dalšími využívanými typy uspořádání jsou například:

- **vážené uspořádání (wdeg)**
- **uspořádání podle absolutního stupně (grlex, tdeg)**
- **eliminační uspořádání (lexdeg)**
- **maticové uspořádání (matrix)**

Pro podrobnější popis těchto typů uspořádání odkážeme čtenáře na [67].

2.2.2 REDUKCE POLYNOMŮ VÍCE PROMĚNNÝCH

Nyní je třeba rozšířit klasický algoritmus dělení polynomů v $k[x]$ pro polynomy více proměnných, definovaných v $k[x_1, \dots, x_n]$. Cílem je vyjádřit polynom $f \in k[x_1, \dots, x_n]$ pomocí množiny polynomů $F = \{f_1, \dots, f_s\}$, $f_i \in k[x_1, \dots, x_n]$, $i \in \{1, \dots, s\}$, a to jako jejich lineární kombinaci a zbytek po dělení $\bar{f}^F \in k[x_1, \dots, x_n]$

$$f = a_1 f_1 + \dots + a_s f_s + \bar{f}^F \tag{2.12}$$

Základní idea procesu dělení je stejná jako v případě polynomů jedné proměnné. Necht' je dáno uspořádání členů $<$. Snažíme se vykrátit hlavní člen polynomu f tak, že například polynom f_2 vynásobíme nějakým vhodným monomem a odečteme od f . Polynom f může být redukován

pomocí f_2 , pokud $in_{<}(f_2)$ dělí $in_{<}(f)$. Potom dostáváme zobrazení

$$f \rightarrow f - \frac{in_{<}(f)}{in_{<}(f_2)} f_2$$

Příklad. Uvažujme polynomy

$$f = x^2 - y^3 \\ \{f_1, f_2\} = \{x - y^7 + y^4 + y^2, y^9 - 2y^6 - y^4 + y^3\}$$

s respektováním uspořádání $x_{lex} > y$.

Zajímá nás, zda polynom f je lineární kombinací polynomů $\{f_1, f_2\}$ a tedy může být beze zbytku redukován.

$$\begin{array}{r} x^2 - y^3 : (x - y^7 + y^4 + y^2, y^9 - 2y^6 - y^4 + y^3) \\ - \underline{x(x - y^7 + y^4 + y^2)} \\ \quad x y^7 - x y^4 - x y^2 - y^3 \\ - \underline{y^7(x - y^7 + y^4 + y^2)} \\ \quad \quad -x y^4 - x y^2 - y^3 - y^9 - y^{11} + y^{14} \\ \quad \quad \underline{y^4(x - y^7 + y^4 + y^2)} \\ \quad \quad \quad -x y^2 + y^{14} - y^3 + y^6 + y^8 - y^9 - 2y^{11} \\ \quad \quad \quad \underline{y^2(x - y^7 + y^4 + y^2)} \\ \quad \quad \quad \quad y^{14} - 2y^{11} - 2y^9 + y^8 + 2y^6 + y^4 - y^3 \\ \quad \quad \quad \quad - \underline{y^5(y^9 - 2y^6 - y^4 + y^3)} \quad (\text{dělení druhým polynomem}) \\ \quad \quad \quad \quad \quad -y^9 + 2y^6 + y^4 - y^3 \\ \quad \quad \quad \quad \quad \underline{1(y^9 - 2y^6 - y^4 + y^3)} \\ \quad \quad \quad \quad \quad \quad 0 \end{array}$$

Uspořádáním kofaktorů (tučné prvky) z procesu dělení dostáváme

$$f = (x + y^7 - y^4 - y^2) f_1 + (y^5 - 1) f_2 + 0.$$

Poznámka. Dělení polynomů o více proměnných je vždy konečný, ale nejednoznačný proces. Záleží na typu použitého uspořádání.

2.3 GRÖBNEROVA BÁZE

Mezi všemi možnými množinami generátorů, bázemi generujícími ideál I , se nachází speciální třída bází s požadovanými algoritmickými vlastnostmi, Gröbnerova báze.

Definice 2.10. Necht' je dáno uspořádání monomů. Řekneme, že konečná podmnožina $G = \{g_1, \dots, g_r\}$ ideálu I je **Gröbnerovou bází**, pokud platí:

1. $I = \langle g_1, \dots, g_r \rangle$
2. $in_{<}(I) = \langle in_{<}(g_1), \dots, in_{<}(g_r) \rangle$.

Věta 2.2. Soustava g_1, \dots, g_r je **Gröbnerovou bází** právě tehdy, když pro každou $f \in \langle g_1, \dots, g_r \rangle$ existuje $j \in \{1, \dots, r\}$ takové, že $in_{<}(g_j)$ dělí $in_{<}(f)$.

Příklad. Využijme předcházející větu a podívejme se, zda funkce $f_1 = -xy + y$, $f_2 = xy + 2y - x - 2$, generující ideál

$$I = \langle -xy + y, xy + 2y - x - 2 \rangle, \quad (2.13)$$

jsou vzhledem k uspořádání $<_{\text{grevlex}}$ přímo bázovými funkcemi GB .

Z definice GB vidíme, že funkce f_1, f_2 vzhledem k danému uspořádání netvoří GB . Neplatí totiž věta 2.2, ani definice 2.10. Například vedoucí člen polynomu $f = f_1 + f_2 = -x + 3y - 2$, který patří do ideálu I , nelze generovat vedoucími členy ideálu I , což je v rozporu s definicí. Žádný z vedoucích členů ideálu I nedělí vedoucí člen $-x$, což je v rozporu s větou 2.2. Gröbnerova báze a vedoucí ideál, generovaný vedoucími monomy, vypadají vzhledem k danému uspořádání následovně

$$\begin{aligned} G_{<_{\text{grevlex}}} &= \{x - 3y + 2, y^2 - y\}, \\ \text{in}_{<_{\text{grevlex}}}(I) &= \langle x, y^2 \rangle, \end{aligned} \quad (2.14)$$

kde již vše platí.

Další Gröbnerova báze pro stejný ideál i typ uspořádání, splňující všechny předpoklady je

$$G'_{<_{\text{grevlex}}} = \{x - 3y + 2, y^2 - y, -xy + y\}. \quad (2.15)$$

Tím zjišťujeme, že Gröbnerova báze není ani jednoznačná, ani nevykazuje požadavky minimality, jak slovo báze evokuje. Nicméně pohledem na obě báze je vidět, že až na třetí rovnici v (2.15) jsou obě báze shodné. Vedoucí ideál báze (2.15) obsahuje navíc člen $-xy$, ale ten lze generovat i vedoucím ideálem báze (2.14) a tedy se zdá, že báze (2.14) by mohla splňovat požadavky minimality a jednoznačnosti. Takovou bázi budeme nazývat *redukovanou Gröbnerovou bází* a podrobněji ji popíšeme v další podkapitole.

Poznámka. Gröbnerova báze, daná ideálem I a způsobem uspořádání, musí obsahovat nejmenší prvek ideálu s respektováním tohoto uspořádání. Pokud tomu tak není, žádný element Gröbnerovy báze nemůže redukovat nejmenší člen, což je v rozporu s definicí.

2.4 ALGORITMUS (BUCHBERGER)

Každý nenulový ideál I má Gröbnerovu bázi danou způsobem uspořádání členů. Gröbnerova báze tedy existuje vždy! Odlišné uspořádání může vést na odlišnou reprezentaci GB , a to někdy velmi výrazně. Ukázali jsme jak zjistit, zda jsou generátory ideálu vzhledem k uspořádání přímo generátory GB , či nikoli. Nyní se podívejme, jak GB nalézt. Budeme potřebovat další pojem algebry, stavební kámen Buchbergerova algoritmu pro nalezení GB .

2.4.1 S-POLYNOM

S-polynom $S(f, g)$ je polynom ve tvaru $uf - vg$, kde $f, g \in k[x_1, \dots, x_n]$ jsou polynomy a u, v jsou monomy vybrané takovým způsobem, že $u \text{in}_{<}(f)$ a $v \text{in}_{<}(g)$ jsou rovné nejmenšímu společnému násobku obou hlavních členů $\text{in}_{<}(f), \text{in}_{<}(g)$ a tedy v důsledku následného odčítání tyto hlavní členy zaniknou.

S-polynom můžeme chápat jako zobecnění Gaussovy eliminace.

$$\begin{aligned} f(x, y, z) &= 0 \\ g(x, y, z) &= 0 \\ f(x, y, z) &= 0 \\ g(x, y, z) &= 0 \\ u(x, y, z)f(x, y, z) - v(x, y, z)g(x, y, z) &= 0 \end{aligned}$$

Výpočet S-polynomu je jádrem Buchbergerova algoritmu pro nalezení GB. Nicméně definice a význam GB je nezávislý na S-polynomu, jak jsme již viděli v předchozí podkapitole.

Věta 2.3. (Hlavní věta teorie Gröbnerovy báze). *Soustava $F = \{f_1, \dots, f_s\}$ je Gröbnerovou bází právě tehdy, když pro každou dvojici $f_i, f_j \in F$ platí, že S-polynom je plně dělitelný soustavou F , neboli $\overline{S(f_i, f_j)}^F = 0$.*

Věta 2.4. (Algoritmus). *Nechť $I = \langle f_1, \dots, f_s \rangle \neq \{0\}$ je ideál generovaný polynomy f_1, \dots, f_s . Potom Gröbnerovu bází ideálu I můžeme nalézt v konečném počtu kroků následujícím algoritmem :*

vstup: $F = \{f_1, \dots, f_s\}$

výstup: Gröbnerova báze $G = \{g_1, \dots, g_t\}$ pro I , $F \subset G$

```

G := F
REPEAT
  G' = G
  FOR každou dvojici {p, q}, p ≠ q z G' DO
    S :=  $\overline{S(p, q)}^{G'}$ 
    IF S ≠ 0 THEN G := G ∪ {S}
UNTIL G = G'

```

Příklad (První GB). Pro daný ideál $I(f_1, f_2) = \langle x^2y + xy - 1, xy^2 + x^2 \rangle$ spočteme GB výše uvedeným algoritmem. Budeme hledat GB s respektováním lexikografického uspořádání $y_{lex} > x$.

$$\begin{aligned} G' &:= \{f_1, f_2\} \\ S(f_1, f_2) &= y(x^2y + xy - 1) - x(xy^2 + x^2) = xy^2 - y - x^3 \end{aligned} \quad (2.16)$$

$$\begin{aligned} &xy^2 - y - x^3 : (xy^2 + x^2, x^2y + xy - 1) \\ &\underline{-(xy^2 + x^2)} \\ &-y - x^3 - x^2 = \overline{S(f_1, f_2)}^{G'} \stackrel{\Delta}{=} f_3 \end{aligned} \quad (2.17)$$

$$\begin{aligned} G' &= \{f_1, f_2, f_3\} \\ S(f_2, f_3) &= (x^2y + x^2) - (xy)(y + x^3 + x^2) = x^2y + xy - 1 \end{aligned} \quad (2.18)$$

protože $S(f_2, f_3)$ a f_1 jsou ekvivalentní, je zřejmé, že

$$\begin{aligned} \overline{S(f_2, f_3)}^{G'} &= 0 \\ S(f_1, f_3) &= (-1)(x^2y + xy - 1) - (-x^2)(y + x^3 + x^2) = -xy + 1 + x^5 + x^4 \end{aligned} \quad (2.19)$$

$$\begin{aligned} & -xy + x^5 + x^4 + 1 : (f_3, f_1, f_2) \\ & \frac{x(y + x^3 + x^2)}{x^5 + 2x^4 + x^3 + 1} = \overline{S(f_1, f_3)}^{G'} \triangleq f_4 \end{aligned} \quad (2.20)$$

$$G' = \{f_1, f_2, f_3, f_4\}$$

$\overline{S(f_1, f_4)}^{G'} = 0$ $\overline{S(f_2, f_4)}^{G'} = 0$ $\overline{S(f_3, f_4)}^{G'} = 0$ a tedy G' je Gröbnerovou bází

$$G' = \{x^2y + xy - 1, xy^2 + x^2, y + x^3 + x^2, x^5 + 2x^4 + x^3 + 1\}. \quad (2.21)$$

2.4.2 REDUKOVANÁ GRÖBNEROVA BÁZE

Našli jsme Gröbnerovu bází G' ideálu I . Tato báze ovšem není v minimální formě, ekvivalentní s tvarem schodovité matice pro lineární soustavy. K určení, zda Gröbnerova báze je takzvanou plně redukovanou Gröbnerovou bází, nám pomůže následující věta.

Věta 2.5. *Gröbnerova báze G' je redukovanou Gröbnerovou bází G , pokud všechny polynomy g_i v bázi G' jsou monické a jsou plně redukovány bázemi $G' - \{g_i\}$, neboli žádný nevedoucí člen polynomů g_i není obsažen v $\text{in}_<(I)$.*

Poznámka. Lze ukázat, že redukovaná Gröbnerova báze, daná lexikografickým uspořádáním, s konečným počtem řešení obsahuje právě jeden polynom pouze jedné proměnné a to nejnižšího možného řádu.

Pokračování příkladu GB. Následující redukce potvrzují výše uvedenou větu 2.5.

$$\overline{f_1}^{(G' - \{f_1\})} = 0 \quad \overline{f_2}^{(G' - \{f_2\})} = 0 \quad \overline{f_3}^{(G' - \{f_3\})} = y + x^3 + x^2 \quad \overline{f_4}^{(G' - \{f_4\})} = x^5 + 2x^4 + x^3 + 1$$

Poslední dva polynomy se již nemohou více redukovat a jsou tedy prvky redukované GB. Druhá rovnice v bázi potvrzuje výše uvedený fakt o lexikografickém uspořádání a jeho vlivu na výskyt polynomu pouze jedné proměnné nejnižšího možného řádu. Hledanou redukovanou Gröbnerovu bází zapíšeme v následujícím tvaru

$$G = \{y + x^3 + x^2, x^5 + 2x^4 + x^3 + 1\}. \quad (2.22)$$

Poznámka. Ačkoli existuje nekonečně mnoho možností uspořádání monomů, existuje pouze konečný počet redukovaných Gröbnerovýchází. Tento zajímavý a netriviální fakt je popsán například v [21].

Pokud při redukci $\overline{f_1}^G, \overline{f_2}^G$ zachováme kofaktory a uspořádáme je do matice U , dostaneme zobrazení $U: G \rightarrow F$ ve tvaru

$$F = UG \quad (2.23)$$

a tedy

$$\begin{bmatrix} x^2y + xy - 1 \\ xy^2 + x^2 \end{bmatrix} = \begin{bmatrix} x^2 + x & -1 \\ xy - x^4 - x^3 & x^2 \end{bmatrix} \begin{bmatrix} y + x^3 + x^2 \\ x^5 + 2x^4 + x^3 + 1 \end{bmatrix}. \quad (2.24)$$

Transformaci $U: G \rightarrow F$ jsme určili z redukce \overline{F}^G . Můžeme nalézt také inverzní zobrazení $C: F \rightarrow G$ ve tvaru $G = CF$. Z procesu redukce S-polynomu (2.16) a (2.17) vidíme, že

$$f_3 = -y f_1 + (x+1) f_2. \quad (2.25)$$

Z redukce (2.19) a (2.20) S-polynomu $S(f_1, f_3)$ dostaneme $f_4 = (-1) f_1 + (x^2 + x) f_3$

a dosazením (2.25) za f_3 dostáváme

$$f_4 = (-1 - xy - x^2 y) f_1 + (x^3 + 2x^2 + x) f_2.$$

Nyní již můžeme sestavit takzvanou matici kofaktorů C , která transformuje původní soustavu F na její Gröbnerovu bázi

$$G = CF.$$

V našem případě dostáváme

$$\begin{bmatrix} y + x^3 + x^2 \\ x^5 + 2x^4 + x^3 + 1 \end{bmatrix} = \begin{bmatrix} -y & x+1 \\ -1 - xy - x^2 y & x^3 + 2x^2 + x \end{bmatrix} \begin{bmatrix} x^2 y + x y - 1 \\ x y^2 + x^2 \end{bmatrix} \quad (2.26)$$

Právě jsme tedy "RUČNĚ" spočetli první a zřejmě i poslední GB včetně obou transformací U a C . Všechny další báze už získáme pomocí matematických nástrojů s implementovaným Buchbergerovým algoritmem, jako například Maple, Mathematica, CoCoA, Singular, Macaulay a další.

2.4.3 KOREKTNOST ALGORITMU

Správnost algoritmu je založena na větě 2.3, jejíž důkaz patří k nejdůležitějším výsledkům v teorii Gröbnerovy báze. Celá síla Gröbnerovy báze je založena na této větě a jejím důkazu. Důkaz byl prvně uveden v doktorské práci Bruno Buchbergera a jeho moderní pojetí je možné nalézt v [18]. Výše uvedená věta platí pro libovolnou F , tedy i pro nekonečnou množinu. Buchberger také dokázal, že pro libovolné polynomy, pro které platí $n.s.n. = in_<(f_1)in_<(f_2)$, redukce $S(f_1, f_2)$ modulo f_1, f_2 je vždy rovna 0. Tento fakt nazval produkčním kritériem. Podíváme-li se například na varietu (2.4) a zavedeme odlišná uspořádání, potom podle kritéria dostáváme

$$\begin{aligned} F = (y - x^2, z - x^3), \quad x_{lex} > y_{lex} > z &\Rightarrow (-x^2)(-x^3) \neq x^3 \Rightarrow F \neq G \\ F = (y - x^2, z - x^3), \quad y_{lex} > z_{lex} > x &\Rightarrow (y)(z) \equiv yz \Rightarrow F \equiv G \end{aligned}$$

aniž bychom museli použít Buchbergerův algoritmus.

2.4.4 KONEČNOST ALGORITMU A DICKSON'S LEMMA

Konečnost algoritmu není zřejmá. Na začátku algoritmu je sice konečný počet dvojic, pro které musíme určit S -polynom, ale v průběhu výpočtu může nastat $S \neq 0$ a S bude přidáno do množiny generátorů. Intuitivně se může zdát, že počet polynomů G může stále narůstat. Ale lze ukázat, pomocí Dicksonova lemmatu, že tento proces se musí zastavit.

Definice 2.11. Ideál $I \subset k[x_1, \dots, x_n]$ je **monomický ideál**, jestliže existuje podmnožina $D \subset \mathbb{Z}_{\geq 0}^n$ taková, že I obsahuje všechny polynomy, které jsou dány konečným součtem ve tvaru $\sum_{a \in D} h_a \mathbf{x}^a$, kde $h_a \in k[x_1, \dots, x_n]$. Potom zapisujeme $I = \langle \mathbf{x}^a : a \in D \rangle$.

Definice říká, že každý polynom je generován svým monomickým ideálem. Ten je jednoznačně definován svými monomy. Následující věta říká, že každý monomický ideál je generován konečným počtem monomů. Z toho plyne, že množina generátorů Gröbnerovy báze nemůže stále narůstat a Buchbergerův algoritmus je konečný.

Dickson's Lemma. Monomický ideál $I = \langle \mathbf{x}^a : a \in D \rangle \subset k[\mathbf{x}]$ může být zapsán ve tvaru $I = \langle \mathbf{x}^{a(1)}, \dots, \mathbf{x}^{a(s)} \rangle$, kde $a(1), \dots, a(s) \in D$, neboli I má konečnou bázi.

2.4.5 SPECIALIZACE (GAUSS, EUKLEIDÉS)

Zajímavým faktem je, že řádkové úpravy Gaussovy eliminace na soustavě lineárních rovnic jsou speciální formou Buchbergerova algoritmu.

Příklad. Podívejme se na následující soustavu zapsanou v maticovém tvaru

$$\left(\begin{array}{ccc|c} 3 & -6 & -2 & 0 \\ 2 & -4 & 0 & 0 \\ 1 & -2 & -1 & 0 \end{array} \right). \quad (2.27)$$

Pomocí Gaussovy eliminace převedeme soustavu do tvaru

$$\left(\begin{array}{ccc|c} 1 & -2 & -1 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{array} \right) \quad (2.28)$$

a poté do tvaru schodovité matice

$$\left(\begin{array}{ccc|c} 1 & -2 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{array} \right). \quad (2.29)$$

Zápisem soustavy (2.27) v jazyce ideálů dostaneme

$$I = \langle 3x - 6y - 2z, 2x - 4y, x - 2y - z \rangle \subset k[x, y, z]$$

jehož Gröbnerova báze vzhledem k lexikografickému uspořádání $x >_{lex} y >_{lex} z$

$$G' = \{z, x - 2y - z\} \quad (2.30)$$

je ekvivalentní s (2.28) a redukovaná Gröbnerova báze

$$G = \{z, x - 2y\} \quad (2.31)$$

je ekvivalentní s (2.29). Dalším zjednodušením Buchbergerova algoritmu na pole jednočlenů dostáváme Euklidův algoritmus, s kterým jsme již pracovali v úvodu.

2.5 VLASTNOSTI A APLIKACE GRÖBNEROVY BÁZE VE VÝPOČETNÍ ALGEBRAICKÉ GEOMETRII

Tato kapitola stručně nastiňuje některé standardní problémy a úlohy v algebraické geometrii a na jednoduchých příkladech ukazuje využití *GB* pro řešení těchto úloh.

2.5.1 DIMENZE VARIETY A FAKTOROVÝ OKRUH

Gröbnerova báze G odhaluje geometrické vlastnosti variety, které nejsou zřejmé z původní soustavy F . Abychom mohli zjistit zda je varieta konečným objektem a kolik řešení původní soustava implikuje, nebo určit dimenzi variety, je nutné uvést další definici a zmínit několik důležitých vět. Ty přinášejí zajímavé souvislosti mezi Gröbnerovou bází a tzv. *faktorovým okruhem*,

viz dále. Také odpovídají na problém rovnosti (2.10).

Definice 2.12. *Nechť je dán ideál I na polynomiálním okruhu $k[x_1, \dots, x_n]$ a způsob uspořádání monomů $>$. Monom $x^a = x_1^{a_1} \dots x_n^{a_n}$ je nazýván **standardním**, pokud není obsažen v ideálu $\text{in}_<(I)$, generovaném vedoucími monomy ideálu I .*

Množina těchto *standardních monomů* neobsažených v $\text{in}_<(I)$ tvoří tzv. *faktorový okruh* $k[x_1, \dots, x_n]/I$ ideálu I . Z předešlého textu víme, že libovolný polynom $f \in k[x_1, \dots, x_n]$ je možné vyjádřit jako

$$f = \sum_{i=1}^l p_i g_i + \bar{f}^G, \quad g_i \in G, \quad p_i \in k[x_1, \dots, x_n],$$

kde G je redukovaná Gröbnerova báze.

Zbytek po dělení \bar{f}^G leží v $k[x_1, \dots, x_n]/I$ a obsahuje pouze *standardní monomy*, které nejsou dělitelné hlavními členy g_i . Množina všech těchto monomů tvoří *standardní bázi faktorového okruhu* a každý zbytek \bar{f}^G lze vyjádřit jako lineární kombinaci prvků tohoto okruhu.

PRÁZDNÁ VARIETA

Velmi často se zajímáme o takové soustavy rovnic, které mají konečný počet řešení. Varieta je potom konečným objektem. V takovém případě je i množina standardních monomů konečná, viz dále. Nejprve se ale podívejme na případ, kdy je varieta příslušného ideálu prázdnou množinou.

Věta 2.6 (The Weak Nullstellensatz). *Nechť k je algebraicky uzavřené těleso a necht' $I \subset k[x_1, \dots, x_n]$ je ideál splňující $V(I) = \emptyset$. Potom $I = k[x_1, \dots, x_n]$.*

Věta říká, že algebraický uzávěr na polynomiálním okruhu je postačující podmínkou, zajišťující, že pouze ideál, jehož varieta je prázdná, je sám celým polynomiálním okruhem. To pro okruh definovaný například nad tělesem reálných čísel již neplatí, protože například pro $I_1 = \langle 1 \rangle = \mathbb{R}[x]$, $I_2 = \langle 1+x^2 \rangle$ je průnik příslušných variet prázdnou množinou, $V(I_1) \cap V(I_2) = \emptyset$. Věta také odpovídá na otázku konzistence množiny polynomů generujících ideál I v \mathbb{C}^n . Varieta je prázdnou množinou, jestliže polynomy nemají žádné společné řešení. Podle věty 2.6 tato situace nastává pouze v případě, kdy ideál obsahuje jednotkový element, který může vygenerovat celý okruh. A tedy množina polynomů F je nekonzistentní tehdy a jen tehdy, pokud GB daná ideálem I generovaným množinou F obsahuje jednotkový element

$$V(F) = \emptyset \Leftrightarrow G = \{1\}.$$

VARIETA NULOVÉ DIMENZE

Jestliže polynomiální soustava F obsahuje konečný počet řešení, potom je varieta konečnou množinou, tzv. varietou nulové dimenze. Důležitou souvislostí je, že počet prvků příslušné standardní báze je také konečný a faktorový okruh tvoří vektorový prostor konečné dimenze.

Definice 2.13. Ideál I je ideálem nulové dimenze, jestliže příslušná varieta $V(I)$ je konečnou množinou.

Rozhodnout, zda ideál I je ideálem nulové dimenze, lze pomocí následující věty, která má přímou souvislost s faktorovým okruhem.

Věta 2.7. *Nechť $I \subset \mathbb{C}[x_1, \dots, x_n]$ je ideál takový, že každá proměnná x_i tvoří monom ve tvaru $x_i^{m_i}$, kde $m_i \in \mathbb{Z}^+$ pro každé i . Potom počet bodů variety je konečnou množinou a je dán nejvýše $m_1 \cdot m_2 \cdot \dots \cdot m_n$ body. Pokud navíc platí $I = I(V(I))$, potom je počet bodů variety roven $m_1 \cdot m_2 \cdot \dots \cdot m_n = \dim(\mathbb{C}[x_1, \dots, x_n]/I)$.*

Poznámka. Pro $n=1$ je věta 2.7 fundamentální větou algebry.

Následující věta upozorňuje na fakt, že nad algebraicky uzavřeným tělesem existuje pouze jedna možnost, kdy různé ideály definují jednu a tu samou varietu.

Věta 2.8 (Hilbert's Nullstellensatz). *Nechť k je algebraicky uzavřené těleso. Jestliže pro $f, f_1, \dots, f_s \in k[x_1, \dots, x_n]$ platí, že $f \in I(V(f_1, \dots, f_s))$, pak existuje přirozené číslo $m \geq 1$ takové, že $f^m \in \langle f_1, \dots, f_s \rangle$, a naopak.*

Definice 2.14. *Nechť $I \subset k[x_1, \dots, x_n]$ je ideál. Potom **radikál** I , označený jako \sqrt{I} je množina*

$$\{f : f^m \in I, \text{ pro libovolné celé číslo } m \geq 1\}.$$

Můžeme tedy charakterizovat ideály I obsahující všechny polynomy, které zaniknou na nějaké varietě V . Pokud platí $I = I(V(I))$, potom řekneme, že ideál I je **radikálem**, neboli pro každé $f^m \in I$ platí, že také $f \in I$. Dostáváme následující větu a odpověď na problém rovnosti (2.10).

Věta 2.9 (The Strong Nullstellensatz). *Nechť k je algebraicky uzavřené těleso. Jestliže I je ideál v $k[x_1, \dots, x_n]$, pak $I(V(I)) = \sqrt{I}$.*

VARIETA VYŠŠÍCH DIMENZÍ

Nyní se podívejme na případ, kdy polynomiální soustava obsahuje nekonečně mnoho řešení, varieta již není konečnou množinou a objektem našeho zájmu je její dimenze.

Nejdůležitějším invariantem lineárního podprostoru je jeho dimenze. Víme, že varieta ideálu je sjednocením konečného počtu lineárních podprostorů nad k^n . Dimenze variety $\dim(V)$ je potom rovna největší dimenzi dílčích podprostorů. Například dimenze sjednocení dvou rovin a přímky je 2. Pro daný ideál, jehož generátory definují dílčí podprostory, hledáme tedy největší dimenzi průniku dílčích podprostorů. Ačkoli je možno postupovat více směry při hledání dimenze variety [29], soustředíme se pouze na zajímavou souvislost dimenze variety s faktorovým okruhem. Z předešlého textu víme, že jistá závislost mezi počtem bodů variety a počtem standardních monomů existuje. Jak ale určit dimenzi variety ze vzájemného vztahu nekonečného počtu bodů variety s nekonečným počtem standardních monomů? Tento vzájemný vztah poprvé objevil David Hilbert a pro monomické ideály popsal v [45]. Pro zajímavost shrňme pouze nejdůležitější poznatky a pro detailnější studium odkažme na [29].

Protože se dále budeme pohybovat ve faktorovém okruhu jehož počet prvků je nekonečný, budeme používat pouze určitý počet těchto prvků, jejichž celkový stupeň je menší nebo roven s .

Lemma. *Počet monomů celkového stupně $\leq s$ v okruhu $k[x_1, \dots, x_n]$ je dán binomickým koeficientem $\binom{n+s}{s}$.*

Omezíme se tedy pouze na množinu polynomů s celkovým stupněm $\leq s$. Potom podle předchozího lemmatu platí, že vektorový prostor $k[x_1, \dots, x_n]_{\leq s}$ bude dimenze $\binom{n+s}{s}$. Ideál

$$I_{\leq s} = I \cap k[x_1, \dots, x_n]_{\leq s}$$

necht' označuje množinu polynomů v $I \subset k[x_1, \dots, x_n]$ s celkovým stupněm $\leq s$.

Definice 2.15. Necht' I je ideál v $k[x_1, \dots, x_n]$. Afinní Hilbertovo funkcí ideálu I nazveme funkci nezáporných celých čísel s , definovanou jako dimenzi vektorového prostoru standardní báze $\leq s$

$$\begin{aligned} {}^a HF_I(s) &= \dim k[x_1, \dots, x_n]_{\leq s} / I_{\leq s} \\ &= \dim k[x_1, \dots, x_n]_{\leq s} - \dim I_{\leq s}. \end{aligned}$$

Věta 2.10. Necht' $I \neq \langle 0 \rangle$ je monomický ideál v $k[x_1, \dots, x_n]$.

Pro všechna $s \geq 0$, ${}^a HF_I(s)$ udává počet standardních monomů celkového stupně $\leq s$.

Pro všechna s dostatečně velká, afinní Hilbertova funkce ideálu I je dána polynomem

$${}^a HF_I(s) = {}^a HP_I(s) = \sum_{i=0}^d b_i \binom{s}{d-i}$$

kde $b_i \in \mathbb{Z}$ a b_0 je kladné.

Stupeň polynomu ${}^a HP_I(s)$ je roven největší dimenzi podprostoru variety $V(I)$.

Věta 2.11. Necht' I je ideál v $k[x_1, \dots, x_n]$. Uvažujme uspořádání $>$ podle celkového stupně. Potom monomický ideál $in_{<}(I)$ má stejnou afinní Hilbertovu funkci jako I .

Věta 2.12 (O dimenzi). Necht' $V(I)$ je afinní varietou ideálu $I \subset k[x_1, \dots, x_n]$. Jestliže k je algebraicky uzavřené těleso, potom

$$\dim V(I) = {}^a HP_I(s).$$

Jestliže navíc uvažujeme uspořádání ${}_{\text{id eg}}>$ podle celkového stupně, potom platí

$$\dim V(I) = {}^a HP_{in_{<}(I)}(s).$$

Tato rovnost platí nad každým tělesem k pokud I je radikálem.

PŘÍKLADY

Podívejme se nyní na několik příkladů shrnujících poznatky této podkapitoly.

Příklad. Uvažujme ideál $I = \langle x^3 y + y^2, y^2 + y \rangle \subset \mathbb{C}[x, y]$. Pro zvolené uspořádání $x_{\text{lex}} > y$ spočteme Gröbnerovu bázi

$$G(I) = \{x^3 y - y, y^2 + y\}.$$

Potom $in_{<}(I) = \langle x^3 y, y^2 \rangle$ a varieta je nekonečným objektem. Dostáváme nekonečně mnoho řešení. Dimenzi variety je v tomto případě snadné určit, protože nekonečný objekt (nevyplňující celý prostor) v dvou-dimenzionálním prostoru může být pouze rovinná křivka. Potvrdíme naši hypotézu výše uvedenou teorií. Určíme počet standardních polynomů $\leq s$. Z obrázku 2.4 (vlevo) vidíme, že v prvním řádku je $s+1$ monomů splňujících podmínku. V druhém řádku jsou další tři standardní monomy. Podle věty 2.10 dostáváme pro s dostatečně velké ($s \geq 3$) afinní Hilbertův polynom

$${}^a HP_{in_{<}(I)}(s) = s + 4,$$

jehož stupeň je podle věty 2.12 roven dimenzi variety $V(I)$, a tedy $\dim(V)=1$.

Příklad. Uvažujme ideál $I=\langle x^3+y^2, xy+y \rangle \subset \mathbb{C}[x, y]$. Opět pro zvolené uspořádání $x >_{lex} y$ spočteme Gröbnerovu bázi

$$G(I)=\{x^3+y, xy+y, y^3-y\}$$

Potom $in_{<}(I)=\langle x^3, xy, y^3 \rangle$ a podle věty 2.7 zjišťujeme, že varieta je nulové dimenze, $\dim(V)=0$. Počet standardním monomů $(1, x, x^2, y, y^2)$, viz obrázek 2.4 (vpravo), je větší než počet řešení $[x, y], [x+1, y+1], [x+1, y-1]$. Totožná řešení generuje podle předešlých vět 2.8 a 2.9 radikál ideálu I . Radikál ideálu I a příslušná GB by vypadaly následovně

$$I=\langle x+y^2, xy+y \rangle \subset \mathbb{C}[x, y]$$

$$G(I)=\{x+y^2, y^3-y\}$$

Počet standardních monomů $(1, y, y^2)$ je v tomto případě přímo roven počtu řešení.

Poznámka. Pro ideál nulové dimenze platí, že prvky standardní báze jsou závislé na použitém způsobu uspořádání, nicméně počet prvků je stále stejný [29].

Příklad. Uvažujme ideál $I=\langle y-x^2, z-x^3 \rangle \subset \mathbb{R}[x, y, z]$ jehož varietou je (2.4). Zvolme dvě různá uspořádání $z >_{lex} y >_{lex} x$, $z >_{grevlex} y >_{grevlex} x$ a spočteme příslušné Gröbnerovy báze

$$G^1(I)=\{y-x^2, z-x^3\}, \langle in_{<}(G^1) \rangle = in_{<}(I)=\langle y, z \rangle$$

$$G^2(I)=\{x^2-y, y^2-xz, xy-z\}, \langle in_{<}(G^2) \rangle = \langle x^2, xy, y^2 \rangle$$

Připomeňme, že ideál I je radikálem, $I=I(V(I))$, což jsme dokázali v kap. 2.1 pomocí parametrizace. Z bázi je zřejmé, že počet standardních monomů je v obou případech nekonečný. Dimenze variety musí být rovna stupni Hilbertova polynomu. V případě báze G^1 dostáváme Hilbertův polynom (viz obrázek 2.5)

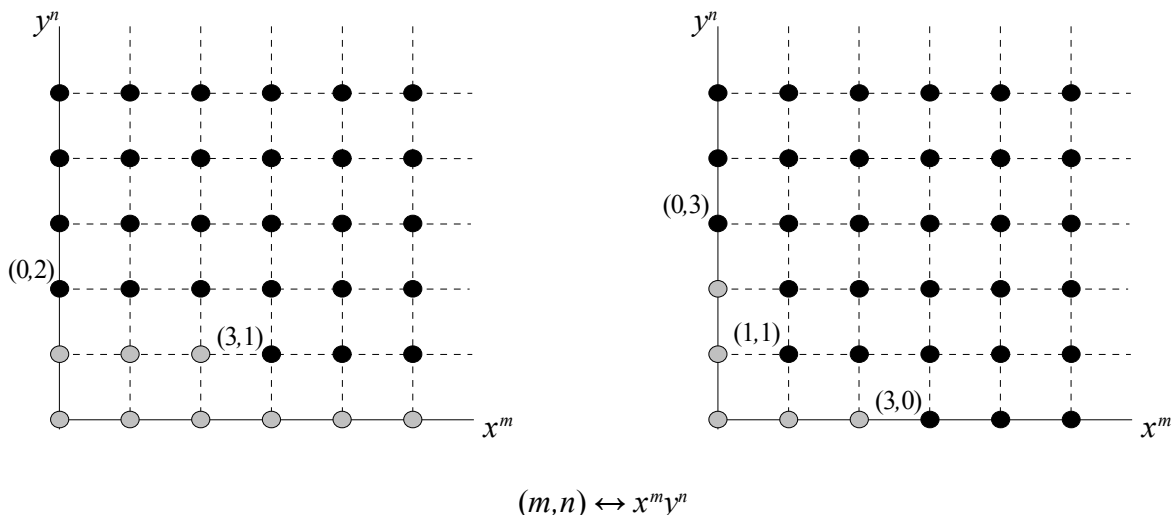
$${}^a HP_{in_{<}(I)}(s)=s+1$$

V případě G^2 dostáváme

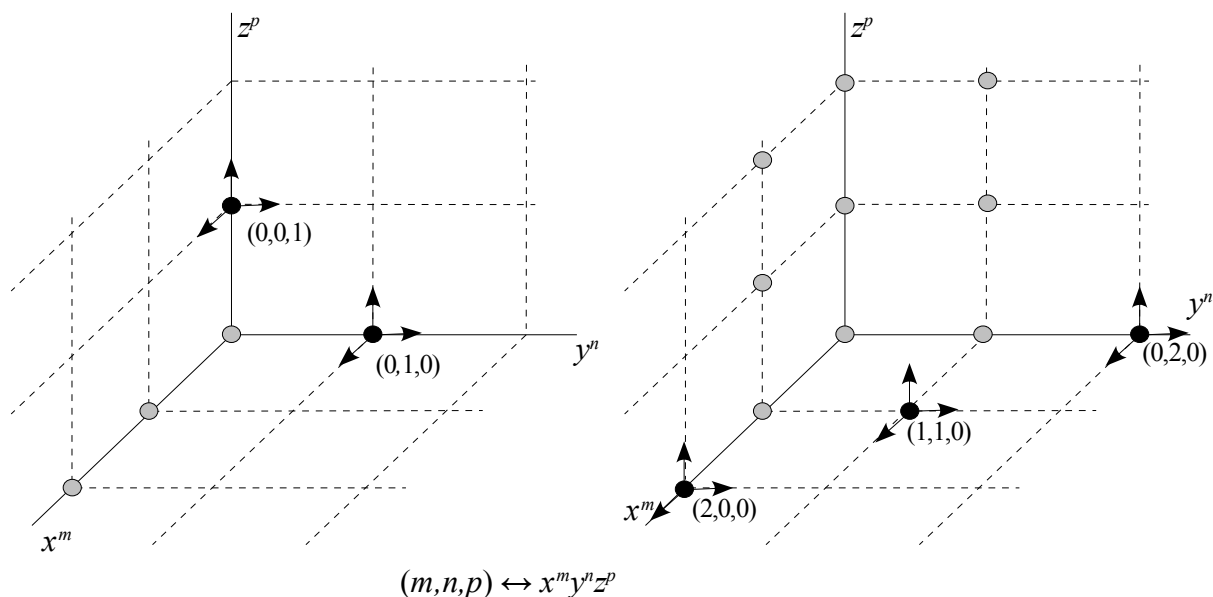
$${}^a HP_{in_{<}(I)}(s)=3s+1$$

Stupeň Hilbertových polynomů je pro obě uspořádání shodný a tedy dimenze variety (2.4) je rovna jedné, což jsme již dříve ukázali.

Vidíme, že složitost GB se výrazně mění vzhledem k uspořádání členů. V předchozím příkladu jsme si při vhodně zvoleném uspořádání ušetřili výpočet GB , neboť ideál byl přímo GB . Pro druhé uspořádání už byl výpočet nevyhnutelný. Nabízí se tedy otázka, jak volit typ uspořádání vzhledem ke složitosti nalezení Gröbnerovy báze. Tomuto problému je věnováno velké úsilí a do dnešního dne bylo nalezeno mnoho postupů, jak redukovat čas výpočtu hledání GB . Čtenáře odkážeme na appendix A a odkazy v něm uvedené.



Obrázek 2.4: Naznačení monomických exponentů v $k[x,y]$. Světlé body definují exponenty standardních monomů. Tmavé body se souřadnicemi jsou exponenty vedoucích monomů ideálu I , generující všechny ostatní tmavé body (exponenty).



Obrázek 2.5: Naznačení monomických exponentů v $k[x,y,z]$ pro báze G^1 (vlevo) a G^2 (vpravo). Světlé body definují exponenty standardních monomů. Tmavé body se souřadnicemi jsou exponenty vedoucích monomů ideálu I , generující všechny ostatní tmavé body (z důvodu přehlednosti vypuštěno).

2.5.2 HLAVNÍ PROBLÉM TEORIE IDEÁLŮ (IDEAL MEMBERSHIP PROBLEM)

Zda polynom f je prvkem daného ideálu $I(F)$ je známý problém, v literatuře nazýván jako *Main problem of polynomial ideal theory*, nebo také jako *Ideal membership problem*. Pro obecnou soustavu F je tento problém jen těžko řešitelný, ale pro Gröbnerovu bázi dostáváme jednoduchý a elegantní rozhodovací proces, který vyplývá z následující věty.

Věta 2.13 (Buchberger). *Nechť F je soustava polynomiálních rovnic a f je polynom ve více proměnných. Pokud F je současně Gröbnerovou bázi, potom $f \in I(F) \Leftrightarrow \bar{f}^F = 0$.*

Tuto skutečnost jsme si již dříve ověřili na našem příkladu první GB . Podívejme se na další příklad.

Příklad. Uvažujme ideál $I = \langle f_1, f_2 \rangle = \langle xz - y^2, x^3 - z^2 \rangle$ a polynom $f = -4x^2y^2z^2 + y^6 + 3z^5$.

Pokud zavedeme lexikografické uspořádání ve tvaru $y >_{lex} z >_{lex} x$, potom generátory ideálu I jsou podle věty 2.2 přímo generátory Gröbnerovy báze

$$G(I) = \{f_1, f_2\}; \quad y >_{lex} z >_{lex} x$$

Pokud nyní spočteme zbytek po dělení \bar{f}^G , zjistíme, že je roven 0 a tedy $f \in I$. Alternativním testem může být přidání dané funkce f do generátorů ideálu

$$I^* = \langle f_1, f_2, f \rangle = \langle xz - y^2, x^3 - z^2, -4x^2y^2z^2 + y^6 + 3z^5 \rangle$$

a určení jeho Gröbnerovy báze G^* . Pokud platí $G \equiv G^*$, což je v tomto případě splněno, oba ideály generují stejnou množinu řešení a tedy $f \in I$.

Jako protipříklad uvažujme původní ideál I a novou funkci $p = xy - 5z^2 + x$. V tomto případě již zbytek po dělení \bar{p}^G není roven 0, ale $\bar{p}^G = yx - 5x^3 + x$, což je v rozporu s teorií a tedy $p \notin I$. Můžeme také využít větu 2.2 a vyhnout se dělení. Podle této věty by muselo platit, že $in_{<}(p) = xy$ je prvkem ideálu $\langle in_{<}(G) \rangle = \langle y^2, z^2 \rangle$ a tedy dělitelný jeho generátory, což zjevně neplatí.

Příklad. Ptáme se, zda polynom $f = x_1^6 + x_2^6 - x_1^2x_2^2$ může být vyjádřen jako polynom v proměnných $p_1 := x_1^2 + x_2^2$ a $p_2 := x_1x_2$.

Zaveďme tedy polynomiální soustavu F v $k[x_1, x_2, p_1, p_2]$ ve tvaru

$$F = \{-p_1 + x_1^2 + x_2^2, -p_2 + x_1x_2\}$$

a určíme její Gröbnerovu bázi

$$G = \{p_2^2 - x_2^2p_1 + x_2^4, x_1p_2 - x_2p_1 + x_2^3, -p_2 + x_1x_2, -p_1 + x_1^2 + x_2^2\}, \quad x_1 >_{lex} x_2 >_{lex} p_1 >_{lex} p_2$$

Vyčíslením zbytku $h := \bar{f}^G$ dostaneme

$$h := p_1^3 - 3p_1p_2^2 - p_2^2$$

Protože výsledek je funkcí pouze proměnných p_i , je polynom f plně redukován generátory Gröbnerovy báze a je tedy možné ho vyjádřit jako

$$f := (x_1^2 + x_2^2)^3 - 3(x_1^2 + x_2^2)(x_1x_2)^2 - (x_1x_2)^2$$

2.5.3 ROVNOST IDEÁLŮ

Redukovaná GB je dána jednoznačně ideálem I a způsobem uspořádání členů. Můžeme tedy snadno rozlišit zda jsou dané ideály shodné porovnáním jejich redukováných GB .

2.5.4 HLEDÁNÍ PRŮNIKU IDEÁLŮ

Uvažujme ideál I generovaný množinou $\{f_1, \dots, f_n\}$ a ideál J generovaný množinou $\{g_1, \dots, g_m\}$. Potom průnik ideálů $I \cap J$ je dán Gröbnerovou bází získanou z množiny $\{tf_1, \dots, tf_n, (1-t)g_1, \dots, (1-t)g_m\}$. Tato vlastnost nám umožňuje nalézt nejmenší společný násobek a největší společný dělitel dvou polynomů f a g , protože ten je generátorem průniku těchto polynomů.

2.5.5 PROBLÉM ELIMINACE

Uvažujme ideál $I \subset k[x_1, \dots, x_n]$. Chceme nalézt všechny polynomy ideálu I , které nezávisí na proměnné x_n , neboli $I \cap k[x_1, \dots, x_{n-1}]$. Geometricky je eliminace proměnných projekcí variety $V(I)$ do podprostoru $k[x_1, \dots, x_{n-1}]$.

Věta 2.14. *Nechť F je soustava polynomiálních rovnic v $k[x_1, \dots, x_n]$ a $i \leq n$. Pokud F je současně Gröbnerovou bází s respektováním uspořádání $x_1 < x_2 < \dots < x_n$, potom*

$$I(F) \cap k[x_1, \dots, x_i] = I(F \cap k[x_1, \dots, x_i]).$$

Z věty vyplývá, že pokud F je Gröbnerovou bází, potom vybráním polynomů obsahujících pouze proměnné $k[x_1, \dots, x_i]$ dostáváme podprostor variety $V(I)$. To pro obecný ideál neplatí a je to další z velkých problémů polynomiální algebry. Nicméně pro jeho Gröbnerovu bází ano. Důsledkem této věty je, že v GB s respektováním lexikografického uspořádání můžeme postupnou substitucí nalézt všechna řešení původní soustavy F .

2.5.6 IMPLICITIZACE

Implicitizaci, neboli převedení parametrického vyjádření variety na implicitní, je možné provést pomocí GB s lexikografickým uspořádáním. Z předchozí podkapitoly víme, že za těchto předpokladů dochází k eliminaci proměnných. Implicitizaci se rozumí převod popisu ideálu variety z prostoru $k[t_1, \dots, t_m, x_1, \dots, x_n]$ do prostoru $k[x_1, \dots, x_n]$. Spočtením GB s vhodným lexikografickým uspořádáním lze dosáhnout vyloučení parametrů t_1, \dots, t_m .

Příklad. Vraťme se k varietě (2.4) a podívejme se na parametrický popis tečného prostoru (2.6) této variety. Ideál tečného prostoru je

$$I = \langle t + u - x, t^2 + 2tu - y, t^3 + 3t^2u - z \rangle.$$

Víme, že v procesu výpočtu GB s lexikografickým uspořádáním se postupně eliminují neznámé od lexikograficky největší k nejmenší. Proto zavedeme uspořádání $t > u > z > y > x$ a spočteme příslušnou bázi

$$G = \left\{ \begin{array}{l} z^2 - 6zyx + 4zx^3 + 4y^3 - 3y^2x^2, \\ z - 2ux^2 + 2yu - 3yx + 2x^3, 2uz - 2ux^3 + 3xz - 4y^2 - yx^2 + 2x^4, \\ y + u^2 - x^2, t + u - x \end{array} \right\}$$

První rovnice, neobsahující žádný parametr, je hledaným implicitním popisem tečného prostoru variety, totožným s (2.7).

Pokud by varieta byla popsána racionálním parametrickým popisem

$$x_1 = \frac{f_1(t_1, \dots, t_m)}{p_1(t_1, \dots, t_m)}$$

$$\vdots$$

$$x_n = \frac{f_n(t_1, \dots, t_m)}{p_n(t_1, \dots, t_m)},$$

potom ideál I je tvořen nejen generátory $(p_1x_1 - f_1, \dots, p_nx_n - f_n)$, ale do generátorů ideálu musíme přidat ještě jednu rovnici $1 - pw$, $p = p_1p_2 \dots p_n$, $w \in \mathbb{R}$, která zajistí nenulovost jmenovatelů. Následně volíme uspořádání pro výpočet GB ve tvaru $w > t_1 \dots t_m > x_1 \dots x_n$. Více lze nalézt v [29].

Příklad. Hledání *vázaného extrému* [69] je příkladem spojení metody Lagrangeových multiplikátorů a Gröbnerovy báze s lexikografickým uspořádáním.

Uvažujme problém nalezení extrému funkce $f(x, y)$ vázaného omezující podmínkou $c(x, y)=0$

$$\begin{aligned} f(x, y)=h: & f(x, y)=2y-(x+1)^2y^2 \\ c(x, y)=0: & c(x, y)=x^2+y^2-9. \end{aligned} \quad (2.32)$$

Využijeme vlastnost Lagrangeových multiplikátorů a budeme hledat vázaný extrém funkce ve tvaru $f-\lambda c \rightarrow \text{extrém}$, kde $\lambda \in \mathbb{R}$ může být libovolná konstanta. Hledáme tedy extrém variety $V(f-h, c)$. Pro extrém funkce musí platit

$$\frac{\partial f}{\partial q} - \frac{\partial \lambda c}{\partial q} = 0, \quad q \in \{x, y, \lambda\}.$$

Dostáváme ideál

$$I = \langle -2y^2x - 2y^2 - 2\lambda x, 2 - 2yx^2 - 4yx - 2y - 2\lambda y, x^2 + y^2 - 9 \rangle.$$

Nyní se přímo nabízí využití vlastnosti Gröbnerovy báze generované ideálem I s vhodným lexikografickým uspořádáním $\lambda >_{\text{lex}} x >_{\text{lex}} y$. Využitím Buchbergerova algoritmu dostaneme

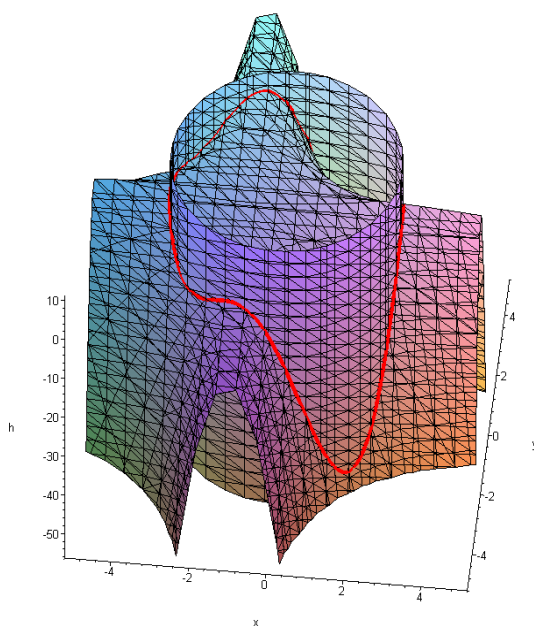
$$G = \left\{ \begin{aligned} &4y^5 - 56y^3 + 180y - 575y^2 - 9 + 352y^4 + 4y^8 - 67y^6, \\ &-1357y^3 + 69x + 2142y - 16y^7 + 264y^5 - 102y^4 + 362y^2 + 8y^6 - 342, \\ &207\lambda + 4y^7 - 48y^6 - 43y^5 + 520y^4 + 46y^3 - 1091y^2 + 373y - 18 \end{aligned} \right\}.$$

První rovnice GB obsahuje pouze proměnnou y , takže můžeme použít metody numerické matematiky a najít všechny kořeny této rovnice. Poté dosadit do druhé a třetí a získat řešení proměnných x a λ . Omezíme se pouze na reálná řešení a výsledkem je bod

$$(x = -0.958, y = 2.843), \quad \lambda = 0.350.$$

Funkce f v tomto bodě nabývá svého maxima $h_{\max} = 5.6716$, viz obrázek 2.6. Parametrizací omezující podmínky ve tvaru $x = 3 \cos(\phi), y = 3 \sin(\phi); \phi \in [0, 2\pi]$ dostaneme parametrizaci

$$\begin{aligned} x &= 3 \cos(\phi), \quad y = 3 \sin(\phi) \\ h &= 6 \sin(\phi) - 81 \sin^2(\phi) \cos^2(\phi) - 54 \sin^3(\phi) \cos(\phi) - 9 \sin^4(\phi). \end{aligned} \quad (2.33)$$



Obrázek 2.6: Vázaný extrém a GB. Červená křivka je variety $V(f-h, c)$, dána rovnicemi (2.33).

2.6 APLIKACE GRÖBNEROVY BÁZE V TEORII ŘÍZENÍ

Tato kapitola je věnována využití GB v teorii řízení. Šíře využití byla již nastíněna v úvodu. Zde se soustředíme na nejdůležitější oblasti. V oblasti robotiky provedeme rozbor využití GB při návrhu robotických manipulátorů z hlediska kinematických úloh. Zaměříme se také na problém singularit a možnosti konfigurace vzhledem k poloze efektoru robota [87]. V automatickém řízení a teorii systémů se zaměříme na problém přiřazení spektrálních vlastností matice dynamiky pomocí výstupní zpětné vazby [88] a ukážeme využití GB při parametrizaci všech zpětných vazeb minimálním počtem parametrů [90]. Zajímavé využití GB při řešení diofantické rovnice v okruhu polynomů více proměnných, se kterými se setkáváme u více rozměrných systémů, je možné nalézt v [16]. Využití GB v optimálním řízení lze nalézt v [106], a bude podrobně představeno v kapitole 3 pro úlohy s omezeními na stav systému. Další možnosti využití lze nalézt například v [7],[104],[109],[110].

2.6.1 GRÖBNEROVA BÁZE A KINEMATICKÁ ÚLOHA

Nedílnou součástí robotiky je návrh samotného mechanismu dle požadavků na pohyb v jeho konfiguračním a pracovním prostoru. V těchto návrzích je obvykle klíčovým požadavkem rozměr stroje, pracovní rozsah, omezení senzorů a aktuátorů, specifické řízení (přesnost, rychlost stroje, robustnost při změnách zatížení atd.), přičemž dobrá ovladatelnost těchto robotů je silně závislá právě na konstrukčních vlastnostech a parametrech stroje. S využitím GB lze vyšetřovat vliv těchto parametrů na řízení [63]. V kinematice řešíme problém redundance mechanismu, problém kolizních stavů robota a problém singulárních bodů a jejich okolí [5],[87]. Je tedy nutné optimalizovat parametry robota tak, abychom se těmito stavům a okolím vyhnuli, nebo je vytěsnili z pracovní oblasti mechanismu. Proto je nutné nalézt kompletní popis konfiguračního a pracovního prostoru mechanismu, jeho singulární body, kolizní polohy mechanismu a také různá nastavení mechanismu v konfiguračním prostoru, která budou vykazovat stejné chování v prostoru pracovním. Tento problém je znám jako problém kinematické úlohy a je rozdělen do dvou podoblastí. Ilustrujme využití Gröbnerovy báze v těchto oblastech na následujícím příkladu, převzatém z [29].

PŘÍMÁ KINEMATICKÁ ÚLOHA

Uvažujme mechanismus s daným počtem pohyblivých částí, na jejichž spojích definujeme přípustné pohyby. Většinou se bude jednat pouze o translace a rotace a my se v dalším omezíme pouze na tyto základní pohyby. V přímé kinematické úloze hledáme zobrazení z prostoru posunutí a natočení do pracovního prostoru robota $r:Q \rightarrow X$, který je nejčastěji popsán v kartézských souřadnicích.

Uvažujme sériový manipulátor s dvěma rameny.

Konfigurační prostor Q je dán $\theta \in S^1 \times S^1$. Pracovní prostor manipulátoru X je definován na euklidovské rovině \mathbb{R}^2 jako $\{[x, y] \in \mathbb{R}^2; [x, y]^T = r(\theta)\}$. Přímé kinematické zobrazení je dáno jako $r: S^1 \times S^1 \rightarrow \mathbb{R}^2; \theta \mapsto r(\theta)$,

$$r(\theta) = \begin{bmatrix} l \cos(\theta_1) + l \cos(\theta_1 + \theta_2) \\ l \sin(\theta_1) + l \sin(\theta_1 + \theta_2) \end{bmatrix}. \quad (2.34)$$

Abychom mohli vyjádřit toto zobrazení v jazyce ideálů a afinních variet, je třeba parametrizovat transcendentní funkce sinus a kosinus na algebraické varietě kružnice $V(c^2 + s^2 - 1)$. Dostáváme parametrizaci $c_i = \cos(\theta_i)$, $s_i = \sin(\theta_i)$ s daným omezením $c_i^2 + s_i^2 - 1 = 0$. Dosazením do (2.34)

a využitím trigonometrických úprav dostaneme polynomiální soustavu rovnic definující kinematické zobrazení. Omezení musíme samozřejmě také zahrnout do generátorů ideálu

$$I = \langle x - l c_1 - l(c_1 c_2 - s_1 s_2), y - l s_1 - l(s_1 c_2 + c_1 s_2), c_1^2 + s_1^2 - 1, c_2^2 + s_2^2 - 1 \rangle. \quad (2.35)$$

Poznámka. Parametrizace sinus a kosinus na varietě kružnice je známý postup, který přináší výhody i nevýhody. Výhodou je, že funkce jsou popsány na celém oboru. Nevýhodou je, že dostáváme rovnice omezení, a tedy zvyšujeme složitost výpočtu Gröbnerovy báze. Další možnou parametrizací je tzn. racionální parametrizace pomocí funkce tangens. Zde již žádné omezující rovnice nepřibudou, ale tato parametrizace nepopisuje celý obor. Podrobněji jsou obě parametrizace popsány v [29].

Poznámka. Pro otevřený řetězec je problém přímé kinematické úlohy vždy řešitelný.

ZPĚTNÁ KINEMATICKÁ ÚLOHA

Pokračujme v popisu sériového manipulátoru a uvažujme konkrétní bod $P = [x, y]$ na euklidovské rovině pracovního prostoru. Úkolem zpětné kinematické úlohy je odpovědět na otázku, zda je možné umístit koncový bod (efektor) manipulátoru do bodu P , a pokud ano, kolik konfigurací manipulátoru je k dispozici. Hledáme tedy inverzní zobrazení $r^{-1}(s)$.

Spočteme Gröbnerovu bázi ideálu (2.35) v polynomiálním okruhu $\mathbb{R}(x, y, l)[s_1, c_1, s_2, c_2]$ s respektováním uspořádání $c_2 >_{\text{lex}} s_2 >_{\text{lex}} c_1 >_{\text{lex}} s_1$,

$$G = \left(\begin{array}{c} c_2 - \frac{x^2 + y^2 - 2l^2}{2l^2} \\ s_1^2 - \frac{(x^2 y + y^3)}{l(x^2 + y^2)} s_1 - \frac{(4l^2 x^2 - x^4 - 2x^2 y^2 - y^4)}{4l^2(x^2 + y^2)} \\ c_1 + \frac{y}{x} s_1 - \frac{(x^2 + y^2)}{2lx} \\ s_2 + \frac{x^2 + y^2}{lx} s_1 - \frac{x^2 y + y^3}{2l^2 x} \end{array} \right). \quad (2.36)$$

Výše uvedený zápis okruhu znamená, že GB bude počítána nad tělesem racionálních funkcí a bude záviset na hodnotách symbolických parametrů x, y, l . Generátory GB popisují všechna řešení inverzního zobrazení $r^{-1}(c)$. Substitucí konkrétních hodnot do Gröbnerovy báze hovoříme o takzvané *specializaci*. Ta může výrazně ovlivnit chování Gröbnerovy báze. Proto je vhodné ptát se, která specializace je vhodná a která nikoli. Existuje také zvláštní druh tzv. úplně (Comprehensive) Gröbnerovy báze [29], která je Gröbnerovou bází za všech možných specializací.

Podívejme se nyní na problém specializace tohoto manipulátoru a rozdělme problém na dílčí úlohy, které mohou ovlivnit řešení soustavy.

A) Poloha efektoru splňuje podmínku: $x \neq 0, x^2 + y^2 \neq 0$.

Z Gröbnerovy báze (2.36) vidíme, že soustava má řešení. Druhá rovnice druhého řádu ukazuje na skutečnost, že řešení jsou dvě, což odpovídá dvěma možným nastavením manipulátoru pro daný bod P v pracovním prostoru, viz obrázek 2.8. Velikost pracovního prostoru manipulátoru je dána opět druhou rovnicí. Ta nabude reálného řešení pouze v případě kladného diskriminantu $D = l^2 x^2 (x^2 + y^2) (4l^2 - x^2 - y^2)$, neboli pouze pokud $0 < x^2 + y^2 \leq 4l^2$. Dostáváme vnější hranici pracovního prostoru manipulátoru v podobě kružnice o poloměru $2l$. Na hranici obě řešení splývají, což je intuitivně zřejmé.

B) Poloha efektoru splňuje podmínku: $x=0, y=0$.

V tomto případě specializace selhala a některé polynomy nejsou definovány. Pokud dosadíme bod P v tomto tvaru do generátorů ideálu a přepočteme Gröbnerovu bázi

$$G = \{c_1^2 + s_1^2 - 1, s_2, 1 + c_2\} \quad (2.37)$$

zjistíme, že soustava má nekonečně mnoho řešení. Úhel θ_1 může být libovolný, zatímco $\theta_2 = \pi$, což znamená, že druhé rameno překrývá první a ukazuje na jeho počátek.

C) Poloha efektoru splňuje podmínku: $x=0, y \neq 0$.

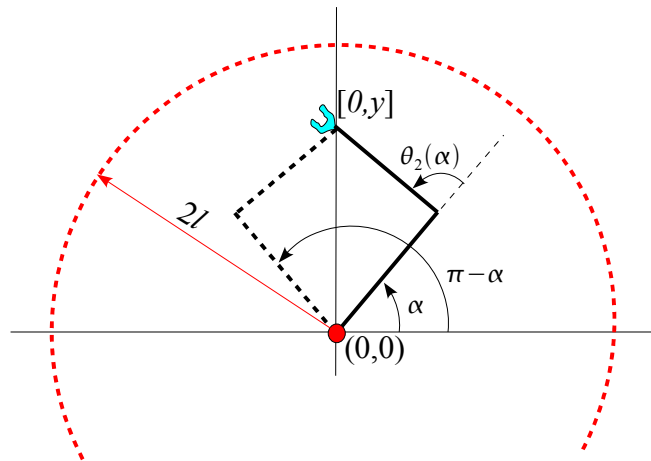
Poslední možnou specializací je umístění bodu P na osu y mimo počátku. Zde opět specializace selhává a my musíme dosadit bod P v tomto tvaru do generátorů ideálu a přepočíst Gröbnerovu bázi

$$G = \left\{ c_2 - \frac{(y^2 - 2l^2)}{2l^2}, s_1 - \frac{y}{2l}, c_1^2 + \frac{(y^2 - 4l^2)}{4l^2}, s_2 - \frac{yc_1}{l} \right\}. \quad (2.38)$$

Zvláštností je, že s_1 již není druhého řádu, zato c_1 ano. Tuto varietu polohy manipulátoru si můžeme snadno představit. Manipulátor může nabývat dvou vzájemně symetrických poloh vůči ose y , obrázek 2.7. Z geometrie úlohy je zřejmé, že úhel θ_2 je pro daný bod P na ose y jednoznačně definován úhlem θ_1 . θ_1 může pro daný bod P nabývat dvou hodnot α nebo $\pi - \alpha$. A protože vlastnosti funkce sinus a kosinus splňují následující vztahy

$$\begin{aligned} \sin \alpha &= \sin(\pi - \alpha) \\ \cos \alpha &= -\cos(\pi - \alpha), \end{aligned}$$

je změna řádů vysvětlena. Vlastnosti, které nám Gröbnerova báze spolu se specializací odhaluje, jsou v literatuře nazývány jako *kinematické singularity*. Pro úplnost dokončíme studii našeho manipulátoru stručným popisem singularit a pro detailnější studii se odkažme na [5].



Obrázek 2.7: Dvouramenný sériový manipulátor – specializace.

JAKOBIAN A KINEMATICKÁ SINGULARITA

Pro popis singularit můžeme kromě Gröbnerovy báze využít i Jakobián kinematického zobrazení. Popis singularit tímto způsobem je velmi dobře znám a v literatuře podrobně popsán [5],[87]. Jakobián můžeme počítat jak ze samotného zobrazení r , tak z jeho Gröbnerovy báze G , protože ta jak víme, generuje stejný ideál. Dimenze Jakobiánu je dána počtem nezávislých stupňů volnosti v konfiguračním prostoru Q a pracovním prostoru X .

Jakobián je definován jako

$$J_r(\boldsymbol{\theta}) = \begin{bmatrix} \nabla r_1(\boldsymbol{\theta}) \\ \nabla r_2(\boldsymbol{\theta}) \\ \vdots \\ \nabla r_n(\boldsymbol{\theta}) \end{bmatrix}.$$

Pro (2.34) dostáváme

$$J_r(\theta_1, \theta_2) = \begin{bmatrix} -l \sin(\theta_1) - l \sin(\theta_1 + \theta_2) & -l \sin(\theta_1 + \theta_2) \\ -l \cos(\theta_1) + l \cos(\theta_1 + \theta_2) & +l \cos(\theta_1 + \theta_2) \end{bmatrix}.$$

Definice 2.16. Kinematickou singularitou robota je bod $j \in Q$ takový, že

$$\text{rank}(J_r(j)) < \min(\dim(Q), \dim(X)).$$

Snížení hodnosti J_r můžeme otestovat položením determinantu Jakobiánu rovno nule. V případě manipulátoru dostáváme

$$\begin{aligned} 0 &= \det J_r(\theta_1, \theta_2) \\ 0 &= \sin(\theta_1 + \theta_2) \cos \theta_1 - \cos(\theta_1 + \theta_2) \sin \theta_1 \\ 0 &= \sin \theta_2, \end{aligned} \quad (2.39)$$

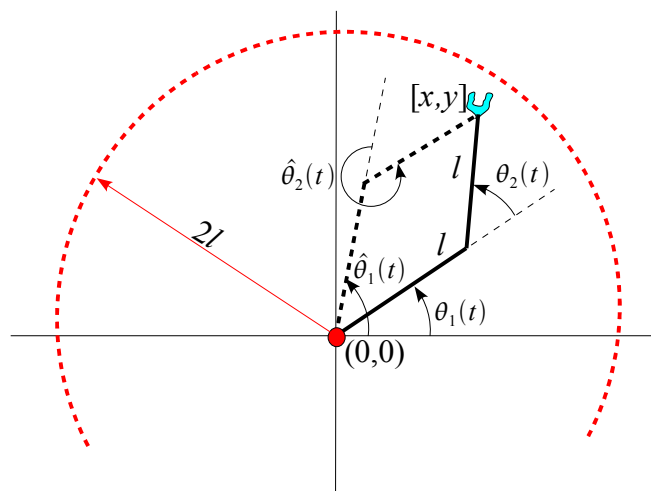
což je případ specializace, kdy koncový bod (efektor) leží buď v počátku soustavy souřadnic $P=[0,0]$, nebo na okraji pracovního prostoru manipulátoru, tj. kružnice o poloměru $2l$. V případě Gröbnerovy báze využijeme generátory ideálu (2.35) a otestujeme přítomnost singularit stejným způsobem. Dostáváme Jakobián

$$J_G(c_1, c_2, s_1, s_2) = \begin{bmatrix} l + l c_2 & l c_1 & -l s_2 & -l s_1 \\ l s_2 & l s_1 & l + l c_2 & l c_1 \\ 2c_1 & 0 & 2s_1 & 0 \\ 0 & 2c_2 & 0 & 2s_2 \end{bmatrix}$$

a testem pomocí determinantu získáme stejný výsledek jako v (2.39),

$$0 = \det J_G(c_1, c_2, s_1, s_2) \rightarrow 0 = 4l^2 s_2 (c_1^2 + s_1^2), \quad 0 = s_2.$$

Určili jsme, že bod $[0,0]$ a kružnice o poloměru $2l$ jsou kinematickou singularitou, viz obrázek 2.8.



Obrázek 2.8: Dvouramenný sériový manipulátor – Počátek souřadnic a kružnice o poloměru $2l$ tvoří kinematickou singularitu; Mimo singulárních poloh existují dvě různé konfigurace robota.

To v praxi například znamená, že počátek souřadnic nemůže systém opustit libovolným směrem aniž by nepootočil první rameno do směru kolmého na tento pohyb. Dimenze pracovního prostoru v tomto bodě je rovna pouze jedné. Gröbnerova báze nám umožňuje algoritmicky popsat chování systému a nalézt speciální oblasti v pracovním prostoru. To vše je možné zabudovat přímo do metod řídicího algoritmu.

V oblasti řízení robotických manipulátorů se setkáváme s problémem jak nalézt parametrickou trajektorii v pracovním prostoru $s(t) \in X$ a k ní korespondující trajektorii v prostoru konfiguračním $j(t) \in Q$ tak, aby platilo $s(t) = r(j(t))$. Samozřejmostí jsou v praxi i další požadavky na tyto trajektorie, jako například omezení na rychlost změny $(\theta_1, \dot{\theta}_2)$ v konfiguračním prostoru. Zde je spojení s vlastnostmi Jakobiho matice opět zřejmé, protože platí

$$s'(t) = J_r(j(t)) j'(t).$$

Pro systémy, které opakují svou činnost a jejichž $s(t)$ je uzavřená, je zase vhodné hledat příslušnou $j(t)$ také uzavřenou. Studium těchto metod se zabývá například [21],[78].

2.6.2 PŘÍRAZENÍ PÓLŮ VÝSTUPNÍ ZPĚTNOU VAZBOU

Rozmístění pólů uzavřeného systému je základním požadavkem v teorii řízení. Při návrhu regulační smyčky se pomocí stavové/výstupní zpětné vazby snažíme systému přiřadit požadované chování [71]. Budeme-li se pohybovat v oblasti lineárních systémů, můžeme i zde využít teorii *GB*. V dalším se omezíme pouze na přiřazení spektrálních vlastností matici dynamiky uzavřeného systému pomocí výstupní zpětné vazby, kde využitím *GB* získáme zajímavé možnosti parametrizace všech přípustných zpětných vazeb pomocí minimálního počtu parametrů [90],[54]. Podobných výsledků lze dosáhnout i u stavové zpětné vazby. Ta byla již ale podrobně prozkoumána v [88], a nalezení minimální parametrizace bylo dosaženo i bez použití *GB*. Poznamenejme, že největší předností tohoto přístupu je opět jeho algoritmizovatelnost.

Ještě než přistoupíme k samotnému problému přiřazení pólů pomocí výstupní zpětné vazby, podívejme se na dva motivační příklady, které jednoduše ilustrují daný problém. V dalším textu, z důvodu stručnosti, předpokládáme základní znalosti z teorie lineárních systémů a teorie řízení.

Příklad. Uvažujme lineární dynamický systém popsany trojicí matic (C, A, B) :

$$A = \begin{bmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -1 \end{bmatrix}, \quad B = \begin{bmatrix} 1 & 1 \\ 1 & 0 \\ 0 & 0 \end{bmatrix}, \quad C = \begin{bmatrix} 1 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

Úkolem je nalézt a parametrizovat všechny výstupní zpětné vazby

$$K = \begin{bmatrix} k_{11} & k_{12} \\ k_{21} & k_{22} \end{bmatrix}$$

přiřazující dané póly s_1, s_2, s_3 . Intuitivní postup pro tento konkrétní příklad je následující:

Určíme matici uzavřeného systému

$$A_{cl} \triangleq A + B K C = \begin{bmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -1 \end{bmatrix} + \begin{bmatrix} 1 & 1 \\ 1 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} k_{11} & k_{12} \\ k_{21} & k_{22} \end{bmatrix} \begin{bmatrix} 1 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$A_{cl} = \begin{bmatrix} k_{11} + k_{21} - 1 & k_{11} + k_{21} & k_{12} + k_{22} \\ k_{11} & k_{11} - 1 & k_{12} \\ 0 & 0 & -1 \end{bmatrix}$$

a korespondující charakteristický polynom

$$\begin{aligned} \det(sI - A_{CL}) &= s^3 + (-2k_{11} - k_{21} + 3)s^2 + (-4k_{11} - 2k_{21} + 3)s - 2k_{11} - k_{21} + 1 \\ &= s^3 + f_1(k)s^2 + f_2(k)s + f_3(k). \end{aligned}$$

Dále definujeme požadavek na póly s_1, s_2, s_3 charakteristického polynomu ve tvaru

$$\begin{aligned} (s - s_1)(s - s_2)(s - s_3) &= s^3 - (s_1 + s_2 + s_3)s^2 + (s_1s_2 + s_1s_3 + s_2s_3)s - s_1s_2s_3 \\ &= s^3 + \bar{f}_1(s_i)s^2 + \bar{f}_2(s_i)s + \bar{f}_3(s_i), \quad i = 1, 2, 3 \end{aligned}$$

neboli, chceme nalézt parametry zpětnovazební matice tak, aby platilo

$$s^3 + f_1(k)s^2 + f_2(k)s + f_3(k) \stackrel{\Delta}{=} s^3 + \bar{f}_1(s_i)s^2 + \bar{f}_2(s_i)s + \bar{f}_3(s_i).$$

Dostáváme polynomiální soustavu

$$\begin{aligned} \bar{f}_1(s_i) - f_1(k) &= 0 \\ \bar{f}_2(s_i) - f_2(k) &= 0 \\ \bar{f}_3(s_i) - f_3(k) &= 0 \end{aligned} \tag{2.40}$$

a k ní korespondující ideál

$$I = \langle -s_1 - s_2 - s_3 + 2k_{11} + k_{21} - 3, s_1s_2 + s_1s_3 + s_2s_3 + 4k_{11} + 2k_{21} - 3, -s_1s_2s_3 + 2k_{11} + k_{21} - 1 \rangle.$$

Poznamenejme, že soustava je v tomto jednoduchém případě lineární soustavou vzhledem k k_{ij} , a lze tedy snadno vyřešit bez použití *GB*. Nicméně pro porovnání s dalším příkladem využijeme i zde *GB*. Z teorie řízení víme, že úplné přiřazení pólů nemusí být pomocí výstupní zpětné vazby možné, [88]. Protože ideál I obsahuje z matice K pouze parametry k_{11}, k_{21} , spočteme *GB* vzhledem k uspořádání $k_{11} > k_{21}$. Zjistíme, že $G = \{1\}$ a soustava nemá řešení. Z toho plyne, že není možné zvolit všechny póly libovolně. Nejméně jeden pól nebude možné libovolně umístit. Přidejme jeden pól do množiny proměnných a spočteme *GB* znovu vzhledem k uspořádání $k_{11} > k_{21} > s_1$. Gröbnerova báze se nezměnila, $G = \{1\}$. To znamená, že další pól není možné ovlivnit. Přidejme další pól. Pro proměnné $k_{11} > k_{21} > s_1 > s_2$ konečně dostáváme Gröbnerovu bázi obsahující řešení

$$G = \{s_2^2 + 2s_2 + 1, 2 + s_1 + s_2, -s_3 + 2k_{11} + k_{21} - 1\}. \tag{2.41}$$

Z prvních dvou rovnic vidíme, že dva póly jsou fixovány na hodnotách $s_{1,2} = -1$. Třetí pól je možné umístit pomocí dvou parametrů k_{11}, k_{21} zpětnovazební matice K . Dostáváme tedy parametrizaci všech přípustných výstupních zpětných vazeb pro s_3

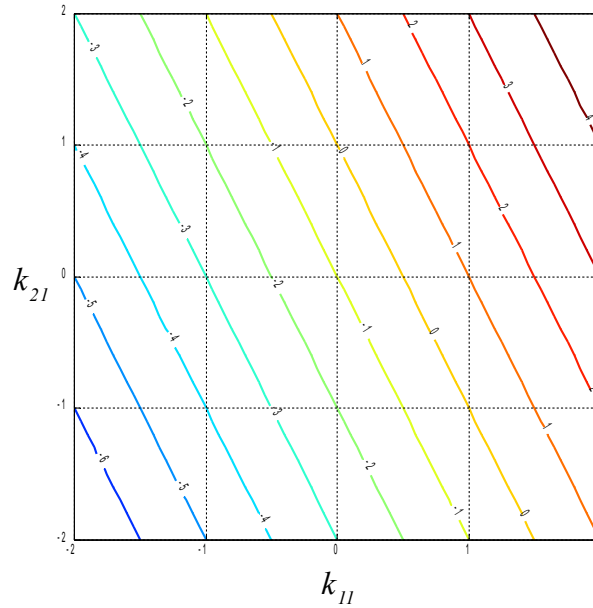
$$K = \begin{bmatrix} k_{11} & k_{12} \\ -2k_{11} + s_3 + 1 & k_{22} \end{bmatrix}$$

Vykreslením třetí rovnice (2.41) dostáváme parametrickou závislost hodnot třetího pólu s_3 na parametrech k_{11}, k_{21} , obrázek 2.9.

Pro kontrolu spočteme Jordanovu formu matice A_{CL}

$$J_{CL} = \begin{bmatrix} 2k_{11} + k_{21} - 1 & 0 & 0 \\ 0 & -1 & 1 \\ 0 & 0 & -1 \end{bmatrix}$$

Zjistíme, že výstupní zpětnou vazbou je opravdu možné ovlivnit pouze jedno vlastní číslo.



Obrázek 2.9: Závislost třetího pólu na parametrech zpětnovazební matice. Konturové přímky definují ekvipotenciální hladiny velikosti pólu s_3 .

Příklad. Pokud modifikujeme vstupní matici do tvaru

$$B = \begin{bmatrix} 1 & 1 \\ 0 & 0 \\ 0 & 1 \end{bmatrix},$$

dostaneme matici uzavřeného systému, při zachování ostatních požadavků, ve tvaru

$$A_{CL} = \begin{bmatrix} k_{11} + k_{21} - 1 & k_{11} + k_{21} & k_{12} + k_{22} \\ 0 & -1 & 0 \\ k_{21} & k_{21} & -1 + k_{22} \end{bmatrix}.$$

Opět dostáváme ideál příslušející polynomiální soustavě ve tvaru

$$I = \left\langle \begin{array}{l} -s_1 - s_2 - s_3 + k_{11} + k_{21} + k_{22} - 3, \\ s_1 s_2 + s_1 s_3 + s_2 s_3 + 2(k_{11} + k_{21} + k_{22}) - k_{11} k_{22} + k_{21} k_{12} - 3, \\ -s_1 s_2 s_3 - 1 + k_{11} + k_{21} + k_{12} + k_{21} k_{22} - k_{11} k_{22} \end{array} \right\rangle,$$

kde rovnice již nejsou lineárními vzhledem k parametřům k_{ij} . V tomto případě má soustava řešení již pro proměnné k_{11}, k_{21}, s_1 a pro uspořádání $k_{11} > k_{21} > s_1$ dostáváme Gröbnerovu bázi

$$G = \left\langle \begin{array}{l} s_1 + 1, \\ k_{21} k_{12} + k_{21} k_{22} + k_{22}^2 - k_{22} s_2 - k_{22} s_3 - 2k_{22} + s_3 + 1 + s_3 s_2 + s_2, \\ k_{11} + k_{21} + k_{22} - s_2 - s_3 - 2 \end{array} \right\rangle. \quad (2.42)$$

Pól s_1 je opět fixován, $s_1 = -1$. Pro s_2, s_3 dostáváme z (2.42) tři řešení

$$k_{12} = \frac{-(k_{11}^2 + k_{11}(-2 + k_{21} - s_2 - s_3) + s_3 + 1 + s_3 s_2 + s_2)}{k_{21}}, \quad k_{11} = k_{11}, \quad k_{21} = k_{21}, \quad k_{22} = -k_{11} - k_{21} + s_2 + s_3 + 2$$

$$k_{21} = 0, \quad k_{11} = s_3 + 1, \quad k_{22} = s_2 + 1, \quad k_{12} = k_{12}$$

$$k_{21} = 0, \quad k_{11} = s_2 + 1, \quad k_{22} = s_3 + 1, \quad k_{12} = k_{12}$$

přičemž druhá dvě jsou vzájemně symetrická vzhledem k symetrii úlohy. Budeme tedy uvažovat pouze první dvě. Vidíme, že první řešení je existenčně závislé na parametru k_{21} . Dostáváme zpětnovazební matici ve tvaru

$$K_1 = \begin{bmatrix} k_{11} & \frac{-(k_{11}^2 + k_{11}(-2 + k_{21} - s_2 - s_3) + s_3 + 1 + s_3 s_2 + s_2)}{k_{21}} \\ k_{21} \neq 0 & -k_{11} - k_{21} + s_2 + s_3 + 2 \end{bmatrix}.$$

Druhé řešení je specializací prvního řešení

$$K_2 = \begin{bmatrix} s_3 + 1 & k_{12} \\ 0 & s_2 + 1 \end{bmatrix}.$$

Se specializací jsme se již setkali v kap. 2.6.1. Matice K_1, K_2 spolu tvoří kompletní parametrizaci všech zpětných vazeb pro s_2, s_3 .

V motivačních příkladech jsme ukázali, jakým způsobem aplikovat teorii GB pro jednodušší problémy. Pro obecný postup je možné využít dřívějších výsledků z [88], a spojit Sylvestrovu maticovou rovnici s teorií GB . Tímto směrem se ubírá práce [54] a my v následujícím textu pouze nastíníme tento přístup a ukážeme jedno z možných řešení pro speciální typ parametrické matice. Úplný popis je uveden v [90].

Uvažujme následující problém. Jsou dány matice $A \in \mathbb{R}^{n \times n}$, $B \in \mathbb{R}^{n \times m}$, $C \in \mathbb{R}^{p \times n}$ popisující lineární dynamický systém. Bez ztráty obecnosti se omezme na případ, kdy dvojice (A, B) je říditelná, a dvojice (C, A) je pozorovatelná. Ptáme se, zda je možné nalézt výstupní zpětnou vazbu $u(t) = K y(t)$, která zaručí, že dynamika uzavřeného systému bude popsána zvolenou maticí $L \in \mathbb{R}^{n \times n}$ v Jordanově formě. Hledáme tedy množinu

$$\kappa(A, B, C, L) = \{K \in \mathbb{R}^{(m \times p)} : (A + BKC) \sim L\}.$$

Převedením do tvaru rovnosti $A + BKC = X L X^{-1}$ a substitucí $H = K C X(H)$, $H \in \mathbb{R}^{m \times n}$ dostáváme známou Sylvestrovu rovnici

$$A X(H) - X(H)L + B H = 0. \quad (2.43)$$

Věta 2.15. *Nechť $\sigma(A) \cap \sigma(L) = \emptyset$ a $\kappa(A, B, C, L) \neq \emptyset$. Potom řešení Sylvestrové rovnice (2.43), $X(H)$ má plnou hodnotu pro skoro každé $H \in \mathbb{R}^{m \times n}$. Libovolná matice $K \in \kappa(A, B, C, L)$ může být vyjádřena vztahem $H [X^T(H) X(H)]^{-1} X^T(H) = K C$.*

Matici H můžeme dále nahradit vhodně zvolenou parametrickou maticí $Q(\alpha)$ tak, že

$$\begin{aligned} Q(\alpha)Z &= K C X(H) \\ Q(\alpha) &= K C X(H) Z^{-1} \\ Q(\alpha) &= K C X(\alpha), \end{aligned} \quad (2.44)$$

kde $X(\alpha)$ je řešením Sylvestrové rovnice

$$A X(\alpha) - X(\alpha)L + B Q(\alpha) = 0. \quad (2.45)$$

Matice $Q(\alpha)$ je vybrána s ohledem na minimální počet parametrů, nutný pro zachování všech řešení. Konstrukce této matice je relativně složitá, a proto čtenáře odkážeme na [88]. Poznamenejme, že pro cyklickou matici L , kdy každému vlastnímu číslu přísluší pouze jeden Jordanův blok, můžeme parametrickou matici $Q(\alpha)$ sestavit podle následující předlohy

$$Q(\alpha) = \begin{bmatrix} 1 & 1 & \dots & 1 \\ * & * & \dots & * \\ \vdots & \vdots & \dots & \vdots \\ * & * & \dots & * \end{bmatrix},$$

kde *-ky značí volné parametry $Q(\alpha)$ a jejich počet je dán vztahem $(m-1)n$. První řádek zajišťuje pozorovatelnost dvojice $(Q(\alpha), L)$ pro libovolné $\alpha \in \mathbb{R}^r$.

Rovnice (2.44) reprezentuje soustavu mn polynomiálních rovnic o $(m-1)n+mp$ neznámých. Existence řešení je podmíněna podle Frobeniovy věty následujícím vztahem

$$\text{rank}[CX(\alpha)] = \text{rank} \begin{bmatrix} CX(\alpha) \\ Q(\alpha) \end{bmatrix}.$$

Řešení lze opět hledat pomocí GB.

Příklad. Uvažujme systém jehož stavový popis je

$$A = \begin{bmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{bmatrix}, \quad B = \begin{bmatrix} 1 & 1 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad C = \begin{bmatrix} 1 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

Hledáme explicitní parametrizaci všech výstupních zpětných vazeb $K = \begin{bmatrix} k_{11} & k_{12} \\ k_{21} & k_{22} \end{bmatrix}$ přiřazujících cyklickou matici

$$L = \begin{bmatrix} -1 & 0 & 0 \\ 0 & -2 & 0 \\ 0 & 0 & -3 \end{bmatrix}.$$

Sestrojíme matici $Q(\alpha) = \begin{bmatrix} 1 & 1 & 1 \\ \alpha_1 & \alpha_2 & \alpha_3 \end{bmatrix}$ a vyřešíme Sylvestrovu rovnici (2.45). Řešením je

$$X(\alpha) = \begin{bmatrix} -1/4 - 5/8\alpha_1 & -2/9 - 10/27\alpha_2 & -3/16 - 17/64\alpha_3 \\ -1/2 + 1/4\alpha_1 & -1/3 + 1/9\alpha_2 & -1/4 + 1/16\alpha_3 \\ -1/2\alpha_1 & -1/3\alpha_2 & -1/4\alpha_3 \end{bmatrix}.$$

Sestrojíme rovnici (2.44) ve tvaru $Q(\alpha) - KCX(\alpha) = 0$. Dostáváme 6 dílčích rovnic, které tvoří množinu generátorů

$$F = \left\{ \begin{array}{l} 1 - k_1 \left(-\frac{1}{4} - \frac{5}{8}\alpha_1 \right) - k_1 \left(-\frac{1}{2} + \frac{1}{4}\alpha_1 \right) + \frac{1}{2}k_2\alpha_1, \\ 1 - k_1 \left(-\frac{2}{9} - \frac{10}{27}\alpha_2 \right) - k_1 \left(-\frac{1}{3} + \frac{1}{9}\alpha_2 \right) + \frac{k_2}{3}\alpha_2, \\ 1 - k_1 \left(-\frac{3}{16} - \frac{17}{64}\alpha_3 \right) - k_1 \left(-\frac{1}{4} + \frac{1}{16}\alpha_3 \right) + \frac{k_2}{4}\alpha_3, \\ \alpha_1 - k_3 \left(-\frac{1}{4} - \frac{5}{8}\alpha_1 \right) - k_3 \left(-\frac{1}{2} + \frac{1}{4}\alpha_1 \right) + \frac{k_4}{2}\alpha_1, \\ \alpha_2 - k_3 \left(-\frac{2}{9} - \frac{10}{27}\alpha_2 \right) - k_3 \left(-\frac{1}{3} + \frac{1}{9}\alpha_2 \right) + \frac{k_4}{3}\alpha_2, \\ \alpha_3 - k_3 \left(-\frac{3}{16} - \frac{17}{64}\alpha_3 \right) - k_3 \left(-\frac{1}{4} + \frac{1}{16}\alpha_3 \right) + \frac{k_4}{4}\alpha_3 \end{array} \right\}. \quad (2.46)$$

Vyřešením této soustavy dostáváme koeficienty matice K v proměnných parametrech α_i

$$k_1 = \frac{-2(\alpha_1 + 22)}{3\alpha_1 - 2}, \quad k_2 = \frac{1}{2} \frac{3\alpha_1^2 + 60\alpha_1 + 140}{\alpha_1(3\alpha_1 - 2)}, \quad k_3 = \frac{-68\alpha_1}{3\alpha_1 - 2}, \quad k_4 = \frac{45\alpha_1 + 106}{3\alpha_1 - 2}$$

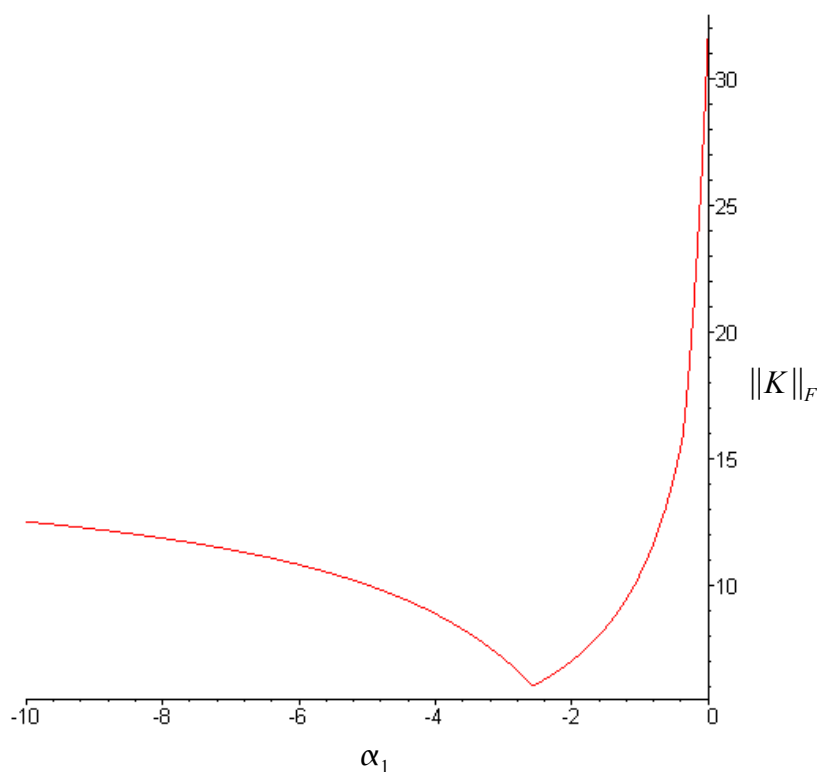
$$\alpha_2 = \frac{102\alpha_1}{\alpha_1 + 90}, \quad \alpha_3 = \frac{68\alpha_1}{\alpha_1 + 56}$$

Zjišťujeme, že parametry α_2, α_3 jsou funkcí parametru α_1 . Matice výstupní zpětné vazby

$$K = \begin{bmatrix} \frac{-2(\alpha_1 + 22)}{3\alpha_1 - 2} & \frac{1}{2} \frac{3\alpha_1^2 + 60\alpha_1 + 140}{\alpha_1(3\alpha_1 - 2)} \\ \frac{-68\alpha_1}{3\alpha_1 - 2} & \frac{45\alpha_1 + 106}{3\alpha_1 - 2} \end{bmatrix}$$

je tedy parametrizována pouze jedním volným parametrem α_1 , $\alpha_1 \neq 2/3$.

Můžeme tedy například hledat takovou hodnotu parametru α_1 , která zajistí minimální Frobeniovu normu $\min_{\alpha_1} \|K(\alpha_1)\|_F = \min_{\alpha_1} \sqrt{\text{tr}(K(\alpha_1) \cdot K(\alpha_1))}$ matice K , viz obrázek 2.10.



Obrázek 2.10: Závislost Frobeniovu normy $\|K\|_F$ na parametru α_1 .

3 ČASOVĚ OPTIMÁLNÍ ŘÍZENÍ ŘETĚZCE INTEGRÁTORŮ JAKO GENERÁTORU POHYBU

Tato kapitola se zaměřuje na generování pohybu s respektováním všech pohybových omezení a obsahuje hlavní výsledky disertační práce. Kritériem kvality je časová optimalita a problém generování trajektorií pohybu je převeden na problém časově optimálního řízení s omezením na stav a vstup systému. V úvodu je definována standardní úloha t-optimálního řízení včetně nastínění standardních postupů řešení. Dále je upozorněno na některé související problémy a odkázáno na literaturu s detailnějším zpracováním. Další část práce se již zaměřuje na řízení konkrétního systému, a to na řetězec dvou a tří integrátorů, jako vhodného generátoru trajektorie polohy, rychlosti a zrychlení. Odvozeny jsou tři standardní druhy generování pohybu, a to *z klidu do klidu*, *z konstantní do konstantní rychlosti* a *obecný přesun*. Generátor respektuje ve všech případech konstantní pohybová omezení.

3.1 OBECNÝ PROBLÉM OPTIMÁLNÍHO ŘÍZENÍ

Cílem optimálního řízení je modifikovat dynamiku systému tak, aby nějaké kvantum, například množství potřebného paliva, bylo minimalizováno. Přesněji řečeno, aby účelový funkcionál, podléhající dynamice systému, počátečním podmínkám a omezením, byl minimalizován. Toho je možné docílit pouze vhodným působením vstupů do systému.

Definice 3.1. *Nechť je dán dynamický systém, popsáný diferenciální rovnicí*

$$\dot{\mathbf{x}} = f(\mathbf{x}, \mathbf{u}, t), \quad (3.1)$$

kde \mathbf{x} je n -dimenzionální vektor stavu systému, \mathbf{u} je m -dimenzionální vektor řízení, a f je funkce, pro kterou platí $f: \mathbb{R}^n \times U \times \mathbb{R} \rightarrow \mathbb{R}^n$ je třídy C^1 pro všechny argumenty. Přejít systému z jednoho stavu do druhého je nazýván pohybem. Nechť je počáteční stav a koncový stav označen jako

$$\mathbf{x}(t_0) = \mathbf{x}_0, \quad \mathbf{x}(t_f) = \mathbf{x}_f$$

a kritérium kvality pohybu bude dáno hodnotou funkcionálu

$$\Psi(x_f, t_f) + \int_0^{t_f} L(x, u, t) dt, \quad (3.2)$$

kde $\Psi(x_f, t_f)$ je dodatečné ohodnocení koncového stavu, t_f je konečný čas přesunu a L - Lagrangian je funkce stejných vlastností jako f .

Problém 3.1 (optimal control). Dán dynamický systém popsáný rovnicí (3.1) a funkcionál popsáný vztahem (3.2). Problémem optimálního řízení je najít mezi všemi přípustnými tvary řízení takové, které převede systém z počátečního stavu \mathbf{x}_0 do stavu koncového \mathbf{x}_f a minimalizuje daný funkcionál.

3.2 OBECNÝ PROBLÉM T-OPTIMÁLNÍHO ŘÍZENÍ

Jestliže budeme chtít minimalizovat čas přesunu systému mezi počátečním stavem \mathbf{x}_0 a koncovým stavem \mathbf{x}_f , potom budeme minimalizovat funkcionál (kritérium kvality) (3.2) ve tvaru

$$\min_{u(\cdot)} \int_0^{t_f} 1 dt. \quad (3.3)$$

Uvažujme lineární t-invariantní systém s jedním vstupem, zapsaný ve stavové reprezentaci

$$\dot{\mathbf{x}} = A\mathbf{x} + b u, \quad (3.4)$$

kde \mathbf{x} značí stav systému, u vstup řízení a dvojice (A, b) jsou konstantní matice příslušných rozměrů. Vývoj stavu takového lineárního systému je dán následující rovnicí

$$\mathbf{x}_f = e^{A t_f} \mathbf{x}_0 + \int_0^{t_f} e^{A(t_f - \tau)} b u(\tau) d\tau. \quad (3.5)$$

kde \mathbf{x}_0 a \mathbf{x}_f je počáteční a koncový stav systému.

Pokud se budeme zajímat o přesun systému z libovolného počátečního stavu do libovolného koncového stavu za minimální čas, můžeme využít výsledků *Potryaginova principu maxima* [4]. Pro úplnost nastíníme obecný postup.

Nechť je dán systém $\dot{\mathbf{x}} = f(\mathbf{x}, u)$, počáteční a koncový stav, \mathbf{x}_0 a \mathbf{x}_f . Hledáme takové řízení u^* , které převede systém z \mathbf{x}_0 do \mathbf{x}_f a přitom minimalizuje kritérium (3.3). Z *Pontryaginova principu maxima* dostaneme, že optimální řízení musí splňovat nutnou a postačující podmínku ve tvaru

$$\frac{\partial H(\mathbf{x}, \mathbf{p}, u^*)}{\partial u} \equiv 0 \quad (3.6)$$

kde *Hamiltonian* $H(\mathbf{x}, \mathbf{p})$ [76] je v případě t-optimálního řízení definován jako

$$H(\mathbf{x}, \mathbf{p}) = \mathbf{p} \cdot f(\mathbf{x}, u) - L, \quad L = 1$$

a \mathbf{p} je takzvaným kostavem (vektorem lagrangeových multiplikátorů) vyplývajícím z hamiltonových kanonických rovnic [1],[76]

$$\frac{\partial H}{\partial \mathbf{x}} = -\dot{\mathbf{p}} \quad \frac{\partial H}{\partial \mathbf{p}} = \dot{\mathbf{x}}.$$

Pro systém s omezeným vstupem $|u| \leq u_{max}$ bychom dostali, že úloha t-optimálního řízení vede

na tzv. *bang-bang* strategii řízení [4]. Řízení se bude skokově měnit mezi jeho hraničními hodnotami. Tímto postupem je možné získat explicitní zákon řízení pro lineární systémy bez stavových omezení, viz kapitola 3.2.1. Bohužel pro systémy se stavovými omezeními, nebo složitější strukturou je zpravidla vždy nutné využít numerických metod.

Problém časově optimálního řízení lze formulovat jako dvou-bodový problém, který lze řešit metodou střelby. Nevýhodou je, že oblast konvergence metod střelby je velmi úzká. Je nutné udělat velmi přesný počáteční odhad optimálního stavu, řízení a vektoru kostavu, který je těžké získat, ale je nutný pro stabilitu algoritmu. Navíc je velmi těžké odhadnout rychlost algoritmu, neboť ta je závislá na přesnosti počátečního odhadu a na požadované přesnosti řešení.

3.2.1 ŘETĚZEC INTEGRÁTORŮ A T-OPTIMÁLNÍ ŘÍZENÍ

Pokud uvažujeme úlohu t -optimálního řízení řetězce integrátorů s omezením pouze na vstup systému

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \vdots \\ \dot{x}_{n-1} \\ \dot{x}_n \end{bmatrix} = \underbrace{\begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \\ 0 & 0 & 0 & \cdots & 0 \end{bmatrix}}_A \underbrace{\begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_{n-1} \\ x_n \end{bmatrix}}_x + \underbrace{\begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix}}_b u, \quad |u| \leq u_{max}, \quad u_{max} \in \mathbb{R}^+$$

je možné nalézt explicitní zpětnou vazbu pomocí klasických metod optimálního řízení uvedených výše. Víme, že pro takový systém t -optimální řízení s omezeným vstupem vede na tzv. *bang-bang* řízení [4],[17], kde maximální počet časových intervalů pro které je řízení konstantní, nebo chcete-li počet přepnutí, je dán řádem systému n .³

Zpětnovazební explicitní vztahy jsou již pro druhý

$$u = -\operatorname{sgn}\left[|x_1|^{\frac{1}{2}} \operatorname{sgn}(x_1) + 2^{-\frac{1}{2}} x_2\right]$$

a třetí řád

$$u = -\operatorname{sgn}\left[x_1 + \frac{1}{3} x_3^3 + x_2 x_3 \operatorname{sgn}\left(x_2 + \frac{1}{2} x_3 |x_3|\right) + \left\{\frac{1}{2} x_3^2 + x_2 \operatorname{sgn}\left(x_2 + \frac{1}{2} x_3 |x_3|\right)\right\}^{\frac{3}{2}} \operatorname{sgn}\left(x_2 + \frac{1}{2} x_3 |x_3|\right)\right]$$

komplikované a mnoho studií bylo provedeno k zjednodušení pro počítačové výpočty, například převedení do tvaru, který neobsahuje odmocniny [41]. Pro řetězec integrátorů 2. a 3. řádu dostaneme

$$u = -\operatorname{sgn}\left[x_1 + \frac{1}{2} x_2 |x_2|\right],$$

$$u = -\operatorname{sgn}\left[x_1 + \frac{1}{3} x_3^3 + x_2 x_3 \operatorname{sgn}\left(x_2 + \frac{1}{2} x_3 |x_3|\right) \cdot \left|x_1 + \frac{1}{3} x_3^3 + x_2 x_3 \operatorname{sgn}\left(x_2 + \frac{1}{2} x_3 |x_3|\right)\right| + \left\{x_2 + \frac{1}{2} x_3^2 \operatorname{sgn}\left(x_2 + \frac{1}{2} x_3 |x_3|\right)\right\}^3\right].$$

Vznikaly i další přístupy zjednodušující nelineární t -optimální zpětnou vazbu pro řetězec integrátorů. Persson [79] zavedl sub-optimální regulátor ve tvaru

³ V literatuře je také uváděno, že počet přepnutí je rovno $n-1$, ale my budeme pro konzistenci uvažovat že na konci řídicího procesu nastavíme řízení na nulu a tedy počet přepnutí je shodný s počtem časových intervalů.

$$u = \operatorname{sgn} v \quad (v \neq 0)$$

$$v = f_1(x_1, \cdot) + f_2(x_2, \cdot) + \dots + f_n(x_n, \cdot),$$

kde

$$f_i(x_i, a) = \begin{cases} K_i |a x_i|^{H/(n+1-i)} \operatorname{sgn} x_i & (x_i \neq 0) \\ 0 & (x_i = 0) \end{cases} \quad (i=1, 2, \dots, n),$$

čímž zjednodušil výsledný tvar zpětnovazební funkce. Optimální hodnoty parametrů H, K_i hledal empiricky pomocí analogového počítače. Ukázal, že pro řetězec integrátorů 1. a 2. řádu je sub-optimální regulátor roven optimálnímu. Také zavedl jakési měřítkování, které nazval *special scaling property*. Využitím parametru $\lambda > 0$ parametrizoval všechny počáteční stavy, které přejdou podobným řízením a tedy po podobných trajektoriích

$$\mathbf{x} = \lambda^{-n} \boldsymbol{\theta}(\lambda t)$$

do počátku. Tento přístup byl posléze zobecněn A. T. Fullerem do tzv. *general scaling property*, [39]. Fuller také ukázal, že sub-optimální regulátor prodlouží potřebný čas k převedení systému do počátku asi o 30%. Dále se zabýval nestabilitou a jakýmsi klouzavým režimem, zvaným *Fuller's phenomenon*, který nastává při uvažování specifických kritérií kvality [15],[101]. Fuller dále využil Perssonův sub-optimální regulátor a pro řetězec tří integrátorů převedl stavový prostor řízeného systému

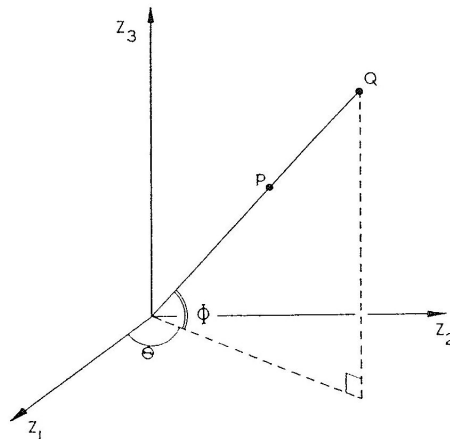
$$\begin{aligned} \mathbf{z} &= \dot{\mathbf{x}}, \\ z_1 &= |a x_1|^{1/3} \operatorname{sgn} x_1 \\ z_2 &= |a x_2|^{1/2} \operatorname{sgn} x_2 \\ z_3 &= a x_3 \end{aligned}$$

do redukovaného prostoru sférických souřadnic (Θ, Φ)

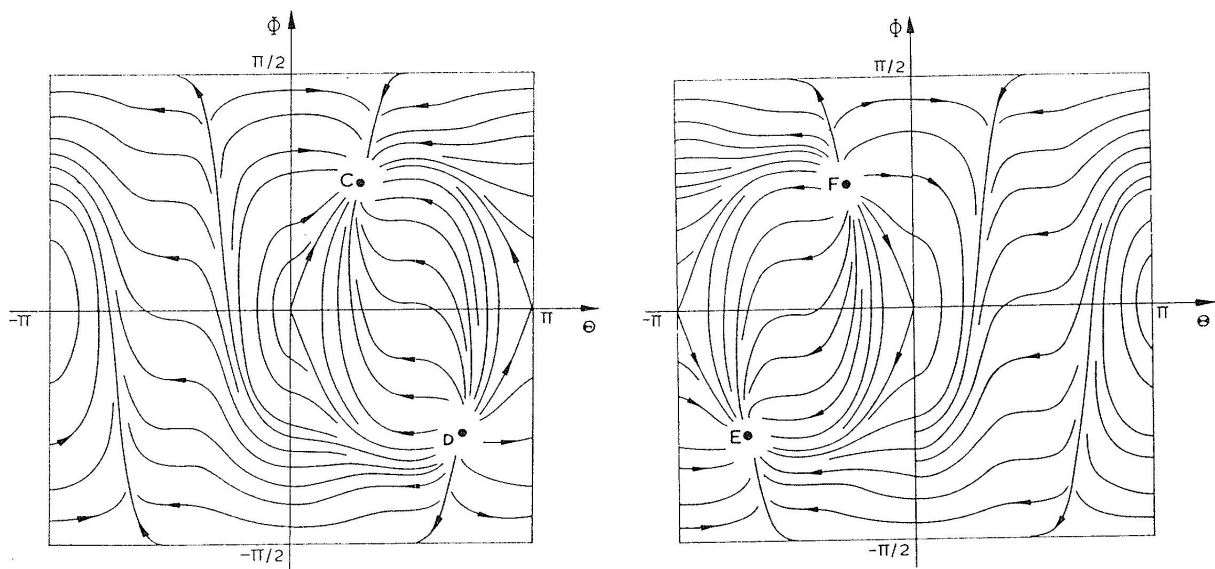
$$\tan \Theta = \frac{z_2}{z_1}$$

$$\tan \Phi = \frac{z_3}{(z_1^2 + z_2^2)^{1/2}},$$

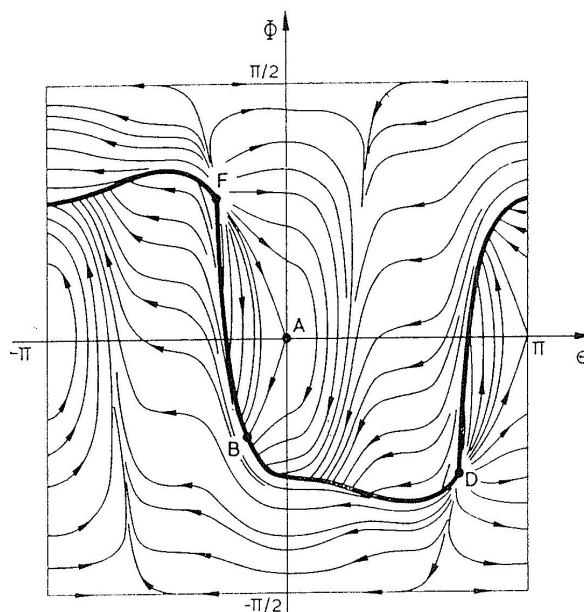
kde se všechny body paprsku vedeného z počátku promítnou na jeden bod sféry, [40]. Tento postup také pracuje s měřítkováním, neboť všechny body podobných trajektorií, například bod $P(z_1, z_2, z_3)$ a $Q(\bar{z}_1, \bar{z}_2, \bar{z}_3)$ viz Obrázek 3.1, mají stejné úhlové souřadnice (Θ, Φ) .



Obrázek 3.1: Projekce třídímního prostoru na povrch sféry.



Obrázek 3.2: Trajektorie systému 3 integrátorů při $u_+(t)$ a $u(t)$ řízení v projekčním prostoru (Θ, Φ)



Obrázek 3.3: Výsledná trajektorie v projekčním prostoru (Θ, Φ) .

Je tedy možné hledat takové množiny počátečních stavů, které vedou na podobné sekvence řízení, obrázek 3.2 a obrázek 3.3. Hledáme jakési zoomy trajektorií. Tento postup se jeví jako velice zajímavý, i když autoři se omezují pouze na nízké řády systému a varují před složitostí. Bohužel vše je řešeno pouze pro řetězec bez omezení a koncovým stavem v počátku. Zdá se ale, že určitá zobecnění by byla možná. Vyžadují ovšem podrobnější zkoumání. E.P.Ryan [83],[84] dále zobecnil práci Fullera pro třetí řád a *time-fuel-optimal* řízení, minimalizující funkcionál

$$\min_{u(\cdot)} \int_0^{t_f} \lambda + (1-\lambda)|u(t)| dt, \quad \lambda \in (0,1].$$

Zabýval se také vlivem perturbací na optimální zpětnou vazbu [85]. Bohužel všechny výše zmíněné přístupy popisují zpětnou vazbu, která nerespektuje omezení na stavové proměnné.

3.2.2 PROBLÉM OBECNÉHO KONCE V T-OPTIMÁLNÍM ŘÍZENÍ

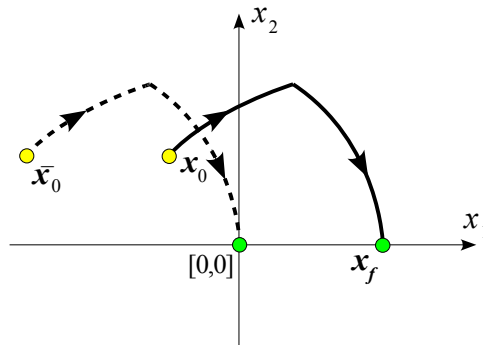
Mnoho nástrojů, které jsou využívány pro řešení t -optimální úlohy, zjednodušuje problém v tom smyslu, že koncový stav \mathbf{x}_f do kterého je systém řízen, je jedinečně počátek stavového prostoru. Pro plánování a napojování řídicích trajektorií je ovšem nutné, aby koncový stav mohl nabývat libovolných hodnot z dané množiny přípustných stavů. Nabízí se tedy otázka, zda je možné transformovat \mathbf{x}_0 a \mathbf{x}_f tak, abychom mohli tyto nástroje použít. Zavedením nového počátečního stavu $\bar{\mathbf{x}}_0$ a vyjádřením transformace ze vztahu (3.5) ale vyplývá

$$\underbrace{\mathbf{x}_f - e^{At_f} \mathbf{x}_0}_{-e^{At_f} \bar{\mathbf{x}}_0} = \int_0^{t_f} e^{A(t_f-\tau)} B u(\tau) d\tau,$$

že tato transformace $\bar{\mathbf{x}}_0 = \mathbf{x}_0 - e^{-At_f} \mathbf{x}_f$ je obecně závislá na času potřebném k přesunu systému, který je ale v t -optimálním řízení neznámou. Například pro řetězec integrátorů 2. řádu dostaneme

$$\bar{\mathbf{x}}_0 = \mathbf{x}_0 - \begin{bmatrix} 1 & -t_f \\ 0 & 1 \end{bmatrix} \mathbf{x}_f = \begin{bmatrix} x_{01} \\ x_{02} \end{bmatrix} - \begin{bmatrix} x_{f1} - t_f x_{f2} \\ x_{f2} \end{bmatrix}.$$

Pokud se tedy koncový stav \mathbf{x}_f nachází ve stavovém prostoru pouze na ose stavu x_1 , neboli $\mathbf{x}_f = [x_{f1} \ 0]^T$ a jedná se o řízení *do klidu*, potom je transformace možná. Ze stavového prostoru je zřejmé, že transformací dojde pouze k horizontálnímu posunu stavové trajektorie, obrázek 3.4.



Obrázek 3.4: Transformace počátečního a koncového stavu.

Pro obecný problém generování trajektorie pohybu je transformace koncového stavu do počátku nemožná. Bylo by možné využít iterační proces a metodou střelby nalézt vhodnou polohu $\bar{\mathbf{x}}_0$, ale vše se zdá být více heuristikou než sofistikovaným a exaktním nástrojem. Je tedy rozumnější uvažovat problém v jeho původní podobě a nalézt obecné řešení, viz. dále.

3.3 ŘETĚZEC INTEGRÁTORŮ S OMEZENÍMI JAKO GENERÁTOR T-OPTIMÁLNÍHO POHYBU

Řetězec integrátorů je velmi zajímavým druhem systému z hlediska plánování a řízení pohybu. Zabýváme-li se generováním a řízením pohybu jakéhokoliv systému, zajímá nás jeho poloha, rychlost a zrychlení, případně další vyšší derivace, jako například derivace zrychlení – *Jerk*. Tyto veličiny mají mezi sebou diferenciální závislost zrovna tak, jako jednotlivé stavy řetězce integrátorů. Tímto způsobem můžeme převést původní úlohu plánování pohybu zpravidla nelineárního systému na problém plánování ve stavovém prostoru řetězce integrátorů. Tento přístup

je velmi využíván, například v [13],[87],[93]. Je založen na faktu, že v případě t-optimálního řízení je tvar řízení apriori specifikován. Podmínky optimality vyžadují, aby v každém časovém okamžiku byla alespoň jedna složka stavu nebo vstupu na hranici omezení, řídicí strategie musí mít tzv. *bang-bang* strukturu nebo obsahovat tzv. *singulární oblouky* [112].

Časově optimální plánování pohybu robota lze tedy charakterizovat jako problém časově optimálního řízení pro lineární systém řetězce integrátorů (s,v,a) s nelineárními omezeními. Dynamika systému-generátoru je dána řetězcem integrátorů a z pohledu plánování pohybu je chápána jako diferenciální omezení. Předdefinovaná cesta v pracovním prostoru stroje nebo v prostoru kloubových souřadnic, podél které se má stroj pohybovat, může být vyjádřena jako geometrické nelineární omezení. Omezení na rychlost, zrychlení, případně derivaci zrychlení (*Jerk*) vyplývají z dynamiky a kinematiky původního systému a jsou nazývána kino-dynamickými omezeními. Další požadavky na pohyb, například ve formě omezení na maximální rychlost koncového efektoru, jsou nazývány fázovými omezeními. Výsledkem těchto předpokladů je optimalizační úloha s omezeními typu rovnosti i nerovnosti (*equality and inequality constraints*).

Plánování pohybu dynamických systémů, které musí respektovat omezení typu rovnosti nebo nerovnosti je velmi obtížným problémem. Ještě obtížnější může být nalezení důkazu, že řešení vůbec existuje, nebo že minimalizuje dané kritérium kvality. Jeden z prvních výsledků t-optimálního řízení s omezením na vstupní proměnné a jejich strmost je uveřejněn v [26]. Zde je zdůrazněn vliv tvaru trajektorie zrychlení na říditelnost mechanického systému. Jako příklad je uváděn vliv nespojitosti zrychlení při změně natočení křídélek a směrůvek letadla na celkovou říditelnost a stabilitu stroje. Z tohoto článku také vyplývá a je dokázáno, že pro řetězec tří integrátorů s konstantními omezeními na vstup a zrychlení, musí mít řízení tzv. *bang-zero-bang* strukturu, uvedenou pro jiný druh systému v [17].

V moderní literatuře se setkáváme s dvěma výraznými směry využití t-optimálního řízení pro plánování pohybu.

První z nich je věnován robotickým manipulátorům. Standardní úlohou v této oblasti je pak problém plánování pohybu podél specifikované cesty s omezeními na síly a momenty aktuátorů. Ačkoli jsou tato omezení na straně aktuátorů většinou uvažována za konstantní, v důsledku nelinearit a složitých kinematických transformací dostaneme problém plánování podél specifické cesty s nekonstantními omezeními na pohybové veličiny, viz motivační příklad v úvodu, kapitola 1.1.2. Příkladem jsou práce [13],[112],[113],[93]. V těchto článcích je předpokládáno, že zrychlení (síla, moment) je přímo řídicí veličinou a lze ho skokově měnit. Problém je převeden do stavového prostoru řetězce dvou integrátorů s nelineárními omezeními typu rovnosti i nerovnosti. Je zde aplikována strategie *bang-bang* řízení a přímou integrací s pomocí metod střelby, nebo podobných algoritmů, jsou hledány přepínací body trajektorie vektoru stavu. Nevýhodou těchto přístupů je časová náročnost výpočtu, neboť je nutné v každém cyklu numericky řešit pohybové rovnice systému. Počet iterací metody střelby je dán přesností hledaného výsledku a nelze ho spolehlivě odhadnout.

Další výrazná skupina prací se věnuje problému plánování pohybu v jedné ose s konstantními nebo po částech konstantními omezeními na dílčí složky pohybu. Příkladem jsou [57],[62], kde je opět využíváno řetězce integrátorů a t-optimální strategie řízení. Mnoho prací se snaží nalézt algoritmy pro třetí řád s omezením na derivaci zrychlení pro různé druhy přesunu, jako například pohyb z klidu do klidu [43], z/do konstantního pohybu a dalších obecnějších variant [55],[56],[57]. V pracích je zaveden termín S-profil (S-křivka), který vyplývá ze strategie *bang-zero-bang* řízení systému s konstantními omezeními. Poznamenejme, že tato strategie je velmi rozšířena a využívána, přestože exaktní důkaz o časové optimalitě není v dostupné literatuře uveden⁴. Pouze pro pohyb z klidu do klidu lze jednoduše ukázat, že tato strategie řízení generuje t-optimální přesun

4 O složitosti důkazů v optimálním řízení a jejich nepřesnostech pojednává článek [42].

systemu. Autoři se také věnují problému přeplánování trajektorií během pohybu systému. Zabývají se standardními úlohami z normy PLCopen jako je *blending*, *aborting*, *buffered*, [55], a plánováním víceosých systémů [32]. V současnosti se pro řešení optimalizačních úloh čím dál více využívá numerických a heuristických metod, především z důvodu nárůstu výpočetního výkonu.

Existují i obecné postupy optimálního řízení založené na modifikovaném principu maxima, do kterého je možné zahrnout omezení v implicitním tvaru [42]. Bohužel tento přístup není pro plánování pohybu vhodný, protože standardní omezení pohybu v praktických aplikacích nelze do tohoto tvaru převést.

Následující podkapitoly řeší problém plánování pohybu z klidu do klidu a jsou přípravou pro obecný problém plánování, který bude řešen v kapitole 3.4 a dále. Nejprve je ale třeba zavést pojem generátor pohybu (trajektorie)

Definice 3.2. *Dán dynamický systém $\frac{d^n s}{dt^n} = u$, $\mathbf{x} = [s, \dot{s}, \ddot{s}, \dots, \frac{d^n s}{dt^n}]^T$, jehož jednotlivé složky stavu je možné reprezentovat jako polohu, rychlost, zrychlení, ... dílčích zobecněných souřadnic. Nechť je dále specifikován počáteční \mathbf{x}_0 a koncový \mathbf{x}_f (požadovaný) stav systému a množina pohybových omezení. Generátorem pohybu je pak takový systém, který ze vstupních dat nalezne trajektorie polohy, rychlosti a zrychlení, které převedou stav systému z počátečního do koncového stavu.*

Poznamenejme, že doposud není v literatuře žádná zmínka o analytickém algoritmu, který by pro třetí řád řetězce integrátorů našel řešení typu *bang-zero-bang* na libovolné množině počátečních a koncových podmínek a s omezeními na všechny stavy a vstup systému.

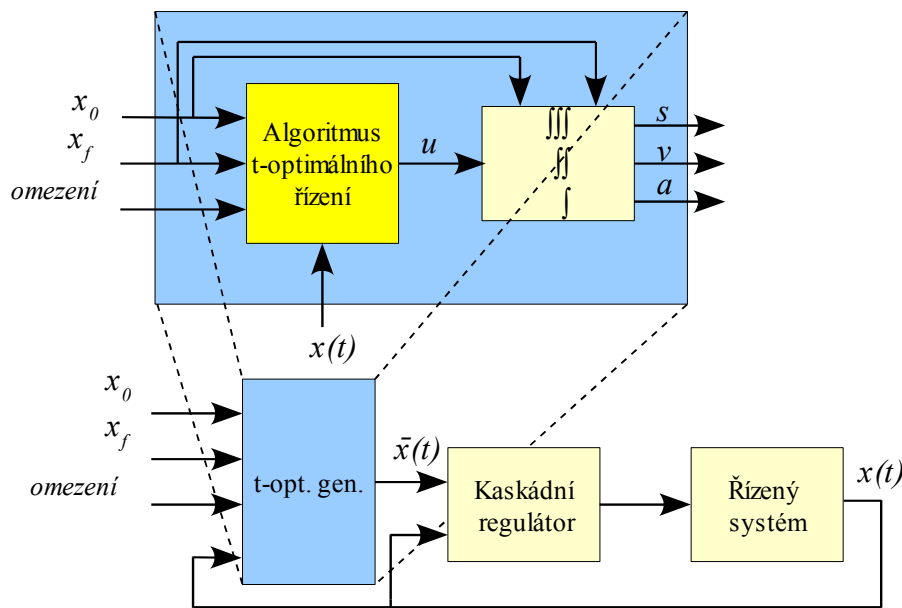
3.3.1 STANDARDNÍ PLÁNOVACÍ ALGORITMUS BAVS

Řetězec dvou integrátorů je systémem druhého řádu, kde výstup můžeme nazývat polohou $s(t)$, první derivaci rychlostí $v(t)$ a druhou derivaci zrychlením $a(t)$. U třetího řádu dostáváme ještě derivaci zrychlení. Tyto jednoduché systémy mohou generovat řídicí trajektorie například pro kaskádní regulaci a přitom respektovat stavová omezení, obrázek 3.5. Strmost řídicích křivek musí vyplývat, nebo být vhodně určena z původního řízeného systému tak, aby zpětnovazební regulátor byl schopen zajistit sledování těchto trajektorií. Časová optimalita je v případě zpětnovazebního řízení nebezpečnou vlastností, což bylo již dříve zmíněno. V případě generování řídicích trajektorií je ale hojně využívána právě proto, že řídicí zákon je dopředu znám, má jednoduchou strukturu a lze relativně snadno implementovat. Navíc mnoho reálných procesů, jako jsou například chemické procesy, je přímo modelováno jako řetězec integrátorů.

V této kapitole je provedeno odvození základní struktury t-optimálního plánovače *BAVS*⁵ využívajícího řetězec integrátorů 3.řádu a respektujícího konstantní stavová omezení. Tento typ generátoru je již dlouhou dobu velice dobře znám, a byl nasazen například firmou Škoda na válcovacích lisech. V té době došlo k velkému zlepšení plynulosti pohybu právě díky vlastnostem třetího řádu. Generátor *BAVS* umí plánovat pohyb pouze z klidu do klidu a bude později metodou Gröbnerovy báze zobecněn.

Pro porovnání s dalšími výsledky provedeme malé zesložnění původního algoritmu ve formě uvažování nesymetrického omezení na zrychlení, viz. dále.

⁵ Jméno generátoru je tvořeno písmeny, která se nejčastěji používají ve vztazích pro vyjádření polohy-*s*, rychlosti-*v*, zrychlení-*a* a derivace zrychlení-*b* (*Jerk*). Poznamenejme, že tento typ generátoru byl používán M. Schlegeltem již v roce 1978 ve válcovnách Škoda Plzeň.



Obrázek 3.5: Řetězec integrátorů jako generátor t-optimálních řídicích trajektorií pro systém s kaskádní regulací.

3.3.2 ZÁKLADNÍ STRUKTURA A VLASTNOSTI

Uvažujme řetězec tří integrátorů ve stavovém popisu

$$\begin{bmatrix} \dot{s} \\ \dot{v} \\ \dot{a} \end{bmatrix} = \underbrace{\begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}}_A \underbrace{\begin{bmatrix} s \\ v \\ a \end{bmatrix}}_x + \underbrace{\begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}}_B u, \quad (3.7)$$

kde jednotlivé složky stavu přirozeně pojmenujeme jako

$$\begin{aligned} s(t) &- \text{poloha} \\ v(t) &- \text{rychlost} \\ a(t) &- \text{zrychlení} \\ b(t) &- \text{Jerk (derivace zrychlení)} \end{aligned} \quad (3.8)$$

Dále uvažujme omezení na stav i vstup systému ve tvaru

$$\begin{aligned} -VM &\leq v(t) \leq VM, \quad VM > 0 \\ -DM &\leq a(t) \leq AM, \quad AM, DM > 0 \\ -BM &\leq u(t) \leq BM, \quad BM > 0 \end{aligned} \quad (3.9)$$

a počáteční $x_0 = [0, 0, 0]$ a koncový stav $x_f = [s_f, 0, 0]$.⁶ Velikost podprostoru vývoje polohy $s(t)$ je dána hraničními polohami počátečního a koncového stavu, protože systém během pohybu z klidu do klidu nemění znaménko rychlosti (nemění směr pohybu). Pro takovou úlohu lze dokázat, že minimální čas přesunu bude zajištěn bang-zero-bang řízením.

⁶ Protože plánujeme pohyb z klidu do klidu, můžeme bez ztráty obecnosti uvažovat počátek v nule a konec na dané poloze s_f .

Věta 3.1. Časově-optimální řízení systému (3.7) s omezeními (3.9) vede na **bang-zero-bang** řízení s maximálním počtem **sedmi** časových intervalů. Existují dvě přípustné sekvence řízení

$$u_+(t) = \begin{cases} BM, & t \in [0, \tau_1) \\ 0, & t \in [\tau_1, \tau_2) \\ -BM, & t \in [\tau_2, \tau_3) \\ 0, & t \in [\tau_3, \tau_4) \\ -BM, & t \in [\tau_4, \tau_5) \\ 0, & t \in [\tau_5, \tau_6) \\ BM, & t \in [\tau_6, \tau_7) \end{cases}, \quad u_-(t) = \begin{cases} -BM, & t \in [0, \tau_1) \\ 0, & t \in [\tau_1, \tau_2) \\ BM, & t \in [\tau_2, \tau_3) \\ 0, & t \in [\tau_3, \tau_4) \\ BM, & t \in [\tau_4, \tau_5) \\ 0, & t \in [\tau_5, \tau_6) \\ -BM, & t \in [\tau_6, \tau_7) \end{cases}, \quad (3.10)$$

kde

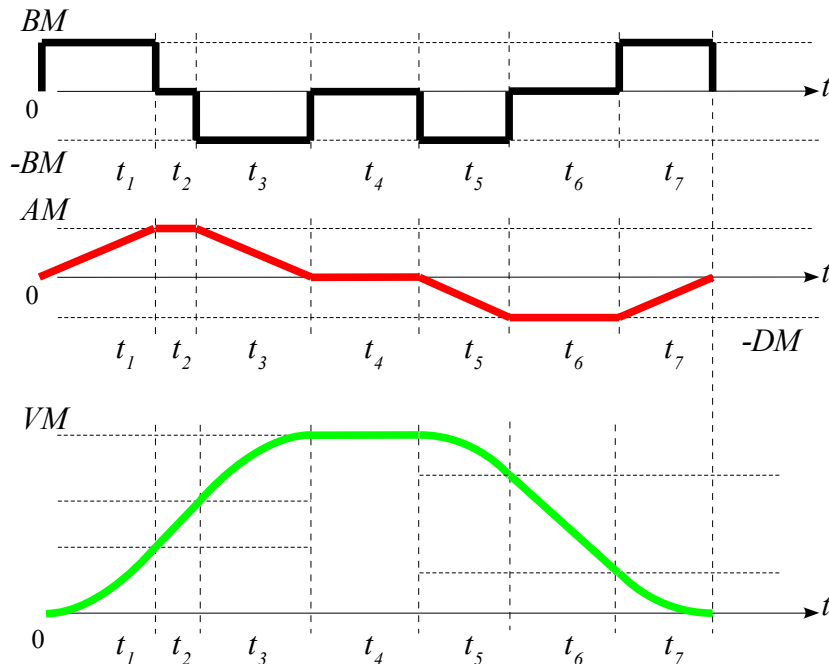
$$\tau_k = \sum_{i=1}^k t_i, \quad i=1..7. \quad (3.11)$$

Podle znaménka polohy s_f lze jednoznačně určit, která sekvence řízení převede systém z počátečního klidového stavu do koncového klidového stavu

$$s_f > 0 \Rightarrow u_+(t) \quad \vee \quad s_f < 0 \Rightarrow u_-(t)$$

Řízení za předpokladu pohybu z **klidu do klidu** existuje vždy. Navíc jednotlivé časové intervaly mohou zaniknout v případě, kdy systém nedosáhne daných stavových omezení.

Z věty vyplývá základní tvar trajektorie Jerku, zrychlení a rychlosti, viz obrázek 3.6.



Obrázek 3.6: Základní tvar trajektorie Jerku, zrychlení a rychlosti pro řídicí sekvenci $u_+(t)$ s vyznačením příslušných časových intervalů.

3.3.3 TVARY TRAJEKTORIE

Ze symetrie řízení a pohybu *z klidu do klidu* vyplývá, že některé časové intervaly budou totožné. Je zřejmé, že musí platit $t_3=t_1, t_7=t_5$, aby integrace zrychlení byla nulová na maximální rychlosti a v koncovém stavu. Ze 7 časových intervalů tak dostáváme pouze 5 různých

$$t_1, t_2, t_4, t_5, t_6 \quad (3.12)$$

pro které musí platit (*některé mohou být nulové, v případě, že systém nedosáhne omezení*)

$$\begin{aligned} t_1 &> 0 \\ t_2 &\geq 0 \text{ (AM omezení)} \\ t_4 &\geq 0 \text{ (VM omezení)} \\ t_5 &> 0 \\ t_6 &\geq 0 \text{ (DM omezení)} \end{aligned} \quad (3.13)$$

Vidíme, že 3 časy mohou zaniknout a dostáváme tak 2^3 základních tvarů trajektorie, pro které nyní odvodíme hodnoty příslušných časových intervalů.⁷

Protože známe přesný tvar řídicí strategie, je možné přímou integrací dostat hodnoty časových intervalů pro každou trajektorii. Pro názornost ale využijeme následujícího obrázku, kde jsou vyznačeny klíčové oblasti a velikost integrálu těchto oblastí. Ty následně použijeme pro další odvozování. Pokud trajektorie nedosáhne omezení, potom bude dosažené maximum označeno při odvozování jako *am*, *dm* nebo *vm*. V případě, že tvar trajektorie omezení nerespektuje, je třeba po výpočtu jednotlivých časových intervalů tato omezení otestovat následujícím způsobem

- test omezení VM

$$VM - BM (t_1 t_2 + t_1^2) = \begin{cases} \geq 0: & \text{'řešení splňuje omezení VM'} \\ < 0: & \text{'řešení neexistuje'} \end{cases} \quad (3.14)$$

- test omezení AM

$$AM - BM t_1 = \begin{cases} \geq 0: & \text{'řešení splňuje omezení AM'} \\ < 0: & \text{'řešení neexistuje'} \end{cases} \quad (3.15)$$

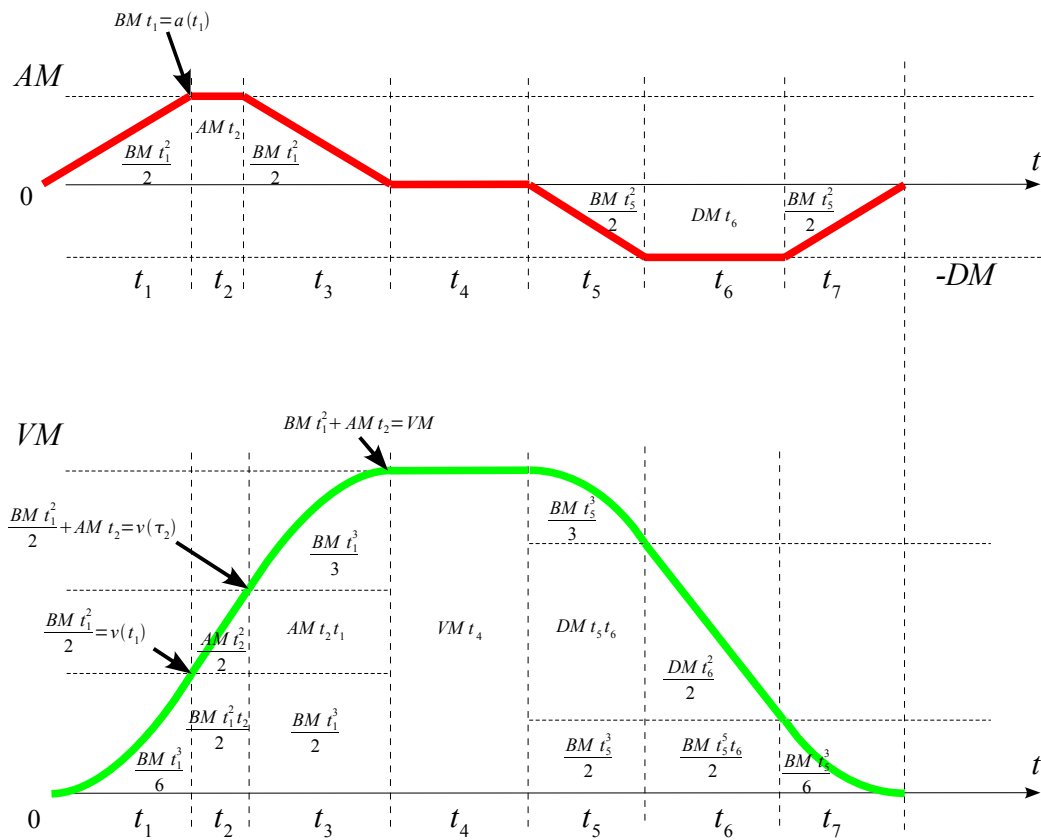
- test omezení DM

$$DM - BM t_5 = \begin{cases} \geq 0: & \text{'řešení splňuje omezení DM'} \\ < 0: & \text{'řešení neexistuje'} \end{cases} \quad (3.16)$$

Testy omezení vyplývají z přímé integrace vývoje stavu systému podél zvolené strategie řízení, viz obrázek 3.7. Hledaným řešením úlohy je pak takové řízení $u^*(t)$, jehož trajektorie existuje, splňuje podmínky (3.14), (3.15) a (3.16) a minimalizuje kritérium kvality (3.11).

Dále je proveden detailní rozbor všech osmi typů trajektorií, včetně analytického výpočtu délek časových intervalů $t_{1..7}$.

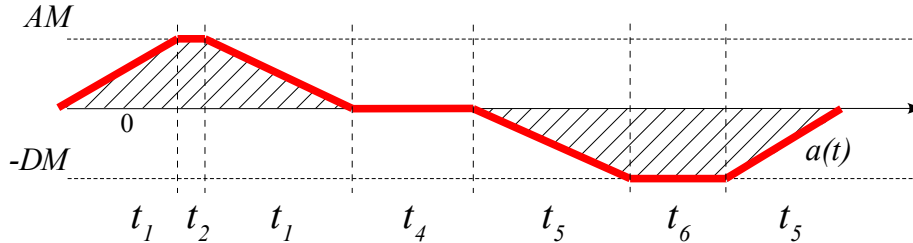
⁷ Ostatní časy zaniknout nemohou, jinak by vstupem byla nulová funkce, což v případě pohybu *z klidu do klidu* může nastat pouze pokud $x_0 = x_f$ a tuto variantu nepředpokládáme.



Obrázek 3.7: Základní tvar trajektorie zrychlení a rychlosti s vyznačením klíčových hodnot v jednotlivých časových intervalech a s vyznačením velikostí ploch pod daným úsekem křivky.

TYP1

Tvar trajektorie odpovídá situaci, kdy systém dosáhne všech omezení. Velikost vyšrafovaných oblastí v obrázku 3.8 musí být shodná.



Obrázek 3.8: Tvar a a základní vlastnosti trajektorie zrychlení pro TYP1.

Pokud je řešením soustavy 7 kladných reálných časových intervalů, potom řešení zároveň vyhovuje omezením a je hledaným řešením úlohy. Přímou integrací (viz obrázek 3.7) lze snadno nalézt hodnoty časových intervalů

Bude platit

$$BM t_1 = AM \Rightarrow t_1 = \frac{AM}{BM}$$

$$-BM t_5 = -DM \Rightarrow t_5 = \frac{DM}{BM}$$

$$BM t_1^2 + AM t_2 = VM \rightarrow BM \frac{AM^2}{BM^2} + AM t_2 = VM \Rightarrow t_2 = \frac{VM}{AM} - \frac{AM}{BM}$$

$$BM t_5^2 + DM t_6 = VM \rightarrow BM \frac{DM^2}{BM^2} + DM t_6 = VM \Rightarrow t_6 = \frac{VM}{DM} - \frac{DM}{BM}$$

Pro určení času t_4 je třeba integrovat celou plochu pod křivkou rychlosti. Sečtením všech vyznačených oblastí, dosazením časů t_1, t_5, t_6, t_7 a pokrácením dostaneme výsledný vztah

$$s_f = \frac{AM VM}{2 BM} + \frac{VM^2}{2 AM} + VM t_4 + \frac{DM VM}{2 BM} + \frac{VM^2}{2 DM} \Rightarrow t_4 = \frac{s_f}{VM} - \frac{(AM + DM)(AM DM + BM VM)}{2 AM BM DM}$$

Dostáváme výsledný soupis jednotlivých vzorců pro délky intervalů

$$t_1 = AM / BM$$

$$t_2 = VM / AM - AM / BM$$

$$t_3 = t_1$$

$$t_4 = \frac{s_f}{VM} - \frac{(AM + DM)(AM DM + BM VM)}{2 AM BM DM} \quad (3.17)$$

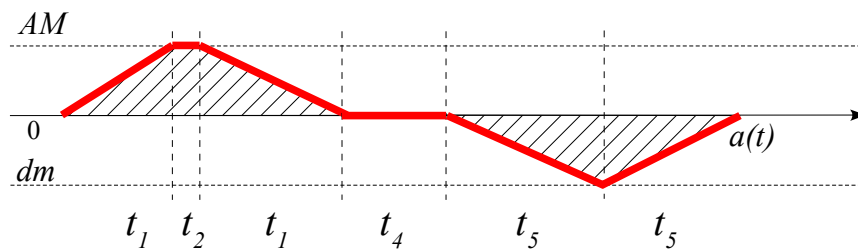
$$t_5 = DM / BM$$

$$t_6 = VM / DM - DM / BM$$

$$t_7 = t_5$$

TYP2

Tvar trajektorie nerespektuje omezení DM a odpovídá situaci, kdy systém omezení DM nedosáhne. Velikost vyšrafovaných oblastí v obrázku 3.9 musí být shodná. Pokud je řešením soustavy 7 nezáporných reálných časových intervalů, potom je nutné otestovat omezení DM podle (3.16).



Obrázek 3.9: Tvar a základní vlastnosti trajektorie zrychlení pro TYP2.

Modifikací (3.17) bude platit

$$t_6=0 \Rightarrow \frac{VM}{dm} - \frac{dm}{BM} = 0 \Rightarrow dm = \sqrt{VM BM}$$

$$t_4 = \frac{s_f}{VM} - \frac{(AM + dm)(AM dm + BM VM)}{2 AM BM dm} = \frac{-AM}{2 BM} + \frac{s_f}{VM} - \frac{VM}{2 AM} - \sqrt{\frac{VM}{BM}}$$

Dostáváme výsledný soupis jednotlivých vzorců pro délky intervalů

$$t_1 = \frac{AM}{BM}$$

$$t_2 = \frac{VM}{AM} - \frac{AM}{BM}$$

$$t_3 = t_1$$

$$t_4 = \frac{-AM}{2 BM} + \frac{s_f}{VM} - \frac{VM}{2 AM} - \sqrt{\frac{VM}{BM}}$$

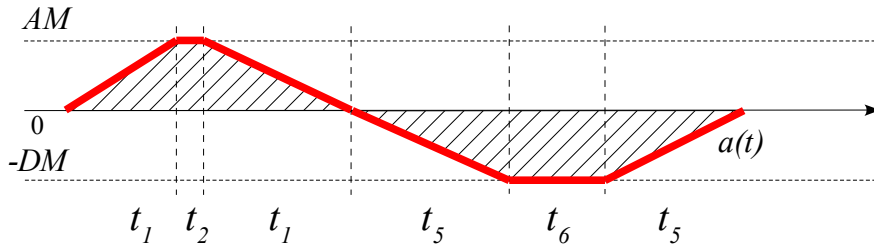
$$t_5 = \sqrt{\frac{VM}{BM}}$$

$$t_6 = 0$$

$$t_7 = t_5$$

TYP3

Tvar trajektorie nerespektuje omezení VM a odpovídá situaci, kdy systém omezení VM nedosáhne. Velikost vyšrafovaných oblastí v obrázku 3.10 musí být shodná. Pokud je řešením soustavy 7 nezáporných reálných časových intervalů, potom je nutné otestovat omezení VM podle (3.14).



Obrázek 3.10: Tvar a základní vlastnosti trajektorie zrychlení pro TYP3.

Modifikací (3.17) bude platit

$$t_4=0 \Rightarrow \frac{s_f}{vm} - \frac{(AM + DM)(AM - DM + BM vm)}{2 AM BM DM} = 0$$

$$\Rightarrow vm = -\frac{AM - DM}{2 BM} + \sqrt{\left(\frac{AM - DM}{2 BM}\right)^2 + \frac{2 AM DM s_f}{AM + DM}}$$

(Druhé řešení kvadratické rovnice není kladné a je tedy nepřipustné).

$$t_2 = \frac{vm}{AM} - \frac{AM}{BM} = -\frac{2 AM + DM}{2 BM} + \sqrt{\left(\frac{AM - DM}{2 BM}\right)^2 + \frac{2 AM DM s_f}{AM + DM}}$$

$$t_6 = \frac{vm}{DM} - \frac{DM}{BM} = -\frac{AM + 2 DM}{2 BM} + \sqrt{\left(\frac{AM - DM}{2 BM}\right)^2 + \frac{2 AM DM s_f}{AM + DM}}$$

Dostáváme výsledný soupis jednotlivých vzorců pro délky intervalů

$$t_1 = \frac{AM}{BM}$$

$$t_2 = -\frac{2 AM + DM}{2 BM} + \sqrt{\left(\frac{AM - DM}{2 BM}\right)^2 + \frac{2 AM DM s_f}{AM + DM}}$$

$$t_3 = t_1$$

$$t_4 = 0$$

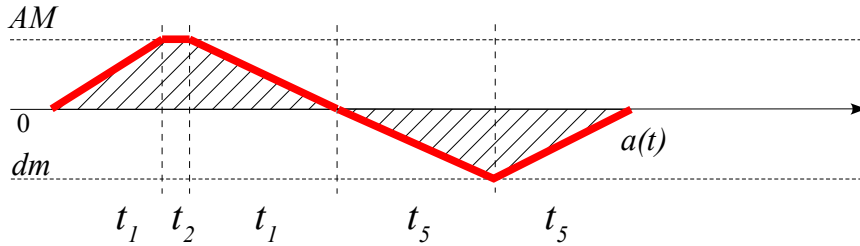
$$t_5 = \frac{DM}{BM}$$

$$t_6 = -\frac{AM + 2 DM}{2 BM} + \sqrt{\left(\frac{AM - DM}{2 BM}\right)^2 + \frac{2 AM DM s_f}{AM + DM}}$$

$$t_7 = t_5$$

TYP4

Tvar trajektorie nerespektuje omezení VM a DM a odpovídá situaci, kdy systém těchto omezení nedosáhne. Velikost vyšrafovaných oblastí v obrázku 3.11 musí být shodná. Pokud je řešením soustavy 7 nezáporných reálných časových intervalů, potom je nutné otestovat omezení podle (3.14) a (3.16).



Obrázek 3.11: Tvar a základní vlastnosti trajektorie zrychlení pro TYP4.

Modifikací (3.17) a porovnáním s typem 7 bude podobně platit

$$t_4=0 \Rightarrow \frac{s_f}{vm} - \frac{(AM + dm)(AM dm + BM vm)}{2 AM BM dm} = 0$$

$$\Rightarrow vm = -\frac{AM dm}{2 BM} + \sqrt{\left(\frac{AM dm}{2 BM}\right)^2 + \frac{2 AM dm s_f}{AM + dm}}$$

(Druhé řešení kvadratické rovnice není kladné a je tedy nepřipustné).

$$t_6=0 \Rightarrow \frac{vm}{dm} - \frac{dm}{BM} = 0 \Rightarrow dm^2 = vm BM$$

Dosazením za vm a řešením rovnice

$$dm^2 + \frac{AM dm}{2} - \sqrt{\left(\frac{AM dm}{4}\right)^2 + \frac{2 AM dm BM^2 s_f}{AM + dm}} = 0$$

dostaneme dvě řešení, která mohou být kladná

$$dm_1 = -AM/2 + 1/2 \sqrt{(AM^2 - 4 BM \sqrt{(2 AM s_f)})}$$

$$dm_2 = -AM/2 + 1/2 \sqrt{(AM^2 + 4 BM \sqrt{(2 AM s_f)})}$$

Dostáváme výsledný soupis jednotlivých vzorců pro délky intervalů

$$t_1 = \frac{AM}{BM}$$

$$t_2 = \frac{dm_{1,2}^2}{AM BM} - \frac{AM}{BM}$$

$$t_3 = t_1$$

$$t_4 = 0$$

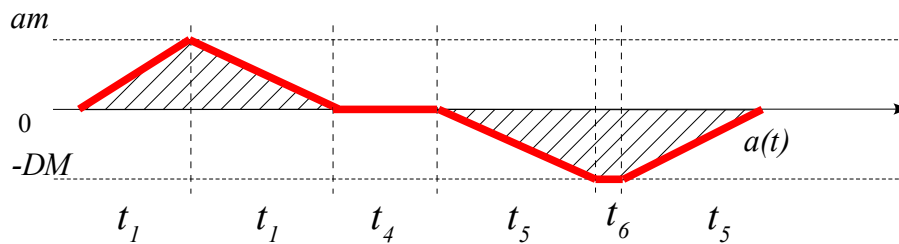
$$t_5 = \frac{dm_{1,2}}{BM}$$

$$t_6 = 0$$

$$t_7 = t_5$$

TYP5

Tvar trajektorie nerespektuje omezení AM a odpovídá situaci, kdy systém omezení AM nedosáhne. Z pohledu symetrie se jedná o ekvivalentní případ s typem 2. Velikost vyšrafovaných oblastí v obrázku 3.12 musí být shodná. Pokud je řešením soustavy 7 nezáporných reálných časových intervalů, potom je nutné otestovat omezení AM podle (3.15).



Obrázek 3.12: Tvar a základní vlastnosti trajektorie zrychlení pro TYP5.

Modifikací (3.17) a porovnáním s typem 2 bude platit

$$t_2=0 \Rightarrow \frac{VM}{am} - \frac{am}{BM} = 0 \Rightarrow am = \sqrt{VM BM}$$

$$t_4 = \frac{s_f}{VM} - \frac{(am + DM)(DM am + BM VM)}{2 DM BM am} = \frac{-DM}{2 BM} + \frac{s_f}{VM} - \frac{VM}{2 DM} - \sqrt{\frac{VM}{BM}}$$

Dostáváme výsledný soupis jednotlivých vzorců pro délky intervalů

$$t_1 = \sqrt{\frac{VM}{BM}}$$

$$t_2 = 0$$

$$t_3 = t_1$$

$$t_4 = \frac{-DM}{2 BM} + \frac{s_f}{VM} - \frac{VM}{2 DM} - \sqrt{\frac{VM}{BM}}$$

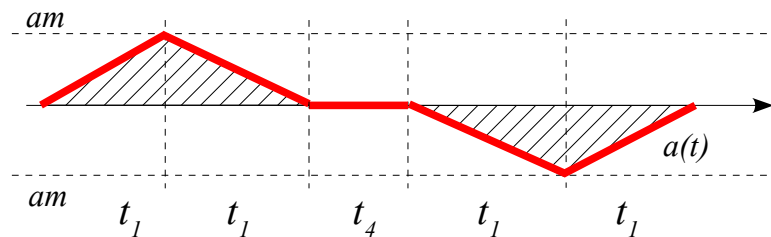
$$t_5 = \frac{DM}{BM}$$

$$t_6 = \frac{VM}{DM} - \frac{DM}{BM}$$

$$t_7 = t_5$$

TYP6

Tvar trajektorie nerespektuje AM , DM omezení a odpovídá situaci, kdy systém dosáhne omezení VM . Velikost vyšrafovaných oblastí v obrázku 3.13 musí být shodná. Pokud je řešením soustavy 7 nezáporných reálných časových intervalů, potom je nutné otestovat omezení AM, DM podle (3.15) a (3.16).



Obrázek 3.13: Tvar a a základní vlastnosti trajektorie zrychlení pro TYP6.

Modifikací (3.17) bude platit

$$t_2=0 \Rightarrow \frac{VM}{am} - \frac{am}{BM} = 0 \Rightarrow am^2 = VM \cdot BM$$

$$t_1 = \frac{\sqrt{VM \cdot BM}}{BM} = \sqrt{\frac{VM}{BM}}$$

$$t_4 = \frac{s_f}{VM} - \frac{am}{BM} - \frac{VM}{am} = 0 \Rightarrow t_4 = \frac{s_f}{VM} - 2 \sqrt{\frac{VM}{BM}}$$

Dostáváme výsledný soupis jednotlivých vzorců pro délky intervalů

$$t_1 = \sqrt{\frac{VM}{BM}}$$

$$t_2 = 0$$

$$t_3 = t_1$$

$$t_4 = \frac{s_f}{VM} - 2 \sqrt{\frac{VM}{BM}}$$

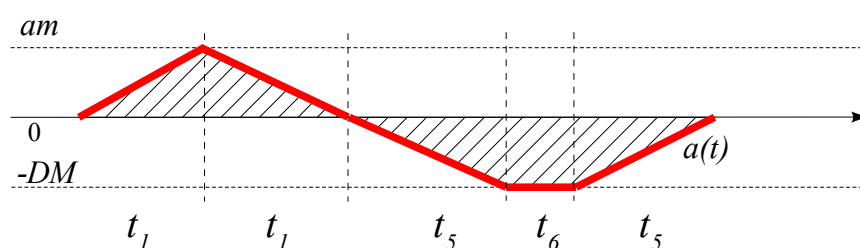
$$t_5 = t_1$$

$$t_6 = 0$$

$$t_7 = t_1$$

TYP7

Tvar trajektorie nerespektuje omezení VM a AM a odpovídá situaci, kdy systém těchto omezení nedosáhne. Z pohledu symetrie se jedná o ekvivalentní případ s typem 4. Velikost vyšrafovaných oblastí v obrázku 3.14 musí být shodná. Pokud je řešením soustavy 7 nezáporných reálných časových intervalů, potom je nutné otestovat omezení podle (3.14) a (3.15).



Obrázek 3.14: Tvar a základní vlastnosti trajektorie zrychlení pro TYP7.

Modifikací (3.17) a porovnáním s typem 4 bude podobně platit

$$t_4=0 \Rightarrow \frac{s_f}{vm} - \frac{(am+DM)(amDM+BMvm)}{2amBM DM} = 0$$

$$\Rightarrow vm = -\frac{amDM}{2BM} + \sqrt{\left(\frac{amDM}{2BM}\right)^2 + \frac{2amDMs_f}{am+DM}}$$

(Druhé řešení kvadratické rovnice není kladné a je tedy nepřipustné).

$$t_2=0 \Rightarrow \frac{vm}{am} - \frac{am}{BM} = 0 \Rightarrow am^2 = vmBM$$

Dosazením za vm a řešením rovnice

$$am^2 + \frac{DMam}{2} - \sqrt{\left(\frac{DMam}{4}\right) + \frac{2amDM BM^2 s_f}{DM+am}} = 0$$

dostaneme dvě řešení, z nichž kladné může být pouze druhé

$$am_1 = -DM/2 + 1/2 \sqrt{(DM^2 - 4BM \sqrt{(2DMs_f)})}$$

$$am_2 = -DM/2 + 1/2 \sqrt{(DM^2 + 4BM \sqrt{(2DMs_f)})}$$

Dostáváme výsledný soupis jednotlivých vzorců pro délky intervalů

$$t_1 = \frac{am_{1,2}}{BM}$$

$$t_2 = 0$$

$$t_3 = t_1$$

$$t_4 = 0$$

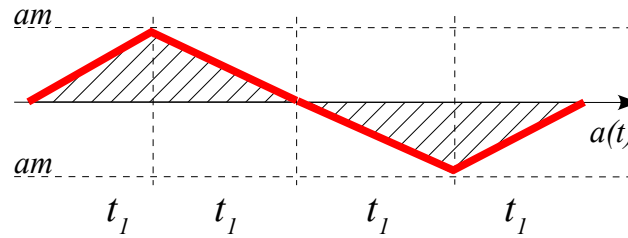
$$t_5 = \frac{DM}{BM}$$

$$t_6 = \frac{am_{1,2}^2}{DM BM} - \frac{DM}{BM}$$

$$t_7 = t_5$$

TYP8

Tvar trajektorie nerespektuje žádné omezení a odpovídá situaci, kdy systém žádného omezení nedosáhne. Velikost vyšrafovaných oblastí v obrázku 3.15 musí být shodná. Pokud je řešením soustavy 7 nezáporných reálných časových intervalů, potom je nutné otestovat omezení AM, DM a VM podle (3.14), (3.15) a (3.16). Poznamenejme, že tato soustava odpovídá obecnému řešení pomocí principu maxima pro systém 3 integrátorů bez stavových omezení, tj. *bang-bang* řízení.



Obrázek 3.15: Tvar a základní vlastnosti trajektorie zrychlení pro TYP8.

Modifikací (3.17) bude platit

$$t_2=0 \Rightarrow \frac{vm}{am} - \frac{am}{BM} = 0 \Rightarrow vm = \frac{am^2}{BM}$$

$$t_4=0 \Rightarrow \frac{s_f}{vm} - \frac{am}{BM} - \frac{vm}{am} = 0 \Rightarrow \frac{s_f BM}{am^2} = \frac{2am}{BM} \Rightarrow am = \sqrt[3]{\frac{s_f BM^2}{2}}$$

$$t_1 = \frac{am}{BM} = \sqrt[3]{\frac{s_f}{2 BM}}$$

Dostáváme výsledný soupis jednotlivých vzorců pro délky intervalů

$$t_1 = \sqrt[3]{\frac{s_f}{2 BM}}$$

$$t_2 = 0$$

$$t_3 = t_1$$

$$t_4 = 0$$

$$t_5 = t_1$$

$$t_6 = 0$$

$$t_7 = t_1$$

3.4 GRÖBNEROVA BÁZE A T-OPTIMÁLNÍ ŘÍZENÍ

Hledání přepínacích variet v t-optimálním řízení je široce studovanou oblastí. Již v 60. letech byla ukázána jejich rozmanitost vzhledem k poloze pólů systému 2. řádu, [25]. Přepínací variety jsou ve velké míře definovány nebo aproximovány polynomiálními funkcemi, a jsou tedy vhodnými adepty na využití Gröbnerovy báze. Uli Walther a Tryphon T. Georgiou představili aplikaci Gröbnerovy báze na řetězec tří integrátorů. Převodli problém z hledání přepínacích variet na kombinatorický problém [106], přičemž neuvažovali stavová omezení ani libovolný koncový stav. Nicméně jejich práce byla hlavním impulzem pro budoucí výsledky této práce. V následujícím textu se omezíme pouze na lineární systémy, a to na řetězec integrátorů s omezeními. S využitím *GB* odvodíme obecný postup pro nalezení přepínacích variet řetězce dvou a tří integrátorů s konstantními omezeními na vstup a všechny stavy systému.

3.4.1 SYSTÉM DRUHÉHO ŘÁDU (2 INTEGRÁTORY S OMEZENÍMI)

Uvažujme stavový popis řetězce dvou integrátorů

$$\begin{bmatrix} \dot{s} \\ \dot{v} \end{bmatrix} = \underbrace{\begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}}_A \underbrace{\begin{bmatrix} s \\ v \end{bmatrix}}_x + \underbrace{\begin{bmatrix} 0 \\ 1 \end{bmatrix}}_B u \quad (3.18)$$

s konstantními omezeními na jejich stavy

$$\begin{aligned} s_{Min} &\leq s(t) \leq s_{Max} \\ -VM &\leq v(t) \leq VM \end{aligned} \quad (3.19)$$

a na vstup systému $u(t) \equiv a(t)$ ve tvaru

$$\begin{aligned} -DM &\leq a(t) \leq AM, \text{ jestliže } v(t) \geq 0 \\ -AM &\leq a(t) \leq DM, \text{ jestliže } v(t) < 0 \end{aligned} \quad (3.20)$$

Předpokládejme, že počáteční stav \mathbf{x}_0 vyhovuje omezením, respektive, že je možné ho přípustným řízením opustit bez porušení omezení. Dále předpokládejme, že koncový stav \mathbf{x}_f je z pohledu řízení a omezení dosažitelný.

Problém 3.2. Nalezněte řízení u , které převede systém (3.18) z počátečního stavu \mathbf{x}_0 do koncového stavu \mathbf{x}_f za minimální čas, při respektování omezení (3.19) a (3.20).

Z již zmíněného článku [26] víme, že pro lineární systém 2. řádu s konstantními omezeními na vstup a jeho strmost bude vstupem strategie řízení typu *bang-zero-bang*. Ta zajistí, že v každém okamžiku je alespoň jedna ze složek stavu nebo vstup systému na maximální nebo minimální hodnotě, a tedy řízení splňuje podmínku časové optimality. Počet časových intervalů t_i je dán podle tvaru omezení následující větou.

Věta 3.2. Časově-optimální řízení pro systém (3.18) s omezeními (3.19) a (3.20) vede na tzv. **bang-zero-bang** řízení s maximálním počtem pěti časových intervalů. Existují dvě přípustné sekvence řízení

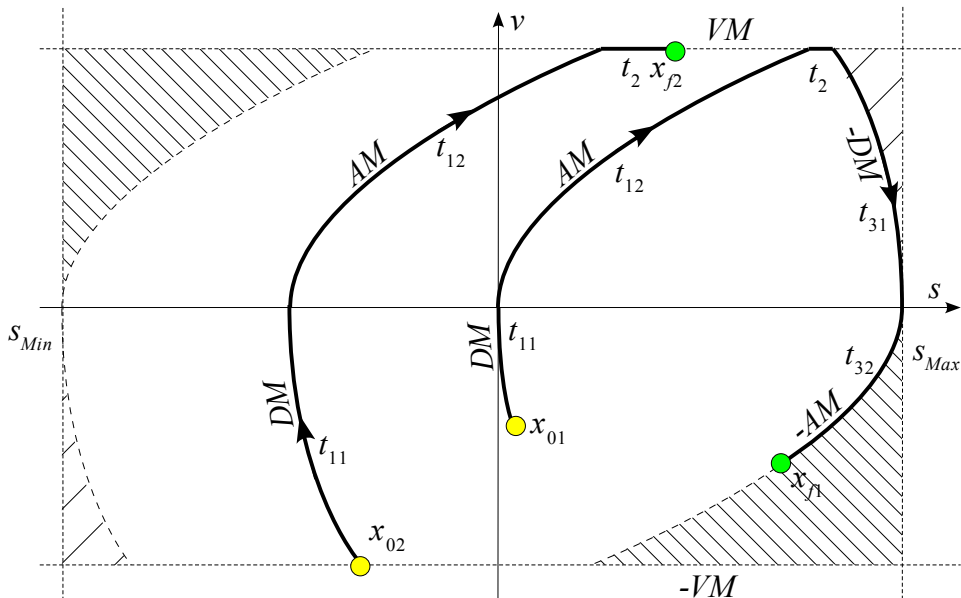
$$u_+(t) = \begin{cases} DM, & t \in [0, t_{11}) \\ AM, & t \in [t_{11}, t_{11} + t_{12}) \\ 0, & t \in [t_{11} + t_{12}, t_{11} + t_{12} + t_2) \\ -DM, & t \in [t_{11} + t_{12} + t_2, t_{11} + t_{12} + t_2 + t_{31}) \\ -AM, & t \in [t_{11} + t_{12} + t_2 + t_{31}, t_{11} + t_{12} + t_2 + t_{31} + t_{32}) \end{cases} \quad (3.21)$$

$$u_+(t) = \begin{cases} -DM, & t \in [0, t_{11}) \\ -AM, & t \in [t_{11}, t_{11} + t_{12}) \\ 0, & t \in [t_{11} + t_{12}, t_{11} + t_{12} + t_2) \\ DM, & t \in [t_{11} + t_{12} + t_2, t_{11} + t_{12} + t_2 + t_{31}) \\ AM, & t \in [t_{11} + t_{12} + t_2 + t_{31}, t_{11} + t_{12} + t_2 + t_{31} + t_{32}) \end{cases} \quad (3.22)$$

kde význam $t_{11}, t_{12}, t_2, t_{31}, t_{32}$ je patrný z obrázku 3.16. Ze znalosti počátečního a koncového stavu lze jednoznačně určit, která řídicí sekvence převede systém do koncového stavu, nebo zda takové řízení vzhledem k daným omezením neexistuje. Navíc jednotlivé časové intervaly mohou zaniknout v případě, kdy systém nedosáhne daných stavových omezení.

Konstrukci důkazu této věty je možné nalézt v článku [26], kde je proveden obecný důkaz pro systém druhého řádu s obecným tvarem všech omezení.

Protože uvažujeme nesymetrické mezní hodnoty řízení (3.20), museli jsme přidat, oproti standardní úloze, dva intervaly, které budou aktivní v případě, kdy systém bude přecházet ze stavu brzdění do stavu zrychlení a naopak, viz obrázek 3.16. Ty ovšem nepřinesou do úlohy žádné zesložnění, protože jsou přímo vyčíslitelné ze vstupních dat, viz dále. Řízení při přechodu $v(T)=0$ nebude měnit znaménko, pouze se bude měnit amplituda ($DM \rightarrow AM$ nebo $-DM \rightarrow -AM$). Toto zobecnění neporušuje podmínku časové optimality a splňuje vlastnosti *bang-zero-bang* řízení.



Obrázek 3.16: Příklady trajektorií v omezeném stavovém prostoru řetězce 2-int.

Ze znalosti počátečního stavu $\mathbf{x}_0 = [s_0, v_0]^T$, koncového stavu $\mathbf{x}_f = [s_f, v_f]^T$ a sekvence řízení $u_{\pm}(t)$ již můžeme vyčíslit vztah (3.5) a získat algebraické rovnice pro neznámé $t_{11}, t_{12}, t_2, t_{31}, t_{32}$. Bez ztráty obecnosti provedeme vyčíslení pouze pro řídicí sekvenci (3.21). Pro (3.22) je úloha ekvivalentní. To vyplývá ze symetrie stavového prostoru a omezení vzhledem k počátku, viz dále.

Z (3.5), (3.18) a (3.21) tedy dostáváme algebraické rovnice stavového přechodu

$$\begin{aligned} eq_1: \quad v_f &= v_0 + AM t_{12} - AM t_{32} + DM t_{11} - DM t_{31} \\ eq_2: \quad s_f &= s_0 + v_0(t_{11} + t_{12} + t_2 + t_{31} + t_{32}) + (t_{11}t_{31} + \frac{t_{11}^2}{2} - t_{31}t_{32} + t_{11}t_2 - \frac{t_{31}^2}{2} + t_{11}t_{32} + t_{11}t_{12}) DM \\ &\quad + (\frac{t_{12}^2}{2} - \frac{t_{32}^2}{2} + t_{12}t_{31} + t_{12}t_2 + t_{12}t_{32}) AM \end{aligned} \quad (3.23)$$

Z omezení (3.19) a (3.20) vyplývají další rovnice, které nám úlohu rozdělí na případ, kdy systém dosáhne maximální rychlosti

$$\begin{aligned} eq_{3b1}: \quad VM - v_0 - DM t_{11} - AM t_{12} &= 0 \\ eq_{3b2}: \quad v_f - VM + DM t_{31} + AM t_{32} &= 0 \end{aligned} \quad (3.24)$$

a na případ, kdy k tomu nedojde

$$eq_{3a}: \quad t_2 = 0. \quad (3.25)$$

Dostáváme tedy dvě soustavy rovnic pro sekvenci řízení $u_{\pm}(t)$

$$\begin{aligned} F_1 &= \{eq_1, eq_2, eq_{3b1}, eq_{3b2}\} \\ F_2 &= \{eq_1, eq_2, eq_{3a}\}, \end{aligned} \quad (3.26)$$

kde soustava F_1 respektuje omezení na rychlost, soustava F_2 nikoli.

Spočteme-li Gröbnerovu bázi pro soustavu F_1 , dostaneme

$$\begin{aligned} G_1 &= \{v_f - VM + DM t_{31} + AM t_{32}, \\ &\quad -2(v_0 DM^2 t_{11} + AM^2 v_f t_{32}) + AM DM (AM t_{32}^2 + DM t_{11}^2) + VM^2 (AM + DM) \\ &\quad 2 AM DM (s_0 - s_f + v_0 t_{11} + v_f t_{32} + t_2 VM) - AM v_f^2 - DM v_0^2 - t_{32}^2 AM^3 - DM^3 t_{11}^2, \\ &\quad -VM + v_0 + DM t_{11} + AM t_{12}\}, \end{aligned} \quad (3.27)$$

kde t_{11} a t_{32} jsou podle polohy \mathbf{x}_0 a \mathbf{x}_f dány následujícími podmínkami

$$t_{11} = \begin{cases} 0, & \text{jestliže } v_0 \geq 0 \\ -\frac{v_0}{DM}, & \text{jestliže } v_0 < 0 \end{cases}, \quad t_{32} = \begin{cases} 0, & \text{jestliže } v_f \geq 0 \\ -\frac{v_f}{AM}, & \text{jestliže } v_f < 0 \end{cases} \quad (3.28)$$

Ty vyplývají z geometrie stavového prostoru a sekvence řízení (3.21). Platí například, že pokud je počáteční rychlost v_0 kladná, potom není omezení DM aktivní a první časový interval je nulový. V opačném případě lze spočítat čas přechodu systému z počáteční do nulové rychlosti integrací druhé pohybové rovnice. Dostaneme rovnici

$$0 = v_0 + DM t_{11}, \quad v_0 < 0, \quad (3.29)$$

z které vyplývá první podmínka (3.28). Podobně lze interpretovat druhou podmínku (3.28). Dosazením těchto podmínek do první a třetí rovnice (3.27) dostáváme

$$t_{12} = \begin{cases} \frac{VM - v_0}{AM}, & \text{jestliže } t_{11} = 0 \\ \frac{VM}{AM}, & \text{jestliže } t_{11} = -\frac{v_0}{DM} \end{cases}, \quad t_{31} = \begin{cases} \frac{VM - v_f}{DM}, & \text{jestliže } t_{32} = 0 \\ \frac{VM}{DM}, & \text{jestliže } t_{32} = -\frac{v_f}{AM} \end{cases}$$

a poslední neznámá t_2 je dána řešením druhé rovnice (3.27)

$$t_2 = \frac{1}{2AMDMVM} (DM^3 t_{11}^2 + (2v_0 t_{11} - t_{11}^2 AM) DM^2 + (-t_{32}^2 AM^2 + (-2s_0 - 2v_0 t_{11} + 2s_f - 2v_f t_{32}) AM + v_0^2 - VM^2) DM + t_{32}^2 AM^3 + 2AM^2 v_f t_{32} + (v_f^2 - VM^2) AM)$$

Soustava F_1 má tedy žádné nebo jedno kladné reálné řešení. Protože tato soustava respektuje omezení na rychlost, je jasné, že pokud má kladné reálné řešení, potom je to hledané t-optimální řešení. To vyplývá z možného pohybu systému po úsecích parabol ve stavovém prostoru. Není tedy nutné hledat řešení soustavy F_2 . Pokud ho nemá, musíme prohledat druhou soustavu. Pro tu musí být splněna podmínka, že maximální dosažená rychlost bude pod hodnotou omezení VM , neboli

$$VM - v_0 - DM t_{11} - AM t_{12} = \begin{cases} \geq 0: & \text{'řešení splňuje omezení VM'} \\ < 0: & \text{'řešení neexistuje'} \end{cases} \quad (3.30)$$

Pro výpočet délek časových intervalů soustavy F_2 postupujeme podobně:

$$G_2 = \{ AM DM (t_{11}^2 + t_{31}^2 + 2t_{31} t_{32}) + 2AM (s_0 - s_f + AM t_{32}^2 + t_{31} v_f + AM t_{31} t_{32} + v_0 t_{11} + 2t_{32} v_f) + DM (2t_{31} v_f - 2v_0 t_{11} + t_{31}^2 - t_{11}^2) - v_0^2 + v_f^2, t_2, v_0 - v_f + AM (t_{12} - t_{32}) + DM (t_{11} - t_{31}) \} \quad (3.31)$$

kde t_{11} a t_{32} jsou podle polohy x_0 a x_f dány podmínkami (3.28).

Substitucí (3.28) do (3.31) dostaneme soustavu bázových rovnic o třech neznámých. První rovnice (3.31) obsahuje pouze jednu neznámou t_{31} v kvadratickém tvaru

$$(DM^2 + AM DM) t_{31}^2 + (2AM DM t_{32} + 2AM v_f + 2DM v_f + 2AM^2 t_{32}) t_{31} + 2AM^2 t_{32}^2 + 2AM s_0 - 2AM s_f + AM DM t_{11}^2 + 2AM v_0 t_{11} - DM^2 t_{11}^2 - v_0^2 + 4AM t_{32} v_f - 2v_0 DM t_{11} + v_f^2 = 0$$

druhá také, $t_2 = 0$ a z třetí rovnice vyjádříme poslední neznámou ve tvaru

$$t_{12} = \frac{v_f - v_0 + AM t_{32} + DM (t_{31} - t_{11})}{AM}. \quad (3.32)$$

Celkem tedy může mít soustava F_2 žádné až dvě reálná řešení. Z toho kladné reálné řešení bude pouze jedno. To je dáno pohybem systému po parabolách, které jsou symetrické podle osy s.

SYMETRIE ÚLOHY A JEJÍ VYUŽITÍ PRO VÝPOČET

Protože problém je symetrický podle počátku, lze pomocí následující transformace převést řešení v množině řízení $u_+(t)$ na ekvivalentní problém v množině řízení $u_-(t)$. Pokud vzájemná poloha x_0 a x_f vede na $u_-(t)$ řízení, potom použitím následující transformace

$$\begin{aligned} x_0 &:= -x_0 \\ x_f &:= -x_f \end{aligned} \quad (\text{PROBLÉM JE SYMETRICKÝ PODLE POČÁTKU}) \quad (3.33)$$

převědeme problém na $u_+(t)$ řízení a vypočteme příslušné délky časových intervalů $t_{11}, t_{12}, t_2, t_{31}, t_{32}$. Ty potom použijeme zpět do $u_-(t)$ řízení.

ROZHODOVACÍ ALGORITMUS

Pro řetězec dvou integrátorů lze z geometrie stavového prostoru a polohy \mathbf{x}_0 a \mathbf{x}_f apriori rozhodnout, která sekvence řízení převede systém do koncového stavu. Známe explicitní popis rozdělovací variety, který lze získat přímo z integrálních křivek systému, viz. obrázek 3.17. Víme, že pro dané sekvence řízení je fázové pole generátoru tvořeno parabolickými úseky se stejnou strmostí. To zajistí, že se integrální křivky nemohou křížit a souvisle vyplňují celý stavový prostor. Rozdělovací varieta tak dělí stavový prostor na dvě souvislé oblasti, ve kterých je jednoznačně definováno, jaká sekvence řízení převede systém ze stavu \mathbf{x}_0 do \mathbf{x}_f . Není tedy třeba vyčíslit řešení soustav pro obě sekvence řízení.

Níže uvedený rozhodovací algoritmus je odvozen přímou integrací systému a umožňuje podle vzájemné polohy \mathbf{x}_0 a \mathbf{x}_f jednoznačně rozhodnout, ve které sekvenci řízení je hledané t-optimální řešení úlohy. Algoritmus nalezne hodnotu polohy s_v na rozdělovací křivce, kterou by systém dosáhl příslušným řízením až do okamžiku získání totožné rychlosti s v_f . Potom porovnáním poloh s_f a s_v určí, na které straně rozdělovací křivky se nachází koncový stav a definuje sekvenci řízení.

Věta 3.3 (Algoritmus). *Nechť jsou dána vstupní data pro spuštění algoritmu. Jestliže počáteční a koncový stav splňuje podmínky řešitelnosti, potom lze jednoznačně určit, která sekvence řízení obsahuje t-optimální řešení. Rozhodovací proces je definován následujícím algoritmem:*

vstup: $\mathbf{x}_0, \mathbf{x}_f$

výstup: sekvence řízení $u_{\pm}(t)$

parametry: AM, DM, VM

IF $v_f - v_0 > 0$
THEN

IF $\text{sign}(v_0) \neq \text{sign}(v_f)$ THEN $s_v = s_0 + \frac{v_f^2}{2AM} - \frac{v_0^2}{2DM}$

ELSEIF $v_0 \geq 0$ THEN $s_v = s_0 + \frac{v_f^2 - v_0^2}{2AM}$

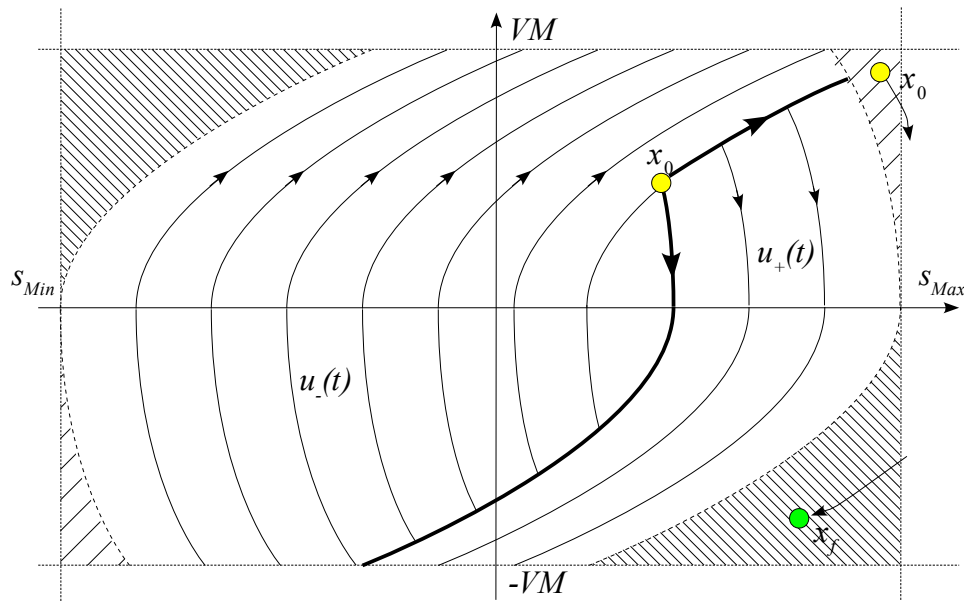
ELSE $s_v = s_0 + \frac{v_f^2 - v_0^2}{2DM}$

ELSE $x_0 = -x_0, x_f = -x_f$

RETURN TO BEGIN

ENDIF

$$u_{\pm}(t) = \text{sign}(s_f - s_v) = \begin{cases} 1: & u_+(t) \\ -1: & u_-(t) \end{cases}$$



Obrázek 3.17: Geometrie stavového prostoru. Varieta (tučně) rozděluje stavový prostor na dvě souvislé oblasti, ve kterých je jednoznačně dáno jaké řízení povede systém ke koncovému stavu. Šrafované oblasti jsou tzn. zakázané oblasti pro x_0 a x_f , ze kterých není možné přejít do jiného přípustného stavu bez překročení omezení.

VÝPOČETNÍ ALGORITMUS

Výsledkem popsaného postupu je jednoduchý algoritmus, který ze vstupních dat rozhodne, zda úloha má řešení a pokud ano, vypočte ho. Zdrojový kód je uveden v příloze B. Jestliže x_0 a x_f leží ve stavovém prostoru v přípustné oblasti $u_{\pm}(t)$, viz obrázek 3.17, řešení musí existovat. Složitost algoritmu je malá a proto ho můžeme využít i jako zpětnovazební nebo korekční regulátor. To samozřejmě přináší další komplikace jako problémy s perturbacemi kolem koncového stavu, diskretizací a další. Ošetření těchto problémů je další z široce zkoumaných kapitol teorie řízení.

Jako ukázkou uvažujme například tyto vstupní hodnoty:

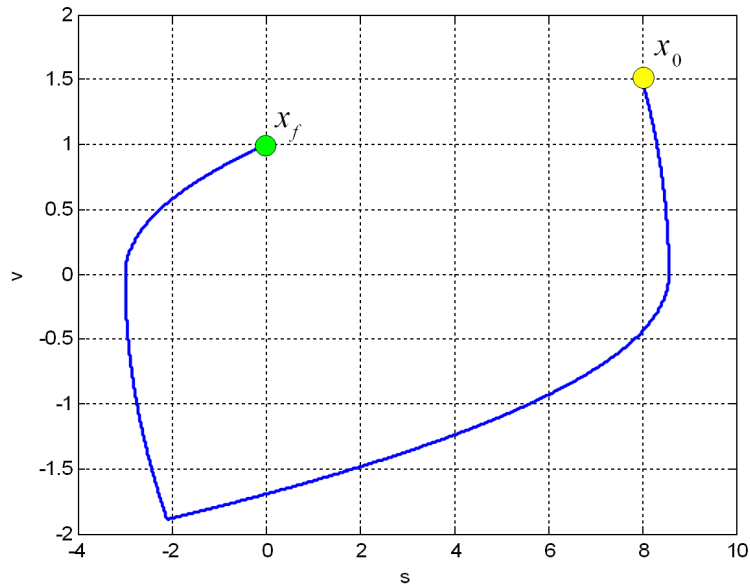
maximální zrychlení :	$AM = 1/6 \text{ m/s}^2$
maximální brzdění :	$DM = 2 \text{ m/s}^2$
maximální rychlost :	$ VM = 2 \text{ m/s}$
omezení pracovního prostoru :	$s_{Min} = -10, s_{Max} = 10$
poč., konc. stav :	$x_0 = [1.5 \ 8]^T, x_f = [1 \ 0]^T$

Výstupem algoritmu je posloupnost $[t_{11}, t_{12}, t_2, t_{31}, t_{32}]$ v sekvenci $u_{\pm}(t)$ řízení

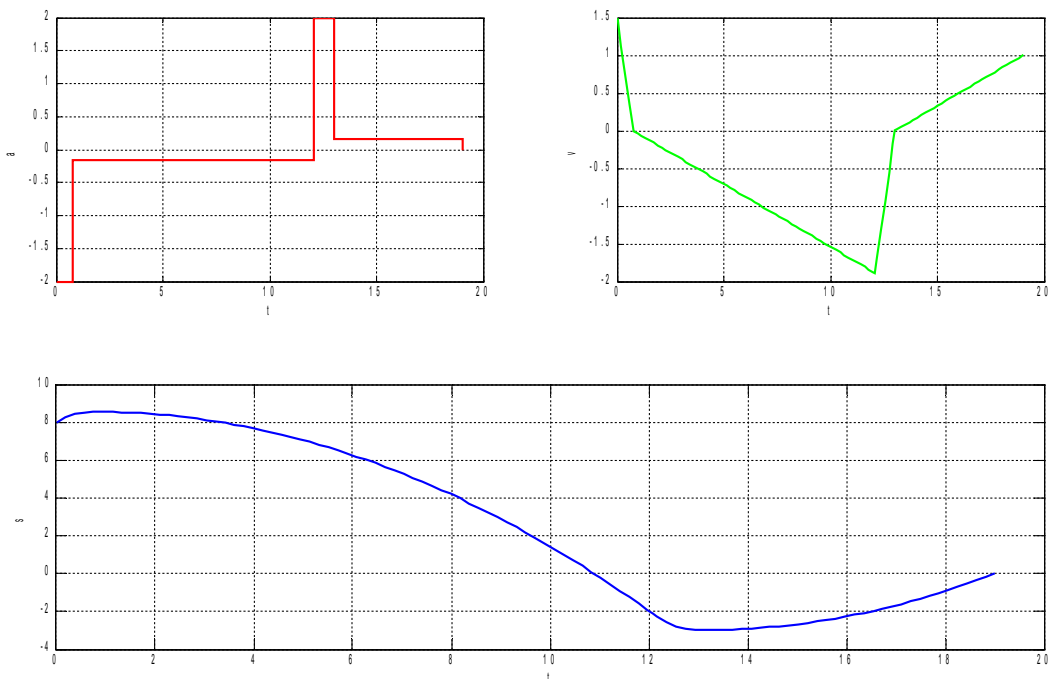
```
U1_MINUS.SOL = 0.7500  12.0000  -0.7188  1.0000  6.0000
                0.7500  11.3171   0       0.9431  6.0000
                0.7500 -11.3171   0      -0.9431  6.0000
```

Zajímáme se pouze o kladná reálná řešení. Pro systém dvou integrátorů existuje vždy pouze jedno takové, které zároveň splňuje podmínku časové optimality (3.6) a minimalizuje funkcionál (3.3). V našem případě je hledaným řešením

```
SOLUTION = [0.7500 11.3171 0 0.9431 6]
```

Obrázek 3.18. T -optimální trajektorie systému ve stavovém prostoru.



Obrázek 3.19: Vstup systému a t -optimální trajektorie jednotlivých stavů v závislosti na čase t .

Poznámka. Gröbnerova báze nám neposkytne odpověď, zda systém překročí omezení na stavový podprostor $[s_{Min}, s_{Max}]$. To lze ovšem velmi snadno otestovat, protože známe \mathbf{x}_0 a \mathbf{x}_f a přesný matematický popis trajektorií. Lze tedy snadno najít maximální a minimální hodnotu $s(t)$ a porovnat ji s hraničními hodnotami. Výpočetní procedura je uvedena v příloze B.

3.4.2 SYSTÉM TŘETÍHO ŘÁDU (3 INTEGRÁTORY S OMEZENÍMI)

Uvažujme systém tří integrátorů

$$\begin{bmatrix} \dot{s} \\ \dot{v} \\ \dot{a} \end{bmatrix} = \underbrace{\begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}}_A \underbrace{\begin{bmatrix} s \\ v \\ a \end{bmatrix}}_x + \underbrace{\begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}}_B u \quad (3.34)$$

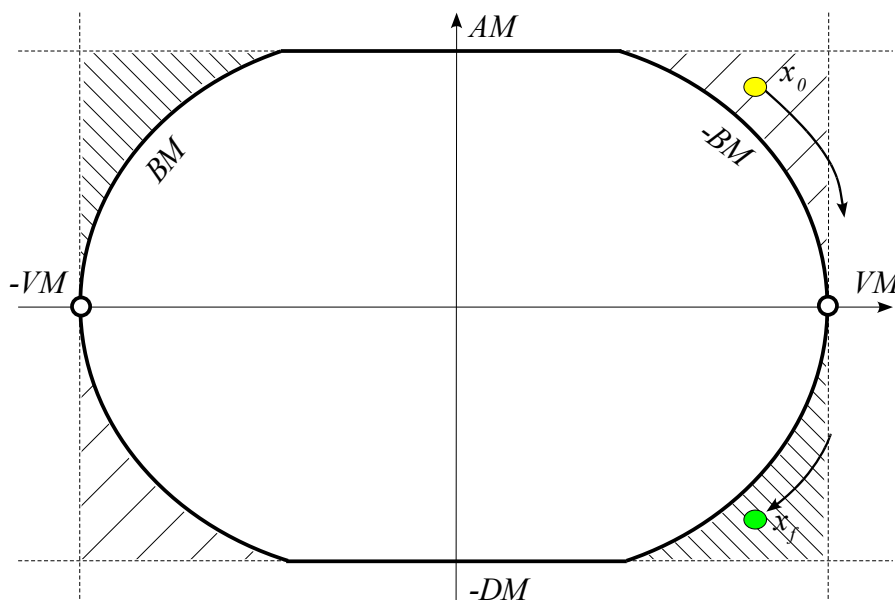
s omezeními na stav i vstup systému

$$\begin{aligned} s_{Min} &\leq s(t) \leq s_{Max} \\ -VM &\leq v(t) \leq VM \\ -DM &\leq a(t) \leq DM \\ -BM &\leq u(t) \leq BM \end{aligned} \quad (3.35)$$

a obecným počátkem $\mathbf{x}_0 = [s_0, v_0, a_0]$ a koncem $\mathbf{x}_f = [s_f, v_f, a_f]$. Předpokládejme, že počáteční stav \mathbf{x}_0 vyhovuje omezením, respektive, že je možné ho přípustným řízením opustit bez porušení omezení. Dále předpokládejme, že koncový stav \mathbf{x}_f je z pohledu řízení a omezení dosažitelný.

Problém 3.3. Nalezněte řízení u , které převede systém (3.34) z počátečního stavu \mathbf{x}_0 do koncového stavu \mathbf{x}_f za minimální čas při respektování omezení (3.35).

Pro řetězec tří integrátorů dochází k velikému zesložiténí. Bohužel se pro obecný přesun zatím nepodařilo najít rozdělující varietu, podobnou jako v případě druhého řádu, viz obrázek 3.17. Nelze již tedy apriori rozhodnout, kterou sekvencí řízení použít. Je nutné prohledávat všechny soustavy, jak pro $u_+(t)$, tak pro $u_-(t)$. Tento fakt je způsobem tím, že rozdělující varietu není pro obecný počátek a konec souvislou množinou. Obecně tedy může nastat situace, že existuje kladné reálné řešení pro obě sekvence řízení. T-optimálním je samozřejmě to, které minimalizuje funkcionál (3.3). Během výpočtů Gröbnerových bází dochází u některých soustav k problémům se specializací. Počet soustav vzroste na 16 a jejich složitost také. Nicméně všechny soustavy lze vyřešit obecně a výsledné vzorce pro vyčíslení časových intervalů je možné algoritmizovat, viz dále.



Obrázek 3.20: Stavový podprostor v - a pro řetězec 3 integrátorů s omezeními, s vyznačením zakázaných oblastí pro počáteční a koncový stav.

Vyznačení všech omezení ve stavovém podprostoru v - a je zobrazeno na obrázku 3.20, podobně jako v případě podprostoru s - v u systému druhého řádu. V tomto případě je podprostor v - a symetrický i vůči vstupu $u(t)$, který má symetrické omezení. Poznamenejme, že maximální rychlost je v tomto podprostoru reprezentována pouze bodem $[\pm VM, 0]$, kde systém může setrvat. Z pohledu této projekce se stav systému v tomto bodě může jevit jako nehybný. Z těchto předpokladů a geometrie stavového prostoru dostáváme následující hypotézu.

Hypotéza 3.4. Časově-optimální řízení pro systém (3.34) s omezeními (3.35) vede na **bang-zero-bang** řízení s maximálním počtem **sedmi** časových intervalů. Existují dvě přípustné sekvence řízení

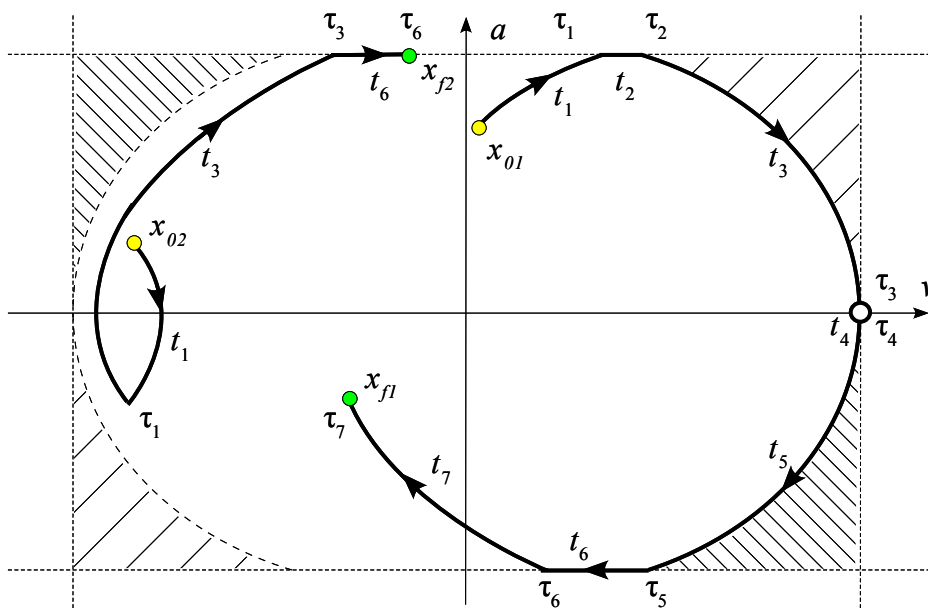
$$u_+(t) = \begin{cases} BM, & t \in [0, \tau_1) \\ 0, & t \in [\tau_1, \tau_2) \\ -BM, & t \in [\tau_2, \tau_3) \\ 0, & t \in [\tau_3, \tau_4) \\ -BM, & t \in [\tau_4, \tau_5) \\ 0, & t \in [\tau_5, \tau_6) \\ BM, & t \in [\tau_6, \tau_7) \end{cases}, \quad u_-(t) = \begin{cases} -BM, & t \in [0, \tau_1) \\ 0, & t \in [\tau_1, \tau_2) \\ BM, & t \in [\tau_2, \tau_3) \\ 0, & t \in [\tau_3, \tau_4) \\ BM, & t \in [\tau_4, \tau_5) \\ 0, & t \in [\tau_5, \tau_6) \\ -BM, & t \in [\tau_6, \tau_7) \end{cases} \quad (3.36)$$

kde $\tau_k = \sum_{i=1}^k t_i$, $i=1..7$ je časové dělení sekvence řízení a t_i , $i=1..7$ je doba, po kterou je řízení

konstantní. Význam těchto proměnných je patrný z obrázku 3.21.

Ze znalosti počátečního a koncového stavu lze prohledáním řešení v obou sekvencích jednoznačně určit, která řídicí sekvence převede systém do koncového stavu, nebo zda řízení vzhledem k daným omezením neexistuje. Navíc jednotlivé časové intervaly mohou zaniknout v případě, kdy systém nedosáhne daných stavových omezení. T-optimální řešení je takové, které minimalizuje funkcionál

$$\min \tau_7 = \min \sum_{i=1}^7 t_i \quad (3.37)$$



Obrázek 3.21: Příklady trajektorií v omezeném stavovém podprostoru v - a systému 3 integrátů s vyznačením časových intervalů.

Dostáváme tedy zobecnění problému *BAVS*, který uvažoval počáteční a koncovou rychlost i zrychlení nulové. Počet časových intervalů je v obou případech shodný. Obrázek 3.21 naznačuje možné trajektorie systému s rozložením časových intervalů. Ze znalosti řídicích sekvencí (3.36), \mathbf{x}_0 a \mathbf{x}_f můžeme nalézt algebraické rovnice stavového přechodu. Postup výpočtu bude obdobný jako u generátoru 2. řádu. Budeme opět uvažovat pouze kladnou sekvenci řízení $u_+(t)$. Pro druhou sekvenci $u_-(t)$ využijeme transformaci (3.42) do $u_+(t)$, obdobně jako v předchozí kapitole.

Pro $u_+(t)$ dostáváme

$$\begin{aligned}
 eq_1: \quad a_f &= a_0 + BM(t_1 - t_5 + t_7 - t_3) \\
 eq_2: \quad v_f &= v_0 + a_0(t_1 + t_2 + t_3 + t_4 + t_5 + t_6 + t_7) \\
 &\quad + BM \left(t_1(t_2 + t_3 + t_4 + t_5 + t_6 + t_7) - t_3(t_4 + t_5 + t_6 + t_7) - t_5(t_6 + t_7) - \frac{t_3^2}{2} - \frac{t_5^2}{2} + \frac{t_7^2}{2} + \frac{t_1^2}{2} \right) \\
 eq_3: \quad s_f &= s_0 + (t_1 + t_2 + t_3 + t_4 + t_5 + t_6 + t_7)v_0 + 1/2 a_0(t_1 + t_2 + t_3 + t_4 + t_5 + t_6 + t_7)^2 \\
 &\quad + 1/6 BM \left[t_1^3 - t_3^3 - t_5^3 - 3t_5^2 t_6 - 3t_5 t_6^2 - 3t_5^2 t_7 - 6t_5 t_6 t_7 - 3t_5 t_7^2 + t_7^3 \right. \\
 &\quad \quad - 3t_3^2(t_4 + t_5 + t_6 + t_7) - 3t_3(t_4 + t_5 + t_6 + t_7)^2 \\
 &\quad \quad \left. + 3t_1^2(t_2 + t_3 + t_4 + t_5 + t_6 + t_7) + 3t_1(t_2 + t_3 + t_4 + t_5 + t_6 + t_7)^2 \right]
 \end{aligned} \tag{3.38}$$

Z omezení (3.35) dostáváme další rovnice, které nám úlohu rozdělí na případ, kdy systém dosáhne své absolutní maximální rychlosti nebo zrychlení

$$\begin{aligned}
 eq_{4a1}: \quad VM &= v_0 + a_0(t_1 + t_2 + t_3) + BM(t_1 t_2 + t_1 t_3 - t_3^2/2 + t_1^2/2) \\
 eq_{4a2}: \quad VM &= v_0 + a_0(t_1 + t_2 + t_3 + t_4) + BM(t_1 t_2 + t_1 t_3 + t_1 t_4 - t_3 t_4 - t_3^2/2 + t_1^2/2) \\
 eq_{5a}: \quad t_1 &= \frac{AM - a_0}{BM} \\
 eq_{6a}: \quad t_7 &= \frac{DM + a_f}{BM}
 \end{aligned} \tag{3.39}$$

a na případ, kdy k tomu nedojde

$$\begin{aligned}
 eq_{4b1}: \quad t_{35} &= t_3 + t_5 \\
 eq_{4b2}: \quad t_4 &= 0 \\
 eq_{5b}: \quad t_2 &= 0 \\
 eq_{6b}: \quad t_6 &= 0
 \end{aligned} \tag{3.40}$$

Celkem tedy dostáváme 8 soustav pro $u_+(t)$

$$\begin{aligned}
 F_1 &= \{eq_1, eq_2, eq_3, eq_{4a1}, eq_{4a2}, eq_{5a}, eq_{6a}\} \\
 F_2 &= \{eq_1, eq_2, eq_3, eq_{4a1}, eq_{4a2}, eq_{5a}, eq_{6b}\} \\
 F_3 &= \{eq_1, eq_2, eq_3, eq_{4b1}, eq_{4b2}, eq_{5a}, eq_{6a}\} \\
 F_4 &= \{eq_1, eq_2, eq_3, eq_{4b1}, eq_{4b2}, eq_{5a}, eq_{6b}\} \\
 F_5 &= \{eq_1, eq_2, eq_3, eq_{4a1}, eq_{4a2}, eq_{5b}, eq_{6a}\} \\
 F_6 &= \{eq_1, eq_2, eq_3, eq_{4a1}, eq_{4a2}, eq_{5b}, eq_{6b}\} \\
 F_7 &= \{eq_1, eq_2, eq_3, eq_{4b1}, eq_{4b2}, eq_{5b}, eq_{6a}\} \\
 F_8 &= \{eq_1, eq_2, eq_3, eq_{4b1}, eq_{4b2}, eq_{5b}, eq_{6b}\}
 \end{aligned} \tag{3.41}$$

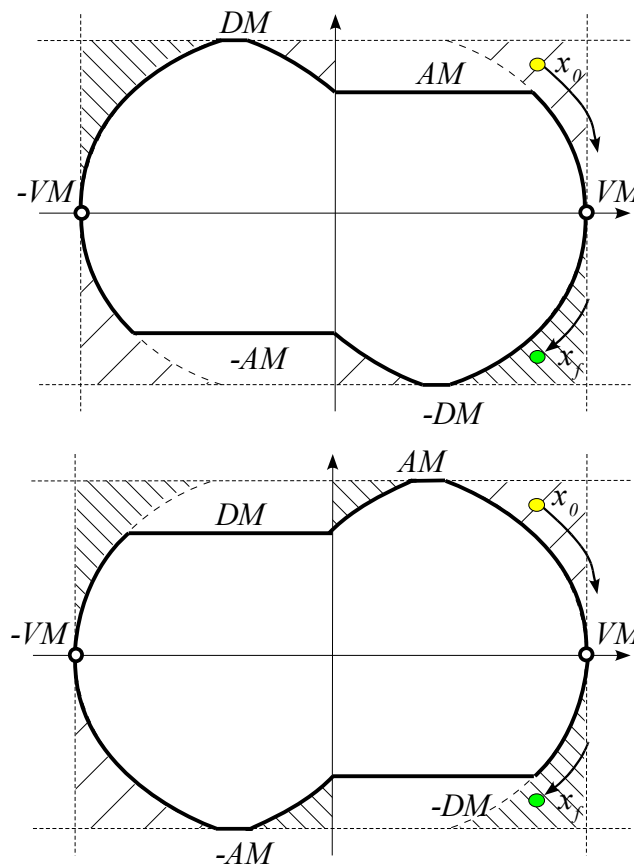
Pro $u_-(t)$ bychom dostali dalších 8 soustav. V důsledku symetrie úlohy můžeme ale využít transformace (3.42) a hledat řešení v sekvenci $u_-(t)$ pomocí rovnic sekvence $u_+(t)$, podobně jako v případě 2.řádu

$$\begin{aligned}
 \mathbf{x}_0 &:= -\mathbf{x}_0 \\
 \mathbf{x}_f &:= -\mathbf{x}_f \quad (\text{PROBLÉM JE SYMETRICKÝ PODLE POČÁTKU}) \\
 AM &:= DM \\
 DM &:= AM
 \end{aligned} \quad (3.42)$$

Hledané řešení se nachází v některé z 16-ti soustav nebo, v případě kdy $\mathbf{x}_0, \mathbf{x}_f$ leží v zakázaných oblastech (obrázek 3.20), neexistuje. Další postup je obecně shodný s postupem pro 2.řád s tou výjimkou, že nyní musíme prohledávat všechny soustavy. Popis a řešení soustav (3.41) bude detailně rozebráno v kapitole 3.5. Dodejme, že překročení mezních poloh s_{Min}, s_{Max} je opět nutné testovat zvlášť. Tento test je možné provést až po nalezení řešení, ale je už triviální.

Poznámka. Porovnáním omezení (3.20) a (3.35) zjistíme, že tvar stavových omezení není u 2. a 3.řádu zcela ekvivalentní. U 2.řádu se mění velikost zrychlení/zpomalení při změně znaménka rychlosti. Na tuto změnu můžeme z praktického hlediska nahlížet jako na zrychlování a brždění systému (například automobilu s rozdílnými parametry akcelerace a brždění). U třetího řádu má omezení jiný charakter. Jedná se o meze integrátoru zrychlení, které jsou závislé pouze na směru zrychlení, nikoli na změně znaménka rychlosti (například řízení výtahu). Ačkoli omezení na zrychlení a zpomalení je u 2.řádu dáno mezemi vstupní funkce $u(t)$, u 3.řádu je to již vnitřní omezení na prvním integrátoru a odtud plyne zesložiténí úlohy.

Pokud bychom chtěli nastavit ekvivalentní omezení jako u 2.řádu, musel by stavový prostor vypadat následovně. Podle vzájemné velikosti AM a DM bychom vybrali jednu z variant vyznačených na obrázku 3.22. Touto změnou by došlo k dalšímu nárůstu časových intervalů a i počtu řešených soustav. Naštěstí je možné detekovat průchod řešení „novými omezeními“ a rozdělit celý pohyb do dílčích pohybů tak, aby je bylo možné spočítat původním postupem.



Obrázek 3.22: Projekce variant stavového prostoru do roviny v-a.

3.5 ALGORITMUS *GBAVS*

Tato kapitola je věnována algoritmu, který řeší problém t-optimálního řízení tří integrátorů s konstantními omezeními na všechny stavy a vstup systému, definovaného v předchozí kapitole. Pro srovnání složitosti budou uvedeny výpočty časových intervalů pro tři standardní typy přesunů systému: *obecný pohyb*, *pohyb z/do konstantní rychlosti*, *pohyb z klidu do klidu*. Postup a funkce algoritmu je obdobná jako pro systém 2.řádu. Algoritmus byl získán rigorózním postupem. Jedná se o analytický postup, bez využití heuristických metod. Pouze, z důvodu jednoduchosti, je využito numerické hledání kořenů polynomu maximálně 4.stupně. Pro tento stupeň polynomu jsou ale známy analytické vztahy pro výpočet kořenů, takže není porušeno předchozí tvrzení.

Jádro algoritmu je založeno na hypotéze 3.4. Ta říká, že řízení bude typu *bang-zero-bang* a bude nabývat maximálně sedmi časových intervalů, jejichž délku chceme určit. Počet intervalů a jejich délka je plně specifikována počátečním $\mathbf{x}_0=[s_0, v_0, a_0]$ a koncovým stavem $\mathbf{x}_f=[s_f, v_f, a_f]$ a omezeními (3.35). Platí-li tato hypotéza, potom následující algoritmus *GBAVS* poskytuje časově optimální řešení uvažované úlohy. V případě, že hypotéza 3.4 neplatí, obdržíme sub-optimální řešení. Ačkoliv exaktní důkaz hypotézy není podle našich znalostí k dispozici, je velmi mnoho dobrých důvodů (plynoucích například z principu maxima) předpokládat, že hypotéza platí.

Z těchto předpokladů vyplývá sestavení osmi soustav polynomiálních rovnic (3.41), každou pro jiný průběh řízení, obdobně jako v kapitole 3.3.1. Provedeme rozbor jednotlivých soustav (3.41). Pro každou soustavu budeme ilustračně vykreslovat profil křivky zrychlení, který je pro danou soustavu specifický. Níže uvedený popis bude proveden bez ztráty obecnosti pouze pro řídicí sekvenci $u_+(t)$, viz kapitola 3.4. Fráze *vyhovuje omezením*, bude znamenat, že omezení *AM*, *DM*, *VM*, *BM* nejsou porušena. Omezení na pracovní prostor S_{min}, S_{max} nebude v dalším bráno v úvahu. Je nutné ho testovat zvlášť, viz komentář dříve. V případě, že tvar trajektorie omezení nerespektuje, je třeba po výpočtu jednotlivých časových intervalů tato omezení otestovat následujícím způsobem

- test omezení VM

$$VM - v_0 - a_0(t_1 + t_2 + t_3) - BM \left(t_1 t_2 + t_1 t_3 - \frac{t_3^2}{2} + \frac{t_1^2}{2} \right) = \begin{cases} \geq 0: & \text{'řešení splňuje omezení VM'} \\ < 0: & \text{'řešení neexistuje'} \end{cases} \quad (3.43)$$

- test omezení AM

$$AM - a_0 - BM t_1 = \begin{cases} \geq 0: & \text{'řešení splňuje omezení AM'} \\ < 0: & \text{'řešení neexistuje'} \end{cases} \quad (3.44)$$

- test omezení DM

$$DM + a_f - BM t_7 = \begin{cases} \geq 0: & \text{'řešení splňuje omezení DM'} \\ < 0: & \text{'řešení neexistuje'} \end{cases} \quad (3.45)$$

kde jednotlivé vztahy jsou získány přímou integrací systému. Hledaným řešením úlohy je pak takové řízení $u^*(t)$, jehož trajektorie existuje, splňuje podmínky (3.43),(3.44) a (3.45) a minimalizuje kritérium (3.11).

Poznámka. Jak již bylo řečeno v úvodu, Gröbnerova báze nám neposkytne přímo řešení polynomiální soustavy, ale může výrazně usnadnit jeho nalezení. To je i případ soustav této optimální úlohy. Každá soustava odpovídá příslušnému tvaru trajektorie. V některých případech je možné vyčíslit určité časové intervaly přímo ze vstupních dat a dosadit je do zbývajících rovnic.

Volba typu uspořádání byla proto specifikována tak, aby výpočet *GB* využíval těchto informací. Upozorníme ještě na poslední soustavu F_8 , u které dochází ke specializaci řešení, o které byla zmínka v kapitole 2.6.1. Ta zamezí využití rovnic v tomto tvaru. Je třeba se pokusit nalézt jinou soustavu bazových rovnic, která by generovala řešení bez specializace. Jedině tak jsme schopni zajistit, že výpočetní algoritmus bude pracovat na celém rozsahu vstupních dat, bez singularit. Takové řešení lze skutečně nalézt a bude podrobně představeno dále.

3.5.1 ZÁKLADNÍ STRUKTURA ALGORITMU *GBAVS*

Řešení problému 3.3 je možné nalézt následujícím postupem, který obsahuje několik důležitých výpočetních a rozhodovacích kroků. Hlavní krok výpočtu využívá soustav (3.41), které byly obecně vyřešeny pomocí *GB*. Řešení všech soustav je nalezeno analytickým postupem a vyjádřeno v explicitním tvaru. Krok 3 algoritmu je tedy redukován na pouhé dosazení do dílčích vztahů pro délky časových intervalů, viz dále. Tímto způsobem je snadné přímo vyčíslit řešení soustav, kromě několika hledání kořenů polynomů maximálně čtvrtého stupně.

ALGORITMUS *GBAVS*

vstup: x_0, x_f, BM, AM, VM

výstup: t_1, \dots, t_7 , řídicí strategie u_+ nebo u_-

krok 1: Test řešitelnosti $x_0 = [a_0, v_0, s_0]$ a $x_f = [a_f, v_f, s_f]$ (zakázané oblasti).

Testováním vrcholů rozjezdových a dojezdových parabol s hraničními parabolami

$$v_{0\text{lim}} = \pm v_0 \mp \frac{a_0^2}{2BM}, \quad v_{f\text{lim}} = \pm v_f \mp \frac{a_f^2}{2BM}$$

dostaneme podmínku pro rychlosti $VM \geq \{|v_{0\text{lim}}|, |v_{f\text{lim}}|\}$. Testování podmínek zrychlení a poloh je triviální. Jestliže je x_0, x_f v přípustné oblasti, potom řešení musí existovat.

krok 2: Vytvoření osmi soustav polynomiálních rovnic $S_i, i = 1, 2, \dots, 8$ pro všechny možné kombinace aktivních omezení, jak pro sekvenci u_+ tak pro sekvenci u_- .

krok 3: Vyčíslení délek časových intervalů t_1, \dots, t_7 všech soustav s vynecháním záporných a komplexních řešení.

krok 4: Test respektování omezení pomocí (3.43), (3.44) a (3.45).

krok 5: Pokud stále existuje víc než jedno pozitivně definitní řešení, potom je vybráno to, které minimalizuje (3.37). Podle řešení je vybrána i příslušná sekvence řízení $u_{\pm}(t)$.

3.5.2 OBECNÝ POČÁTEK A KONEC

V této kapitole je uveřejněn analytický popis řešení všech osmi soustav polynomiálních rovnic⁸ (3.41) pro délky časových intervalů t_1, \dots, t_7 . Explicitní zápis řešení je získán pomocí *GB* s různým typem uspořádání pro každou soustavu. Změnou uspořádání bylo možné redukovat složitost výstupních rovnic. Protože je známa trajektorie pohybu pro každou dílčí soustavu, je možné některé délky časových intervalů apriori určit a není nutné vždy využít lexikografické uspořádání za účelem eliminace proměnných. To není v některých případech, vzhledem ke složitosti rovnic, ani možné vyčíslit.

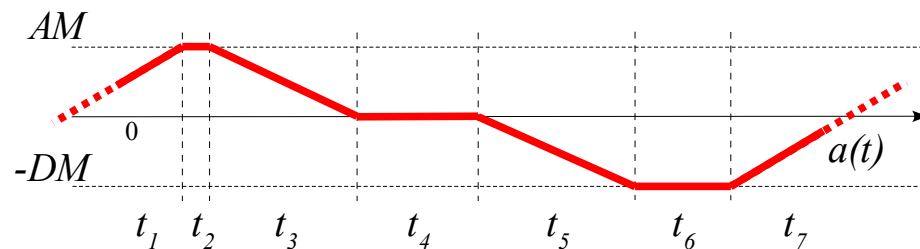
⁸ Níže uvedené rovnice jsou z důvodu jejich délky získané pouze exportem z použitých softwarů Maple a Mathematica. Proto někdy zcela nesplňují správnost matematického zápisu, ani velikost použitého písma. Tímto prosím o shovívavost.

Pro obecný počáteční a koncový stav je nutné najít všechna pozitivně definitní řešení pro obě sekvence řízení a vybrat časově optimální, pro které platí (3.37).

ŘEŠENÍ SOUSTAVY F_1

Soustava $F_1 = \{eq_1, eq_2, eq_3, eq_{4a1}, eq_{4a2}, eq_{5a}, eq_{6a}\}$ odpovídá situaci, kdy systém dosáhne všech omezení. Pokud je řešením soustavy 7 reálných časových intervalů, potom řešení zároveň vyhovuje omezením a je hledaným řešením úlohy.

Pomocí GB s uspořádáním $\mathbf{tdeg}(t[1], t[2], t[3], t[4], t[5], t[6], t[7])$ lze vyjádřit explicitní vzorce délek jednotlivých časových intervalů.



Obrázek 3.23: Typický tvar trajektorie zrychlení pro soustavu F_1 .

Výstupní rovnice řešení soustavy F_1 použité v algoritmu $GBAVS$:

$$s1t1 = \frac{AM - a_0}{BM}$$

$$s1t2 = \frac{-2 AM^2 + a_0^2 + 2 BM (VM - v_0)}{2 AM BM}$$

$$s1t3 = \frac{AM}{BM}$$

$$s1t4 = \frac{12 AM^2 DM \left(\frac{a_0^2}{2} + BM (-VM - v_0) \right) + DM \left(3 a_0^4 - 12 BM a_0^2 v_0 + 12 BM^2 (-VM^2 + v_0^2) \right) + AM \left(3 a_f^4 + DM^2 \left(6 a_f^2 + 12 BM (-VM - v_f) \right) - 12 BM a_f^2 v_f + 12 BM^2 (-VM^2 + v_f^2) \right) + DM \left(8 (-a_0^3 + a_f^3) + 24 BM^2 (-s_0 + s_f) + 24 BM (a_0 v_0 - a_f v_f) \right)}{24 AM BM^2 DM VM}$$

$$s1t5 = \frac{DM}{BM}$$

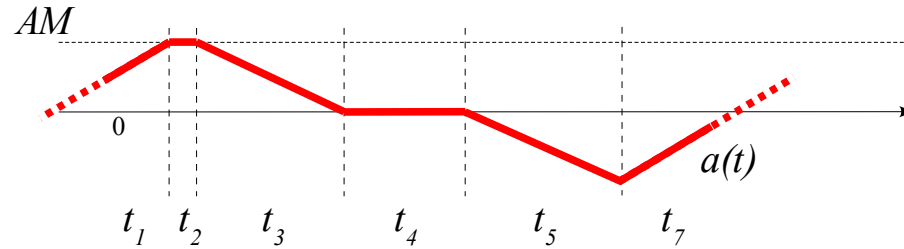
$$s1t6 = \frac{-2 DM^2 + a_f^2 + 2 BM (VM - v_f)}{2 BM DM}$$

$$s1t7 = \frac{DM + a_f}{BM}$$

ŘEŠENÍ SOUSTAVY F_2

Soustava $F_2 = \{eq_1, eq_2, eq_3, eq_{4a1}, eq_{4a2}, eq_{5a}, eq_{6b}\}$ nerespektuje DM omezení. Pokud je řešením soustavy 7 nezáporných reálných časových intervalů, potom je nutné otestovat omezení DM podle (3.45).

Pomocí GB s uspořádáním $tdeg(t[1], t[2], t[3], t[4], t[5], t[6], t[7])$ lze vyjádřit explicitní vzorce délek jednotlivých časových intervalů.



Obrázek 3.24: Typický tvar trajektorie zrychlení pro soustavu F_2 .

Výstupní rovnice řešení soustavy F_2 použité v algoritmu $GBAVS$:

$$s2t1 = s1t1$$

$$s2t2 = s1t2$$

$$s2t3 = s1t3$$

$$s2t4 = \frac{3 a_0^4 + 6 AM^2 (a_0^2 + 2 BM (-VM - v_0)) - 12 BM a_0^2 v_0 + 12 BM^2 (-VM^2 + v_0^2) + 6 AM (a_f^2 + 2 BM (-VM - v_f)) \sqrt{2 a_f^2 + 4 BM (VM - v_f)} + 8 AM (-a_0^3 + a_f^3 + 3 BM^2 (-s_0 + s_f) + 3 BM (a_0 v_0 - a_f v_f))}{24 AM BM^2 VM}$$

$$s2t5 = \frac{\sqrt{2 a_f^2 + 4 BM (VM - v_f)}}{2 BM}$$

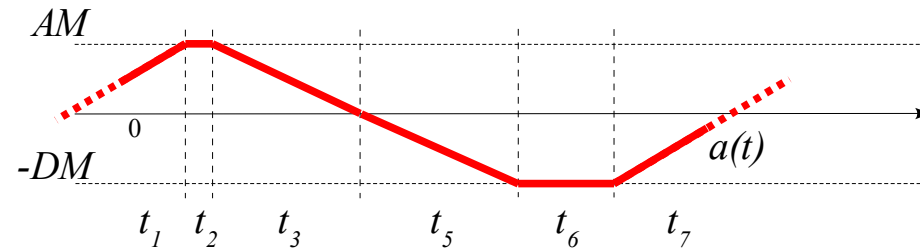
$$s2t6 = 0$$

$$s2t7 = s2t5 + \frac{a_f}{BM}$$

ŘEŠENÍ SOUSTAVY F_3

Soustava $F_3 = \{eq_1, eq_2, eq_3, eq_{4b1}, eq_{4b2}, eq_{5a}, eq_{6a}\}$ nerespektuje VM omezení. Pokud je řešením soustavy 7 nezáporných reálných časových intervalů, potom je nutné otestovat omezení VM podle (3.43)

Pomocí GB s uspořádáním $tdeg(t[1], t[2], t[3], t[4], t[5], t[6], t[7])$ lze vyjádřit explicitní vzorce délek jednotlivých časových intervalů.



Obrázek 3.25: Typický tvar trajektorie zrychlení pro soustavu F_3 .

Výstupní rovnice řešení soustavy F_3 použité v algoritmu $GBAVS$:

$$p0 = 12 DM^4 - 3 a_0^4 - 12 DM^2 a_f^2 + 3 a_f^4 + 6 AM^2 (2 DM^2 - a_0^2 - a_f^2) + 8 AM (3 DM^3 + a_0^3 - 3 DM a_f^2 - a_f^3) + 12 BM^2 (2 AM (s_0 - s_f) - v_0^2 + v_f^2) + 12 BM (a_0^2 v_0 + 2 DM^2 v_f - a_f^2 v_f + AM^2 (v_0 + v_f) + 2 AM (-a_0 v_0 + 2 DM v_f + a_f v_f))$$

$$p1 = 24 BM (AM + DM) \left(\frac{AM DM}{2} + DM^2 - \frac{a_f^2}{2} + BM v_f \right)$$

$$p2 = 12 BM^2 DM (AM + DM)$$

$$s3t6 = \text{roots}[p0 + s3t6 (p1 + p2 s3t6) = 0, s3t6]$$

$$s3t1 = s1t1$$

$$s3t2 = \frac{2 (-AM^2 + DM^2) + a_0^2 - a_f^2 + 2 BM (DM s3t6 - v_0 + v_f)}{2 AM BM}$$

$$s3t3 = s1t3$$

$$s3t4 = 0$$

$$s3t5 = s1t5$$

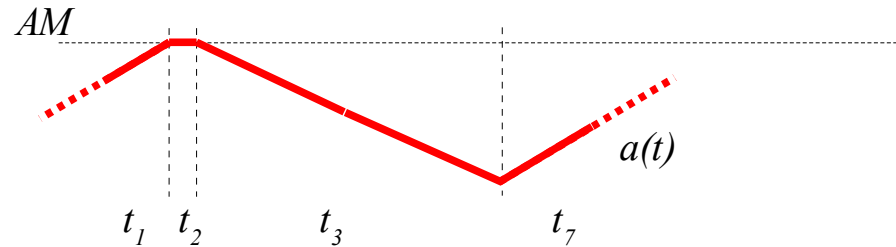
$$s3t6 = \text{roots}[p0 + s3t6 (p1 + p2 s3t6) = 0, s3t6]$$

$$s3t7 = s1t7$$

ŘEŠENÍ SOUSTAVY F_4

Soustava $F_4 = \{eq_1, eq_2, eq_3, eq_{4b1}, eq_{4b2}, eq_{5a}, eq_{6b}\}$ nerespektuje DM a VM omezení. Pokud je řešením soustavy 7 nezáporných reálných časových intervalů, potom je nutné otestovat omezení DM a VM omezení stejně jako v soustavách F_2 a F_3 .

Pomocí GB s uspořádáním $\mathbf{plex}(t[1], t[2], t[3], t[4], t[5], t[6], t[7])$ lze vyjádřit explicitní vzorce délek jednotlivých časových intervalů.



Obrázek 3.26: Typický tvar trajektorie zrychlení pro soustavu F_4 .

Výstupní rovnice řešení soustavy F_4 použité v algoritmu $GBAVS$:

$$p0 == -3 a_0^4 + 3 a_f^4 + 12 BM (a_0^2 v_0 + a_f^2 v_f) + 12 BM^2 (-v_0^2 + v_f^2) + 6 AM^2 (-a_0^2 + a_f^2 + 2 BM (v_0 + v_f)) + 8 AM (a_0^3 - a_f^3 + 3 BM^2 (s_0 - s_f) + 3 BM (-a_0 v_0 - a_f v_f))$$

$$p1 == -48 BM (-AM + a_f) \left(-\frac{AM a_f}{2} + \frac{a_f^2}{2} + BM v_f \right)$$

$$p2 == 12 BM^2 (AM^2 - 6 AM a_f + 5 a_f^2 + 2 BM v_f)$$

$$p3 == -24 BM^3 (-AM + 2 a_f)$$

$$p4 == 12 BM^4$$

$$s4t1 == s1t1$$

$$s4t2 == \frac{-2 AM^2 + 2 BM^2 s4t7^2 + a_0^2 + a_f^2 + 2 BM (-2 s4t7 a_f - v_0 + v_f)}{2 AM BM}$$

$$s4t3 == s1t3 + s4t7 - \frac{a_f}{BM}$$

$$s4t4 == 0$$

$$s4t5 == 0$$

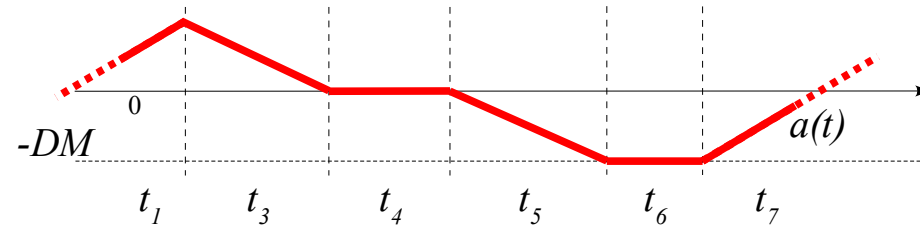
$$s4t6 == 0$$

$$r4t7 == \text{roots}[p0 + p1 s4t7 + p2 s4t7^2 + p3 s4t7^3 + p4 s4t7^4 == 0, s4t7]$$

ŘEŠENÍ SOUSTAVY F_5

Soustava $F_5 = \{eq_1, eq_2, eq_3, eq_{4a1}, eq_{4a2}, eq_{5b}, eq_{6a}\}$ nerespektuje AM omezení. Pokud je řešením soustavy 7 nezáporných reálných časových intervalů, potom je nutné otestovat omezení AM podle (3.44).

Pomocí GB s uspořádáním $\mathbf{tdeg}(\mathbf{t}[7], \mathbf{t}[6], \mathbf{t}[5], \mathbf{t}[4], \mathbf{t}[3], \mathbf{t}[2], \mathbf{t}[1])$ lze vyjádřit explicitní vzorce délek jednotlivých časových intervalů. Soustava F_5 je výpočetně ekvivalentní se soustavou F_2 .



Obrázek 3.27: Typický tvar trajektorie zrychlení pro soustavu F_5 .

Výstupní rovnice řešení soustavy F_5 použité v algoritmu $GBAVS$:

$$s5t1 == s5t3 - \frac{a_0}{BM}$$

$$s5t2 == 0$$

$$s5t3 == \frac{\sqrt{2 a_0^2 + 4 BM (VM - v_0)}}{2 BM}$$

$$s5t4 == \frac{3 a_f^4 + 6 DM (a_0^2 + 2 BM (-VM - v_0)) \sqrt{2 a_0^2 + 4 BM (VM - v_0)} + 6 DM^2 (a_f^2 + 2 BM (-VM - v_f)) - 12 BM a_f^2 v_f + 12 BM^2 (-VM^2 + v_f^2) + 8 DM (-a_0^3 + a_f^3 + 3 BM^2 (-s_0 + s_f) + 3 BM (a_0 v_0 - a_f v_f))}{24 BM^2 DM VM}$$

$$s5t5 == s1t5$$

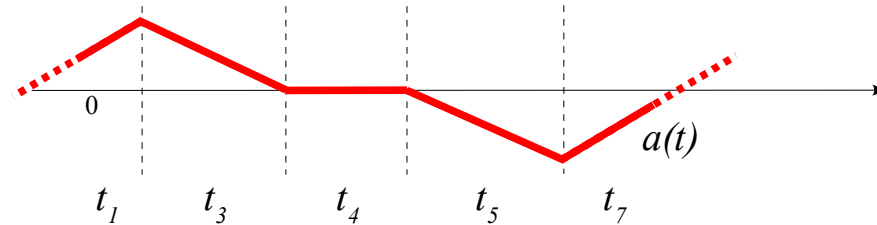
$$s5t6 == s1t6$$

$$s5t7 == s1t7$$

ŘEŠENÍ SOUSTAVY F_6

Soustava $F_6 = \{eq_1, eq_2, eq_3, eq_{4a1}, eq_{4a2}, eq_{5b}, eq_{6b}\}$ nerespektuje AM a DM omezení. Pokud je řešením soustavy 7 nezáporných reálných časových intervalů, potom je nutné otestovat omezení AM a DM omezení stejně jako v soustavách F_2 a F_5 .

Pomocí GB s uspořádáním $\mathbf{tdeg}(t[1], t[2], t[3], t[4], t[5], t[6], t[7])$ lze vyjádřit explicitní vzorce délek jednotlivých časových intervalů.



Obrázek 3.28: Typický tvar trajektorie zrychlení pro soustavu F_6 .

Výstupní rovnice řešení soustavy F_6 použité v algoritmu $GBAVS$:

$$s_{6t1} = s_{5t1}$$

$$s_{6t2} = 0$$

$$s_{6t3} = s_{5t3}$$

$$s_{6t4} = \frac{-4 a_0^3 + 4 a_f^3 - 12 BM^2 (s_0 - s_f) + 12 BM a_0 v_0 + 2 BM t_3 (3 a_0^2 - 6 BM (VM + v_0)) - 12 BM a_f v_f + 2 BM t_5 (3 a_f^2 - 6 BM (VM + v_f))}{12 BM^2 VM}$$

$$s_{6t5} = s_{2t5}$$

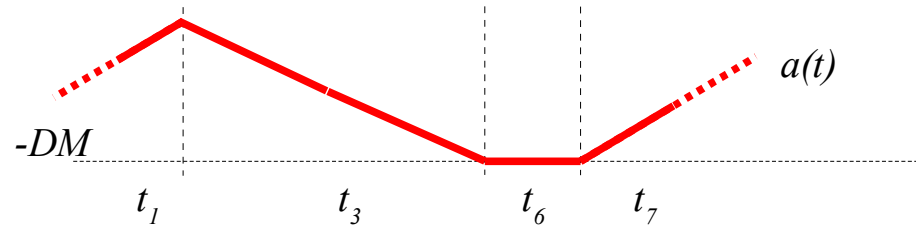
$$s_{6t6} = 0$$

$$s_{6t7} = s_{2t7}$$

ŘEŠENÍ SOUSTAVY F_7

Soustava $F_7 = \{eq_1, eq_2, eq_3, eq_{4b1}, eq_{4b2}, eq_{5b}, eq_{6a}\}$ nerespektuje AM a VM omezení. Pokud je řešením soustavy 7 nezáporných reálných časových intervalů, potom je nutné otestovat omezení AM a VM omezení stejně jako v soustavách F_5 a F_3 . Soustava F_7 je výpočetně ekvivalentní se soustavou F_4 .

Pomocí GB s uspořádáním $\mathbf{plex}(t[7], t[6], t[5], t[4], t[3], t[2], t[1])$ lze vyjádřit explicitní vzorce délek jednotlivých časových intervalů.



Obrázek 3.29: Typický tvar trajektorie zrychlení pro soustavu F_7 .

Výstupní rovnice řešení soustavy F_7 použité v algoritmu $GBAVS$:

$$p0 = 3 a_0^4 - 3 a_f^4 + 12 BM (a_0^2 v_0 + a_f^2 v_f) + 12 BM^2 (v_0^2 - v_f^2) + 6 DM^2 (a_0^2 - a_f^2 + 2 BM (v_0 + v_f)) + 8 DM (a_0^3 - a_f^3 + 3 BM^2 (s_0 - s_f) + 3 BM (a_0 v_0 + a_f v_f))$$

$$p1 = 48 BM (DM + a_0) \left(\frac{DM a_0}{2} + \frac{a_0^2}{2} + BM v_0 \right)$$

$$p2 = 12 BM^2 (DM^2 + 6 DM a_0 + 5 a_0^2 + 2 BM v_0)$$

$$p3 = 24 BM^3 (DM + 2 a_0)$$

$$p4 = 12 BM^4$$

$$s7t1 = \text{roots}[p0 + p1 s7t1 + p2 s7t1^2 + p3 s7t1^3 + p4 s7t1^4 = 0, s7t1]$$

$$s7t2 = 0$$

$$s7t3 = \frac{DM}{BM} + s7t1 + \frac{a_0}{BM}$$

$$s7t4 = 0$$

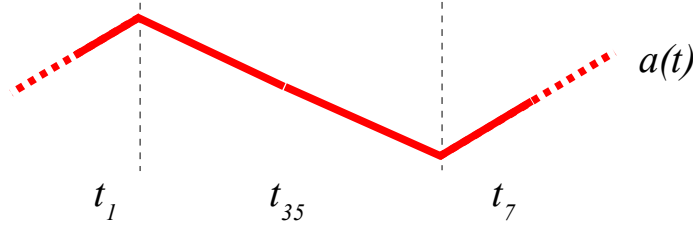
$$s7t5 = 0$$

$$s7t6 = \frac{-2 DM^2 + 2 BM^2 s7t1^2 + a_0^2 + a_f^2 + 2 BM (2 s7t1 a_0 + v_0 - v_f)}{2 BM DM}$$

$$s7t7 = s1t7$$

ŘEŠENÍ SOUSTAVY F_8

Soustava $F_8 = \{eq_1, eq_2, eq_3, eq_{4b1}, eq_{4b2}, eq_{5b}, eq_{6b}\}$ nerespektuje žádné omezení a odpovídá situaci, kdy systém *žádného* omezení nedosáhne. Pokud je řešením soustavy 7 nezáporných reálných časových intervalů, potom je nutné otestovat omezení AM, DM a VM stejně jako v soustavách F_5, F_2 a F_3 . Poznamenejme, že tato soustava odpovídá obecnému řešení systému tří integrátorů bez stavových omezení, *bang-bang* řízení.



Obrázek 3.30: Typický tvar trajektorie zrychlení pro soustavu F_8 .

Pomocí Gröbnerovy báze s uspořádáním $\mathbf{plex}(t[1], t[2], t[6], t[7], t[3], t[5], t[4])$ můžeme nalézt explicitní vzorce délek jednotlivých časových intervalů. GB vyjde v následujícím tvaru.

$$GB = \{t_4, t_5, t_2, t_6, -3 a_0^4 + 6 a_0^2 a_f^2 - 3 a_f^4 + 16 BM a_0^3 t_{35} - 16 BM a_f^3 t_{35} + 48 BM^3 s_0 t_{35} - 48 BM^3 s_f t_{35} - 24 BM^2 a_0^2 t_{35}^2 - 24 BM^2 a_f^2 t_{35}^2 + 12 BM^4 t_{35}^4 + 12 BM a_0^2 v_0 - 12 BM a_f^2 v_0 - 48 BM^2 a_0 t_{35} v_0 + 48 BM^3 t_{35}^2 v_0 - 12 BM^2 v_0^2 - 12 BM a_0^2 v_f + 12 BM a_f^2 v_f + 48 BM^2 a_f t_{35} v_f + 48 BM^3 t_{35}^2 v_f + 24 BM^2 v_0 v_f - 12 BM^2 v_f^2, 8 a_0^3 - 6 a_0^2 a_f - 2 a_f^3 + 24 BM^2 s_0 - 24 BM^2 s_f + 6 BM a_0^2 t_7 - 6 BM a_f^2 t_7 - 15 BM a_0^2 t_{35} - 9 BM a_f^2 t_{35} + 6 BM^3 t_{35}^3 - 24 BM a_0 v_0 + 12 BM a_f v_0 - 12 BM^2 t_7 v_0 + 30 BM^2 t_{35} v_0 + 12 BM a_f v_f + 12 BM^2 t_7 v_f + 18 BM^2 t_{35} v_f, -2 a_0^3 - 6 a_0 a_f^2 + 8 a_f^3 - 24 BM^2 s_0 + 24 BM^2 s_f + 6 BM a_0^2 t_1 - 6 BM a_f^2 t_1 + 9 BM a_0^2 t_{35} + 15 BM a_f^2 t_{35} - 6 BM^3 t_{35}^3 + 12 BM a_0 v_0 - 12 BM^2 t_1 v_0 - 18 BM^2 t_{35} v_0 + 12 BM a_0 v_f - 24 BM a_f v_f + 12 BM^2 t_1 v_f - 30 BM^2 t_{35} v_f\}$$

Podrobnějším zkoumáním bychom zjistili, že rovnice pro délky časových intervalů t_1 a t_7 jsou si velmi „podobné“, což se dá očekávat ze symetrie řízení, viz také obrázek trajektorie zrychlení 3.30. Můžeme zkusit přímo řešit získané bazové rovnice. Bohužel ale v řešení pro t_1 a t_7 dochází k dělení vstupními parametry, které mohou nabývat nulových hodnot, viz (3.46). Dochází k již dříve zmíněné specializaci řešení. Mohli bychom přidat jmenovatel do generátorů ideálu a spočítat GB znovu. Získané řešení by platilo ve speciálních případech. Jednalo by se o singulární řešení, které je z výpočetního hlediska nevhodné. V průběhu testování algoritmu se ukázalo, že toto řešení nelze v praxi použít v důsledku numerické nestability.

$$s8t1 = -\frac{2 a_0^3 + 6 a_0 a_f^2 - 8 a_f^3 + 24 BM^2 s_0 - 24 BM^2 s_f - 9 BM a_0^2 t_{35} - 15 BM a_f^2 t_{35} + 6 BM^3 t_{35}^3 - 12 BM a_0 v_0 + 18 BM^2 t_{35} v_0 - 12 BM a_0 v_f + 24 BM a_f v_f + 30 BM^2 t_{35} v_f}{6 BM (-a_0^2 + a_f^2 + 2 BM v_0 - 2 BM v_f)}$$

$$s8t2 = 0$$

$$s8t35 = \text{roots} \left[-3 a_0^4 + 6 a_0^2 a_f^2 - 3 a_f^4 + 12 BM^4 t_{35}^4 + 12 BM a_0^2 v_0 - 12 BM a_f^2 v_0 - 12 BM^2 v_0^2 - 12 BM a_0^2 v_f + 12 BM a_f^2 v_f + 24 BM^2 v_0 v_f - 12 BM^2 v_f^2 + t_{35}^2 (-24 BM^2 a_0^2 - 24 BM^2 a_f^2 + 48 BM^3 v_0 + 48 BM^3 v_f) + t_{35} (16 BM a_0^3 - 16 BM a_f^3 + 48 BM^3 s_0 - 48 BM^3 s_f - 48 BM^2 a_0 v_0 + 48 BM^2 a_f v_f), 5 t_3 \right]$$

$$s8t4 = 0$$

$$s8t5 = 0$$

$$s8t6 = 0$$

$$s8t7 = \frac{8 a_0^3 - 6 a_0^2 a_f - 2 a_f^3 + 24 BM^2 s_0 - 24 BM^2 s_f - 15 BM a_0^2 t_{35} - 9 BM a_f^2 t_{35} + 6 BM^3 t_{35}^3 - 24 BM a_0 v_0 + 12 BM a_f v_0 + 30 BM^2 t_{35} v_0 + 12 BM a_f v_f + 18 BM^2 t_{35} v_f}{6 BM (-a_0^2 + a_f^2 + 2 BM v_0 - 2 BM v_f)}$$

(3.46)

Proto je vhodné zavést nové parametry maximálního/minimálního dosaženého zrychlení am, dm , viz kapitola 3.3.3, a specifikovat délky časových intervalů ve tvaru

$$\left\{ t_1 = \frac{am - a_0}{BM}, t_2 = 0, t_{35} = \frac{am + dm}{BM}, t_4 = 0, t_5 = 0, t_6 = 0, t_7 = \frac{dm + a_f}{BM} \right\} \quad (3.47)$$

Tyto vztahy dosadit zpět do GB , a hledat řešení tentokrát pro proměnné am, dm . Tímto postupem dostaneme explicitní vztahy pro parametry am, dm , bez specializace, a tyto vztahy spolu s (3.47) tvoří úplné řešení osmé soustavy, viz dále. Testování omezení AM a DM lze v tomto případě provést přímo porovnáním s dosaženými hodnotami am, dm .

Výstupní rovnice řešení soustavy F_8 použité v algoritmu $GBAVS$:

$$p0 = -a_0^6 - 9 a_0^4 a_f^4 - 16 a_0^3 a_f^3 + 9 a_0^2 a_f^4 + 17 a_f^6 + 48 BM^2 a_0^3 s_0 - 48 BM^2 a_f^3 s_0 + 72 BM^4 s_0^2 - 48 BM^2 a_0^3 s_f + 48 BM^2 a_f^3 s_f - 144 BM^4 s_0 s_f + 72 BM^4 s_f^2 + 6 BM a_0^4 v_0 + 36 BM a_0^2 a_f^2 v_0 + 48 BM a_0 a_f^3 v_0 - 18 BM a_f^4 v_0 - 144 BM^3 a_0 s_0 v_0 + 144 BM^3 a_0 s_f v_0 - 36 BM^2 a_0^2 v_0^2 - 36 BM^2 a_f^2 v_0^2 + 72 BM^3 v_0^3 + 18 BM a_0^4 v_f + 48 BM a_0^3 a_f v_f - 36 BM a_0^2 a_f^2 v_f - 102 BM a_f^4 v_f + 144 BM^3 a_f s_0 v_f - 144 BM^3 a_f s_f v_f - 72 BM^2 a_0^2 v_0 v_f - 144 BM^2 a_0 a_f v_0 v_f + 72 BM^2 a_f^2 v_0 v_f + 72 BM^3 v_0^2 v_f + 36 BM^2 a_0^2 v_f^2 + 180 BM^2 a_f^2 v_f^2 - 72 BM^3 v_0 v_f^2 - 72 BM^3 v_f^3$$

$$p1 = -48 a_0^3 a_f^2 + 48 a_f^5 - 144 BM^2 a_f^2 s_0 + 144 BM^2 a_f^2 s_f + 144 BM a_0 a_f^2 v_0 + 96 BM a_0^3 v_f - 240 BM a_f^3 v_f + 288 BM^3 s_0 v_f - 288 BM^3 s_f v_f - 288 BM^2 a_0 v_0 v_f + 288 BM^2 a_f v_f^2$$

$$p2 = 18 a_0^4 - 36 a_0^2 a_f^2 + 18 a_f^4 - 72 BM a_0^2 v_0 + 72 BM a_f^2 v_0 + 72 BM^2 v_0^2 + 72 BM a_0^2 v_f - 72 BM a_f^2 v_f - 144 BM^2 v_0 v_f + 72 BM^2 v_f^2$$

$$p3 = 48 a_0^3 - 48 a_f^3 + 144 BM^2 s_0 - 144 BM^2 s_f - 144 BM a_0 v_0 + 144 BM a_f v_f$$

$$p4 = 36 a_0^2 - 36 a_f^2 - 72 BM v_0 + 72 BM v_f$$

$$dm = \text{roots}[p0 + dm p1 + dm^2 p2 + dm^3 p3 + dm^4 p4 == 0, dm]$$

$$\begin{aligned} am = & \left(dm a_0^6 - 17 dm a_f^6 - 12 a_f^7 - 12 a_0^5 (-dm^2 + a_f^2 - 2 BM v_f) + 12 a_f^5 (dm^2 - 2 BM v_0 + 7 BM v_f) + 3 dm a_0^4 (6 dm^2 - 9 a_f^2 - 2 BM v_0 + 18 BM v_f) + 6 a_f^4 (3 dm^3 + 6 BM^2 (s_0 - s_f) + BM dm (-9 v_0 + 17 v_f)) + \right. \\ & 72 BM^2 a_f v_f (dm^2 (-v_0 + v_f) - 2 BM (dm (s_0 - s_f) + (v_0 - v_f) v_f)) - 72 BM^2 (BM dm (dm (s_0 - s_f) + (v_0 - v_f) v_f)^2 (v_0 - v_f) - dm^3 (v_0 - v_f)^2 + BM^2 (s_0 - s_f) (dm (s_0 - s_f) + 2 (v_0 - v_f) v_f)) + \\ & 12 BM a_f^3 (dm^2 (2 v_0 - 5 v_f) + 2 BM (2 dm s_0 - 2 dm s_f + 5 v_0 v_f - 8 v_f^2)) + 36 BM a_f^2 (2 BM^2 (s_0 - s_f) (v_0 - 2 v_f) + 2 dm^3 (v_0 - v_f) + BM dm (-dm s_0 + dm s_f - 3 v_0^2 + 6 v_0 v_f - 5 v_f^2)) - \\ & 12 BM a_0 v_0 (4 dm a_f^3 + 3 a_f^4 - 12 BM dm a_f v_f - 3 a_f^2 (dm^2 - 2 BM v_0 + 4 BM v_f) - 6 BM (dm^2 (v_0 - v_f) + 2 BM (dm (s_0 - s_f) + (v_0 - v_f) v_f))) + \\ & 4 a_0^3 (4 dm a_f^3 + 3 a_f^4 - 12 BM dm a_f v_f - 3 a_f^2 (dm^2 - 5 BM v_0 + 4 BM v_f) - 3 BM (dm^2 (5 v_0 - 2 v_f) + 2 BM (2 dm s_0 - 2 dm s_f + 5 v_0 v_f - 2 v_f^2))) + 3 a_0^2 \\ & \left. (9 dm a_f^4 + 4 a_f^5 + 12 BM a_f v_f (dm^2 + 2 BM v_f) - 4 a_f^3 (dm^2 + 5 BM v_f) - 12 a_f^2 (dm^3 + BM^2 (s_0 - s_f) + 3 BM dm (-v_0 + v_f)) + 12 BM (2 BM^2 (s_0 - s_f) v_f + 2 dm^3 (-v_0 + v_f) + BM dm (dm s_0 - dm s_f + v_0^2 - 6 v_0 v_f + 3 v_f^2))) \right) / \\ & (a_0^6 + a_f^6 + 48 BM^2 a_f^3 (s_0 - s_f) - 3 a_0^4 (3 a_f^2 + 2 BM (v_0 - 3 v_f)) + 6 BM a_f^4 (3 v_0 - v_f) - 144 BM^3 a_f (s_0 - s_f) v_f + 16 a_0^3 (a_f^3 + 3 BM^2 (-s_0 + s_f) - 3 BM a_f v_f) - \\ & 48 BM a_0 v_0 (a_f^3 + 3 BM^2 (-s_0 + s_f) - 3 BM a_f v_f) - 36 BM^2 a_f^2 (v_0^2 + 2 v_0 v_f - v_f^2) - 72 BM^3 (BM (s_0 - s_f)^2 + (v_0 - v_f)^2 (v_0 + v_f)) - 9 a_0^2 (a_f^4 - 4 BM a_f^2 (v_0 + v_f) + 4 BM^2 (-v_0^2 + 2 v_0 v_f + v_f^2))) \end{aligned}$$

$$s8t1 = \frac{am - a_0}{BM}$$

$$s8t2 = 0$$

$$s8t35 = \frac{am + dm}{BM}$$

$$s8t4 = 0$$

$$s8t5 = 0$$

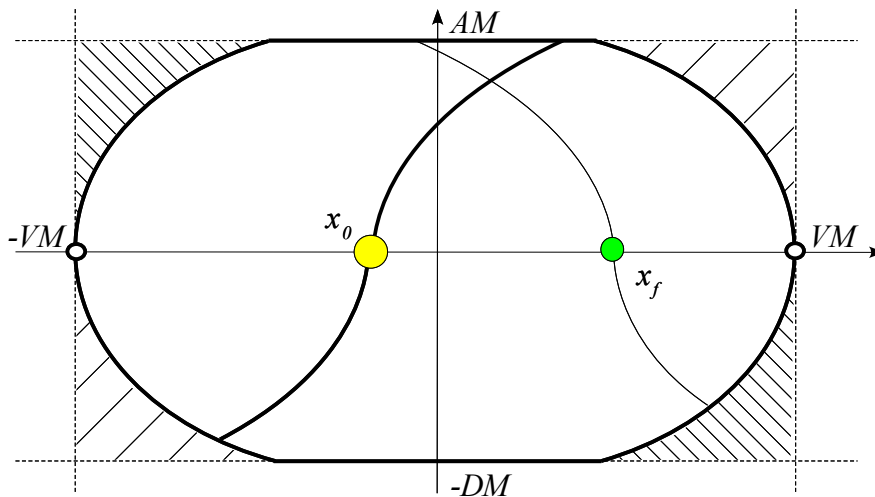
$$s8t6 = 0$$

$$s8t7 = \frac{dm + a_f}{BM}$$

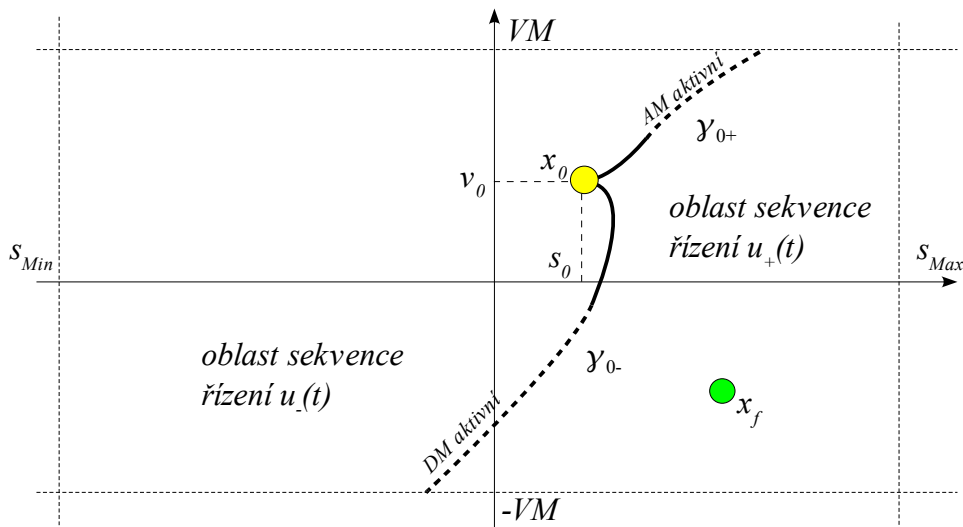
3.5.3 POHYB Z KONSTANTNÍ RYCHLOSTI DO KONSTANTNÍ RYCHLOSTI

Tento druh přesunu může být nejčastěji využívaným pohybem při plánování a napojování jednotlivých pohybů především u rotačních strojů nebo například pásových dopravníků, kde je kladen důraz na rychlost pohybu. Z pohledu dynamiky je konstantní rychlost systému takzvaným relativním rovnovážným stavem a jedná se o speciální případ obecného pohybu, kdy počáteční x_0 a koncový stav x_f leží ve stavovém podprostoru s - v , $x_0, x_f \in \mathbb{R}^2 \subset \mathbb{R}^3$. Projekcí stavového prostoru do roviny v - a se x_0 a x_f zobrazí pouze na horizontální ose, obrázek 3.31. V tomto případě lze i pro generátor 3.řádu s omezeními nalézt *rozdělující varietu*, která dělí stavový podprostor počátečních a koncových stavů na dvě souvislé oblasti tak, že lze apriori rozhodnout, která sekvence řízení generuje t -optimální přesun systému. To umožní hledat řešení pouze v osmi soustavách z původních šestnácti, a tedy dvojnásobně zkrátit čas výpočtu.

Dále je uveden původní explicitní popis rozdělující variety a příslušný rozhodovací algoritmus.



Obrázek 3.31: Stavový podprostor v - a pro řetězec tří integrátorů s omezeními, s vyznačenými rozjezdovými a dojezdovými křivkami. Při pohybu z/do konstantní rychlosti leží počáteční a koncový stav na horizontální ose.



Obrázek 3.32: Geometrie stavového podprostoru s - v . Varieta γ_0 (tučně) rozděluje stavový podprostor na dvě souvislé oblasti, ve kterých je jednoznačně dáno, jaké řízení povede systém ke koncovému stavu.

EXISTENCE A NALEZENÍ ROZDĚLUJÍCÍ VARIETY

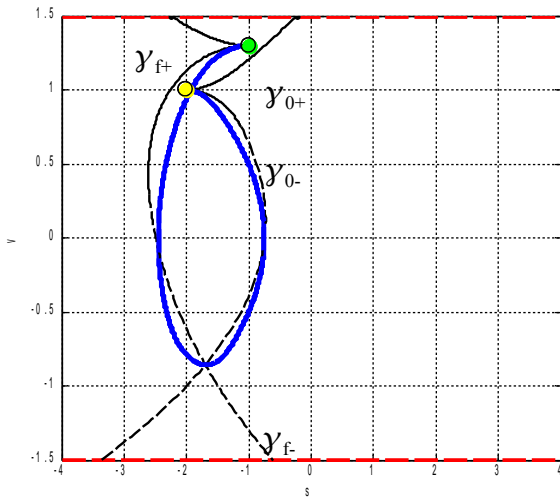
Předpokládáme pohyb z *konstantní rychlosti do konstantní rychlosti*. Dále předpokládáme, že platí hypotéza 3.4, a přípustné t-optimální řízení je obsaženo v jedné ze sekvencí $u_+(t)$ nebo $u_-(t)$. Potom lze najít rozdělující křivky a sestavit následující souhrn vlastností. Naznačení důkazu je provedeno dále.

1. $\mathbf{x}_0 = [s_0, v_0, 0]$, $\mathbf{x}_f = [s_f, v_f, 0]$ leží v rovině s - v , $\mathbf{x}_0, \mathbf{x}_f \in \mathbb{R}^2 \subset \mathbb{R}^3$.
2. Ze strategie řízení vyplývá, že trajektorie systému může protnout rovinu s - v pouze jednou v čase τ_3 , případně na této rovině setrvat po dobu délky časového intervalu t_4 . Dále musí platit rovnosti $t_3 = t_1$, $t_7 = t_5$. To zajistí návrat stavu systému do roviny s - v . Ze sedmi časových intervalů dostáváme tedy pouze pět různých. Hledáme posloupnost délek intervalů

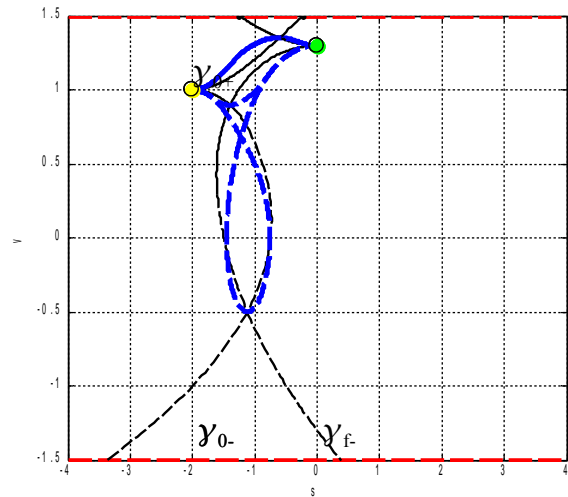
$$[t_1, t_{2(AM \text{ omezení})}, t_1, t_{4(VM \text{ omezení})}, t_5, t_{6(DM \text{ omezení})}, t_5] \quad (3.48)$$
3. Lze najít všechny body průtnutí roviny s - v , které by bylo možné dosáhnout přípustným řízením $u_+(t)$ a $u_-(t)$, $t \in [0, \tau_3]$, až do maximální (minimální) rychlosti VM ($-VM$). Tyto body tvoří rozdělující křivku γ_0 : $\gamma_0 = \gamma_{0+} \cup \gamma_{0-}$, γ_{0+} , pro sekvenci $u_+(t)$, γ_{0-} , pro sekvenci $u_-(t)$, $t \in [0, \tau_3]$, která leží v rovině s - v a vychází z počátečního stavu, viz obrázek 3.32.
4. Podobně lze sestavit rozdělující křivku γ_f pro koncový stav s uvažováním záporného času γ_f : $\gamma_f = \gamma_{f+} \cup \gamma_{f-}$, γ_{f+} , pro strategii $u_+(t)$, γ_{f-} , pro strategii $u_-(t)$, $t \in [0, -2t_5 - t_6]$.
5. Pro sekvenci řízení $u_+(t)$ mohou mít γ_{0+} a γ_{f+} žádný, jeden nebo dva průsečíky $\bar{\Gamma}_i, i=0, 1, 2$, nebo x_f leží přímo na γ_{0+} . Totéž platí pro sekvenci řízení $u_-(t)$ a γ_{0-} a γ_{f-} . Průsečík znamená existenci řešení pro danou sekvenci řízení.
6. Ze strategie řízení také vyplývá, že křivky $\gamma_{0+}, \gamma_{f+}, \gamma_{0-}, \gamma_{f-}$ jsou jednoznačné funkce vzhledem k $v(t)$. Projekce křivky stavu $\Gamma(t)$ do roviny s - v , musí být z pohledu rychlosti $v(t)$, $t \in [0, \tau_3]$ monotónní funkcí vzhledem k parametru t , neboť systém pro $u_+(t)$ stále zrychluje (pro $u_-(t)$ stále zpomaluje).
7. Protože křivky γ leží v rovině s - v , průsečík také leží v rovině s - v a je to zároveň bod nejvyšší dosažené rychlosti pro strategii $u_+(t)$ (nejnižší pro strategii $u_-(t)$). Jestliže je průsečík nad (pod) hodnotou VM ($-VM$), potom je průsečíkem chápána mezní přímka omezení VM ($-VM$).
8. Z rovnic popisující sekvence řízení vyplývá, že podle vzájemné polohy $\mathbf{x}_0, \mathbf{x}_f \in \mathbb{R}^2 \subset \mathbb{R}^3$ mohou existovat maximálně tři průsečíky $\bar{\Gamma}_i, i=0, 1, 2, 3$, a to tak, že jestliže existují dva v jedné strategii řízení, potom v druhé už může existovat maximálně jeden. Úloha může mít tedy maximálně tři řešení $\Gamma_i(t), i=0, 1, 2, 3$, viz naznačení důkazu dále.
9. Protože tvar řízení je až na délku časových intervalů specifikován, a platí (3.48), potom v důsledku symetrie řídicí sekvence musí platit, že projektivní trajektorie $\Gamma_i(t)$, příslušné danému průsečíku $\bar{\Gamma}_i$, se vzájemně nekříží. Potom lze z těchto trajektorií vybrat t-optimální trajektorii systému, pro kterou v rovině s - v platí

$$\min_{u(\cdot)} \int_0^{t_f} 1 dt \Rightarrow \max_{u(\cdot)} \int_{s_0}^{s_f} v ds \quad (3.49)$$

10. Protože strategie řízení je sestavena tak, že maximalizuje rychlost maximálním možným zrychlením, je zřejmé, že t-optimální trajektorie systému je taková křivka, která prochází průsečíkem $\bar{\Gamma}_i$ s nejvyšší absolutní hodnotou rychlosti.
11. Využitím výše uvedených vlastností rozdělovacích křivek lze ukázat, že γ_{0+} a γ_{f+} se mohou křížit pouze pokud je stav x_f napravo od křivky γ_0 . V opačném případě ke křížení nedojde a řešení ve strategii $u_+(t)$ neexistuje. Také platí, že pokud dojde pouze k jednomu křížení, potom je trajektorie systému procházející tímto bodem t-optimální trajektorií (rozdělovací křivky s jedním průsečíkem jsou vždy na nejvyšší hladině rychlosti).

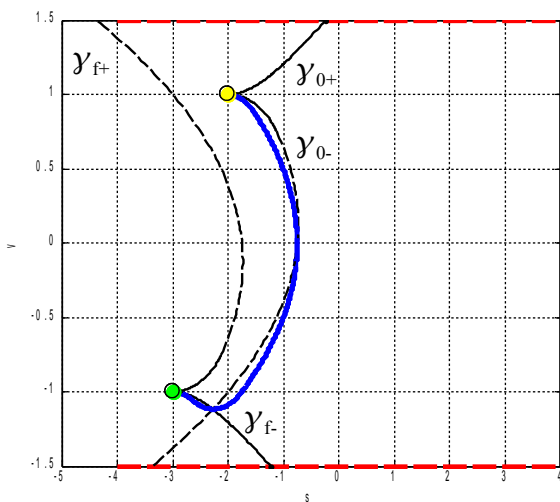


Stav x_f je vlevo od rozdělovací variety $\Rightarrow u_-(t)$

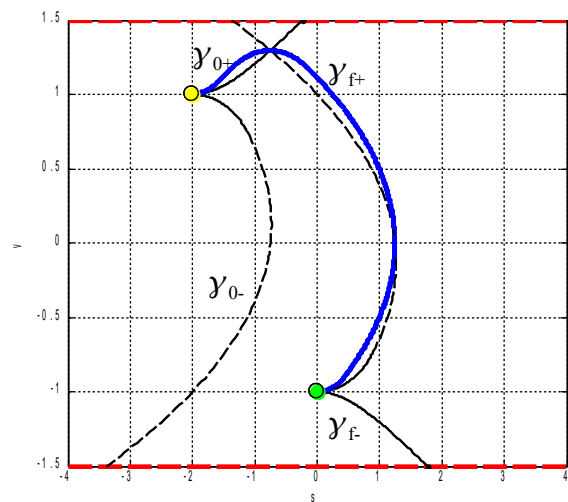


Stav x_f je vpravo od rozdělovací variety $\Rightarrow u_+(t)$.
Dochází ještě ke křížení γ_0 a γ_{f-} , ale ani jedno řešení $u_-(t)$ není časově optimální.

Obrázek 3.33: Rozdělovací křivky a jejich možnosti křížení pro různé polohy x_0 a x_f . Modré křivky jsou projektivní trajektorie systému $\Gamma_i(t)$. V případě, že dojde ke křížení obou rozdělovacích křivek, potom existují tři řešení.



Stav x_f je vlevo od rozdělovací variety $\Rightarrow u_-(t)$



Stav x_f je vpravo od rozdělovací variety $\Rightarrow u_+(t)$

Obrázek 3.34: Rozdělovací křivky a jejich možnosti křížení pro některé polohy x_0 a x_f . Modré křivky jsou projekce t-optimální trajektorie systému do roviny s-v.

ROZHODOVACÍ ALGORITMUS

Algoritmus definuje t-optimální řídicí sekvenci na základě vzájemné polohy rozdělující variety γ_0 a bodu x_f . Jestliže je bod x_f vpravo od γ_0 , potom t-optimální řídicí sekvence je $u_+(t)$, v opačném případě je to sekvence $u_-(t)$. Nalezení rozdělující variety γ_0 vyplývá explicitně z výše uvedených předpokladů a vlastností. Rozdělující křivka γ_{0+} pro sekvenci $u_+(t)$ začíná v bodě x_0 a končí na hranici omezení VM . Rozdělující křivka γ_{0-} pro sekvenci $u_-(t)$ začíná v bodě x_0 a končí na hranici omezení $-VM$. Výsledná rozdělující varieta γ_0 je pak složena z obou rozdělujících křivek, viz obrázek 3.32. Odvození provedeme pro její část γ_{0+} . Pro γ_{0-} je odvození ekvivalentní. Zavedeme dva parametry $t_a \in [0, t_1]$, $t_b \in [0, t_2]$, které budou parametrizovat rozdělující křivku γ_{0+} . První parametr t_a parametrizuje rozdělující křivku před dosažením omezení AM , druhý pak parametrizuje druhou část křivky v případě, že bylo omezení AM dosaženo. Podle bodu 3 vlastností platí, že všechny body protnutí, tvořící křivku γ_{0+} , dostaneme řízením

$$u_+(t) = \begin{cases} BM, & t \in [0, \tau_1) \\ 0, & t \in [\tau_1, \tau_2) \\ -BM, & t \in [\tau_2, \tau_3) \end{cases} \quad (3.50)$$

kde $\tau_1 = t_a$, $\tau_2 = t_a + t_b$, $\tau_3 = 2t_a + t_b$, $t_a \in [0, t_1]$, $t_b \in [0, t_2]$, a t_1 a t_2 jsou určeny tak, že musí platit $v(\tau_3) = VM$, $\tau_3 = 2t_1 + t_2$.

Vztahy definující γ_{0+} získáme z rovnic systému (3.34) a příslušné sekvence řízení (3.50). Uvažujme její první část (obrázek 3.32, plný úsek křivky γ_{0+}), kde nebylo dosaženo AM omezení. Potom platí $\tau_1 = \tau_2$ a dosazením (3.50) do (3.34) dostaneme

$$a(\tau_1) = BM t_a, \quad v(\tau_1) = v_0 + \frac{BM}{2} t_a^2, \quad s(\tau_1) = s_0 + v_0 t_a + \frac{BM}{6} t_a^3$$

a tedy

$$\begin{aligned} a(\tau_3) &= a(\tau_1) - BM t_a = 0 \\ v(\tau_3) &= v(\tau_1) + a(\tau_1) t_a - \frac{BM}{2} t_a^2 = v_0 + BM t_a^2 \\ s(\tau_3) &= s(\tau_1) + v(\tau_1) t_a + \frac{a(\tau_1)}{2} t_a^2 - \frac{BM}{6} t_a^3 = s_0 + 2v_0 t_a + BM t_a^3. \end{aligned} \quad (3.51)$$

Vyloučením parametru t_a dostaneme implicitní popis první části rozdělující křivky γ_{0+}

$$s = s_0 + \sqrt{\frac{v - v_0}{BM}} (v_0 + v), \quad v \in [v_0, VM] \quad (3.52)$$

Ze vztahu (3.51) lze dosazením VM za $v(\tau_3)$ získat vztah pro velikost t_1

$$t_1 = \sqrt{\frac{VM - v_0}{BM}},$$

definující rozsah parametrizace pomocí t_a .

Jestliže platí $t_1 < AM/BM$, potom omezení AM nebude dosaženo a γ_{0+} je parametrizována pouze parametrem t_a , dána vztahy (3.51), nebo implicitním popisem (3.52). V opačném případě je třeba po dosažení rovnosti $t_a = AM/BM$ parametrizovat rozdělující křivku parametrem t_b . Odvození opět vyplývá z (3.50) a (3.34). Pro tento případ platí

$$\begin{aligned}
a(\tau_1) &= BM t_a = AM \\
v(\tau_1) &= v_0 + \frac{BM}{2} t_a^2 = v_0 + \frac{AM^2}{2 BM} & a(\tau_2) &= a(\tau_1) = AM \\
s(\tau_1) &= s_0 + v_0 t_a + \frac{BM}{6} t_a^3 = s_0 + v_0 \frac{AM}{BM} + \frac{AM^3}{6 BM^2} & v(\tau_2) &= v(\tau_1) + AM t_b \\
a(\tau_3) &= a(\tau_2) - BM t_a = 0 & s(\tau_2) &= s(\tau_1) + v(\tau_1) t_b + \frac{AM}{2} t_b^2 \\
v(\tau_3) &= v(\tau_2) + a(\tau_2) t_a - \frac{BM}{2} t_a^2 = v_0 + AM t_b + \frac{AM^2}{BM} \\
s(\tau_3) &= s(\tau_2) + v(\tau_2) t_a + \frac{a(\tau_2)}{2} t_a^2 - \frac{BM}{6} t_a^3 = s_0 + \frac{AM^3}{BM^2} + v_0 \left(t_b + \frac{2 AM}{BM} \right) + \frac{3 AM^2}{2 BM} t_b + \frac{AM}{2} t_b^2
\end{aligned} \tag{3.53}$$

Vyloučením parametru t_b dostaneme implicitní popis druhé části rozdělovací křivky (viz obrázek 3.32, čárkovaný úsek křivky y_{0+})

$$s = s_0 + \frac{AM^2(v_0 + v) + BM(v^2 - v_0^2)}{2 AM BM}, \quad v \in [v_0 + AM^2 / 2 BM, VM] \tag{3.54}$$

Ze vztahu (3.53) lze dosazením VM za $v(\tau_3)$ získat vztah pro velikost t_2

$$t_1 = AM / BM, \quad t_2 = \frac{VM - v_0 - AM^2 / BM}{AM} \tag{3.55}$$

definující rozsah parametrizace pomocí t_b .

Rozhodovací algoritmus pak využívá vztahy (3.52) a (3.54) pro určení, zda koncový stav leží vpravo nebo vlevo od rozdělovací křivky.

Věta 3.5. (Algoritmus). *Nechť jsou dána vstupní data pro spuštění algoritmu GBAVS. Jestliže počáteční a koncový stav leží v podprostoru s - v , potom lze apriori jednoznačně určit, která sekvence řízení obsahuje t -optimální řešení. Rozhodovací proces je definován následujícím algoritmem:*

vstup: $\mathbf{x}_0, \mathbf{x}_f$

výstup: sekvence řízení $u_{\pm}(t)$

parametry: BM, AM, DM, VM

$$\begin{aligned}
& \text{IF } v_f - v_0 > 0 \\
& \quad \text{THEN } B_M = BM, A_M = AM \\
& \quad \text{ELSE } B_M = -BM, A_M = -DM \\
& \text{END} \\
& t_a = \sqrt{\frac{v_f - v_0}{B_M}} \\
& \text{IF } t_a < A_M / B_M \\
& \quad \text{THEN } s_v = s_0 + \sqrt{\frac{v_f - v_0}{B_M}} (v_0 + v_f) \\
& \quad \text{ELSE } s_v = s_0 + \frac{A_M^2 (v_0 + v_f) + B_M (v_f^2 - v_0^2)}{2 A_M B_M} \\
& \text{END} \\
& u_{\pm}(t) = \text{sign}(s_f - s_v) = \begin{cases} 1: & u_+(t) \\ -1: & u_-(t) \end{cases}
\end{aligned}$$

NAZNAČENÍ DŮKAZU EXISTENCE ROZDĚLUJÍCÍ VARIETY

Úplný důkaz existence rozdělující variety vyžaduje vyšetřit všechny kombinace vzhledem k uvažovaným omezením, která systém může dosáhnout či nikoli. Zápis důkazu by byl velmi dlouhý a proto pouze naznačíme, jakým způsobem je možné ho provést, včetně klíčových kroků, které dokazují výše zmíněná tvrzení v bodech vlastností. Využijeme implicitní vztahy (3.52) a (3.54).

Nejprve uvažujme pohyb, při kterém nebudou dosaženy limity zrychlení. Potom pro γ_0 křivku platí implicitní popis

$$s = s_0 + \sqrt{\frac{v - v_0}{u}}(v_0 + v), \quad u = \begin{cases} BM & \text{pro } v_0 \leq v \leq VM \\ -BM & \text{pro } v_0 \geq v \geq -VM \end{cases} \quad (3.56)$$

Pro γ_f křivku bychom dostali implicitní popis ve tvaru

$$s = s_f - \sqrt{\frac{v - v_f}{u}}(v_f + v), \quad u = \begin{cases} BM & \text{pro } v_f \leq v \leq VM \\ -BM & \text{pro } v_f \geq v \geq -VM \end{cases} \quad (3.57)$$

Vyšetříme nejprve průběh křivky γ_{0+} . Úlohu rozdělíme na případy podle znaménka rychlosti v_0 .

Pro $v_0 > 0$ můžeme rovnici (3.56) zapsat a vyšetřovat ve tvaru

$$s = f_{v_0}^+(v), \quad v > v_0 > 0, \quad u = BM$$

$$f_{v_0}^+{}' = \frac{3v - v_0}{2BM \sqrt{\frac{v - v_0}{BM}}} \Rightarrow \text{extrem funkce v bodě } v = v_0/3 \quad (3.58)$$

Tato funkce je rostoucí na celém svém rozsahu, protože extrém nastává v bodě $v_0/3$ což je pro $v_0 > 0$ pod hodnotou v_0 .

Pro $v_0 < 0$ dostaneme funkci $s = f_{v_0}^+(v)$, $v > v_0 < 0$, $u = BM$, opět s extrémem v bodě $v_0/3$.

Vyšetřením průběhu dostaneme

$$f_{v_0}^+(v) = \begin{cases} \text{klesající pro } v_0 < v < v_0/3 \\ \text{rostoucí pro } v_0/3 < v \leq VM \end{cases} \quad (3.59)$$

Pro γ_{f+} bude situace podobná. Úlohu rozdělíme na případy podle znaménka rychlosti v_f .

Pro $v_f > 0$ dostaneme

$$s = f_{v_f}^+(v), \quad v > v_f > 0, \quad u = BM$$

$$f_{v_f}^+{}' = -\frac{3v - v_f}{2BM \sqrt{\frac{v - v_f}{BM}}} \Rightarrow \text{extrem funkce v bodě } v = v_f/3 \quad (3.60)$$

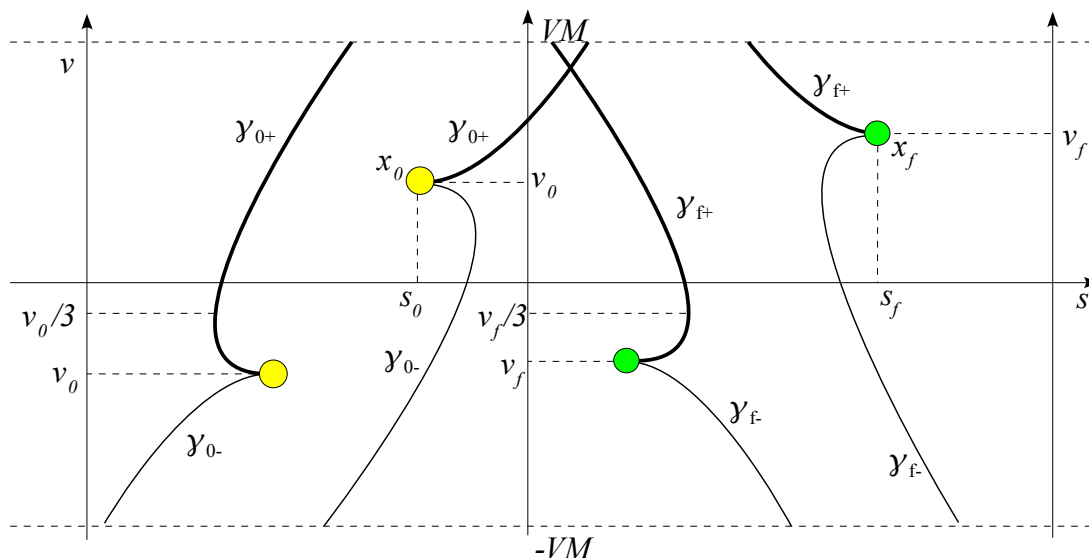
$$f_{v_f}^+(v) = \begin{cases} \text{rostoucí pro } v_f < v \leq VM \end{cases}$$

Pro $v_f < 0$ dostaneme $s = f_{v_f}^+(v)$, $v > v_f < 0$, $u = BM$, opět s extrémem v bodě $v_f/3$,

$$f_{v_f}^+(v) = \begin{cases} \text{rostoucí pro } v_f < v < v_f/3 \\ \text{klesající pro } v_f/3 < v \leq VM \end{cases} \quad (3.61)$$

Výsledkem analýzy průběhů křivek γ_{0+} a γ_{f+} je, že pokud je počáteční i koncová rychlost záporná, potom může dojít k žádnému až dvěma průsečíkům těchto křivek. V ostatních případech

může dojít maximálně k jednomu průsečíku. Pro sekvenci řízení $u_-(t)$ bychom dostali podobné vlastnosti křivek γ_{0-} a γ_{f-} , viz obrázek 3.35. Vyšetřením jejich průběhů bychom zjistili, že pro kladnou počáteční a koncovou rychlost mohou nastat maximálně dva průsečíky. V ostatních případech maximálně jeden.



Obrázek 3.35: Možnosti poloh stavu x_0 a x_f a příslušných křivek γ pro případ, že omezení na zrychlení není dosaženo.

Jestliže je během pohybu dosaženo omezení na zrychlení, potom je třeba také vyšetřit vliv omezení na tvar γ_0 a γ_f . Pro γ_{0+} křivku bude platit

$$s = \begin{cases} f_{v_0}^{+1}(v): f_{v_0}^{+1}(v) = s_0 + \sqrt{\frac{v-v_0}{BM}}(v_0+v), & \text{pro } v \in \left[v_0, v_0 + \frac{AM^2}{2BM} \right] \\ f_{v_0}^{+2}(v): f_{v_0}^{+2}(v) = s_0 + \frac{AM^2(v_0+v) + BM(v^2 - v_0^2)}{2AMBM}, & \text{pro } v \in \left(v_0 + \frac{AM^2}{2BM}, VM \right] \end{cases} \quad (3.62)$$

Opět je třeba vyšetřit vliv znaménka v_0 na tvar funkce. Vyšetřením průběhu (3.62) pro $v_0 > 0$ dostaneme

$$f_{v_0}^{+1}(v) = \begin{cases} \text{rostoucí pro } v_0 < v < v_0 + \frac{AM^2}{2BM} \\ f_{v_0}^{+2}(v) = \begin{cases} \text{rostoucí pro } v_0 + \frac{AM^2}{2BM} < v \leq VM \end{cases} \end{cases} \quad (3.63)$$

Funkce $f_{v_0}^{+2}(v)$ je na svém rozsahu také stále rostoucí, protože její extrém leží v bodě $-\frac{AM^2}{2BM}$.

$$f_{v_0}^{+2}{}' = \frac{AM^2 + 2BMv}{2AMBM} \Rightarrow \text{extrem funkce v bodě } v = -\frac{AM^2}{2BM}. \quad (3.64)$$

Vyšetřením průběhu (3.62) pro $v_0 < 0$ dostaneme

$$f_{v_0}^{+1}(v) = \begin{cases} \text{klesající pro } v_0 < v < v_0/3 \\ \text{rostoucí pro } v_0/3 < v \end{cases} \quad (3.65)$$

$$f_{v_0}^{+2}(v) = \begin{cases} \text{klesající pro } v_0 + \frac{AM^2}{2BM} < v < -\frac{AM^2}{2BM} \\ \text{rostoucí pro } -\frac{AM^2}{2BM} < v \end{cases} \quad (3.66)$$

nicméně výsledná křivka γ_{0+} má pouze jeden extrém v bodě

$$v = \begin{cases} v_0/3, & \text{pro } v_0/3 < v_0 + \frac{AM^2}{2BM} \\ -\frac{AM^2}{2BM}, & \text{pro } v_0/3 > v_0 + \frac{AM^2}{2BM}. \end{cases} \quad (3.67)$$

Dva extrémy funkce nikdy nemá, protože pak by musely současně platit obě nerovnosti

$$v_0/3 < -\frac{AM^2}{2BM} \wedge v_0 + \frac{AM^2}{2BM} > v_0/3 \quad (3.68)$$

což zjevně neplatí.

Pro γ_{f+} křivku bude podobně platit

$$s = \begin{cases} f_{v_f}^{+1}(v): f_{v_f}^{+1}(v) = s_f - \sqrt{\frac{v-v_f}{BM}}(v_f+v), & \text{pro } v \in \left[v_f, v_f + \frac{DM^2}{2BM} \right] \\ f_{v_f}^{+2}(v): f_{v_f}^{+2}(v) = s_f - \frac{DM^2(v_f+v) + BM(v^2 - v_f^2)}{2DM BM}, & \text{pro } v \in \left(v_f + \frac{DM^2}{2BM}, VM \right] \end{cases} \quad (3.69)$$

a pro $v_f > 0$ je křivka (3.69) klesající na celém rozsahu. Pro $v_f < 0$ má křivka jeden extrém v bodě

$$v = \begin{cases} v_f/3, & \text{pro } v_f/3 < v_f + \frac{DM^2}{2BM} \\ -\frac{DM^2}{2BM}, & \text{pro } v_f/3 > v_f + \frac{DM^2}{2BM} \end{cases} \quad (3.70)$$

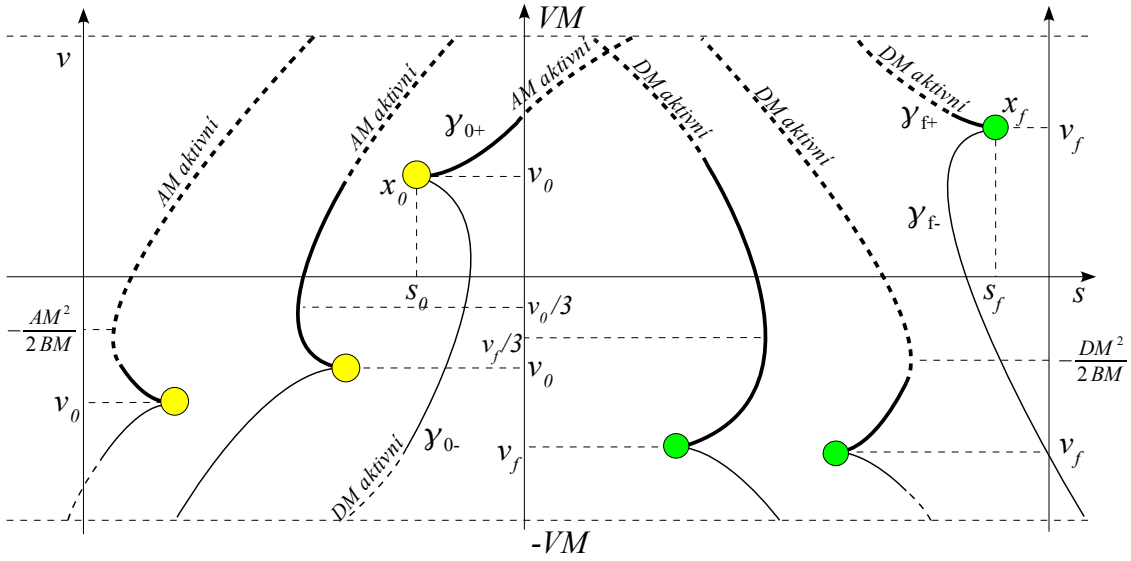
I v tomto případě je výsledkem analýzy průběhů křivek γ_{0+} a γ_{f+} , že pro počáteční i koncovou zápornou rychlost může dojít k žádnému až dvěma průsečíkům. V ostatních případech maximálně k jednomu. Pro strategii řízení $u_-(t)$ bychom dostali podobné vlastnosti křivek γ_{0-} a γ_{f-} , viz obrázek 3.36. Vyšetřením jejich průběhů bychom zjistili, že pro kladnou počáteční a koncovou rychlost mohou nastat maximálně dva průsečíky. V ostatních případech maximálně jeden.

Sestrojením tabulky vlastností

v_0	v_f	počet řeš. v $u_+(t)$	počet řeš. v $u_-(t)$
>0	>0	0,1	0,1,2
>0	<0	0,1	0,1
<0	>0	0,1	0,1
<0	<0	0,1,2	0,1

zjistíme, že maximální počet řešení je v každém případě 3.

Zbývá dokázat, že k určení t-optimální strategie řízení stačí zjistit vzájemnou polohu γ_0 a x_f .



Obrázek 3.36: Možnosti poloh stavu x_0 a x_f a příslušných křivek γ pro případ, že omezení na zrychlení může být dosaženo.

Důkaz. Důkaz provedeme pro $v_0 > 0$ a rozdělíme ho na případ, kdy $v_f \geq v_0$ a na případ kdy $v_f < v_0$. Opět využijeme implicitních popisů křivek γ_0 a γ_f . Pro zápornou hodnotu v_0 je důkaz obdobný a můžeme ho snadno provést stejným způsobem pomocí transformace (3.42).

Předpokládejme, že bod x_f leží vpravo od křivky γ_0 . Potom s využitím (3.53) a (3.62) dostaneme vztah mezi x_0 a x_f ve tvaru

$$s_f > \begin{cases} s_0 + \sqrt{\frac{v_f - v_0}{B_M}}(v_0 + v_f), & \text{pro } \begin{cases} v_0 \leq v_f \leq \min\{v_0 + \frac{A_M^2}{2B_M}, VM\} \\ v_0 > v_f \geq \max\{v_0 + \frac{A_M^2}{2B_M}, -VM\} \end{cases} \\ s_0 + \frac{A_M^2(v_0 + v_f) + B_M(v_f^2 - v_0^2)}{2A_M B_M}, & \text{pro } \begin{cases} v_0 + \frac{A_M^2}{2B_M} \leq v_f \leq VM \\ v_0 + \frac{A_M^2}{2B_M} > v_f \geq -VM \end{cases} \end{cases} \quad (3.71)$$

kde

$$B_M = \begin{cases} BM & \text{pro } v_0 \leq v_f \\ -BM & \text{pro } v_0 > v_f \end{cases}, \quad A_M = \begin{cases} AM & \text{pro } v_0 \leq v_f \\ -DM & \text{pro } v_0 > v_f \end{cases} \quad (3.72)$$

Dokažme, že v tomto případě existuje průsečík křivek γ_{0+} a γ_{f+} , což znamená existenci řešení v sekvenci $u_+(t)$.

- $v_f \geq v_0$: Z předchozího víme, že pro tento případ je γ_{0+} rostoucí na celém svém rozsahu a γ_{f+} klesající na celém svém rozsahu. Aby existoval průsečík stačí aby platila nerovnost $s_f > s_0$. Ta pro libovolnou hodnotu $v_f: v_f \geq v_0$ a (3.71) platí vždy.

- $v_f < v_0$: Aby existoval průsečík křivek γ_{0+} a γ_{f+} musí platit, že křivka γ_{f+} bude pro rychlost v_0 nabývat vyšší hodnoty než s_0 . Z (3.69) dostáváme nerovnici

$$s_0 < \begin{cases} s_f - \sqrt{\frac{v_0 - v_f}{BM}} (v_f + v_0), & \text{pro } v_0 < v_f + \frac{DM^2}{2BM} \\ s_f - \frac{DM^2(v_f + v_0) + BM(v_0^2 - v_f^2)}{2DM BM}, & \text{pro } v_0 > v_f + \frac{DM^2}{2BM} \end{cases} \quad (3.73)$$

Z (3.71) za předpokladu $v_f: v_f < v_0$ dostaneme

$$s_f > \begin{cases} s_0 + \sqrt{\frac{v_f - v_0}{-BM}} (v_0 + v_f), & \text{pro } v_f \in \left[\max\left\{v_0 - \frac{DM^2}{2BM}, -VM\right\}, v_0 \right) \\ s_0 + \frac{DM^2(v_0 + v_f) - BM(v_f^2 - v_0^2)}{2DM BM}, & \text{pro } v_f \in \left[-VM, v_0 - \frac{DM^2}{2BM} \right) \end{cases} \quad (3.74)$$

Zbývá dokázat, že když platí (3.74), potom platí (3.73) a průsečík existuje.

Jestliže platí první nerovnost (3.74), potom může, vzhledem k uvedeným podmínkám, nastat pouze první nerovnost (3.73). Navíc obě nerovnosti představují stejný vztah. Dále pokud platí druhá nerovnost (3.74), potom může nastat pouze druhá nerovnost (3.73) a také tyto nerovnosti jsou ekvivalentní.

Dokázali jsme existenci řešení v sekvenci $u_+(t)$ za předpokladu, že bod x_f leží vpravo od křivky γ_0 . Protože příslušné nerovnosti jsou vždy ekvivalentní, znamená to, že rozdíl mezi hodnotou s_f a křivkou γ_0 v bodě v_f je totožný s rozdílem hodnot $\gamma_f(v_0)$ a s_0 . dostáváme vztah

$$s_f - \gamma_0(v_f) = \gamma_f(v_0) - s_0. \quad (3.75)$$

Nakonec naznačme postup důkazu, že toto řešení je zároveň t-optimálním řešením úlohy.

Důkaz. Předpokládejme, že bod x_f je vpravo od γ_0 (obě strany rovnice (3.75) jsou kladné) a řešení v sekvenci $u_+(t)$ existuje. Potom, s využitím vztahu (3.75) a postupů uvedených výše, lze dokázat, že průsečík křivek γ_{0-} a γ_{f-} může existovat pouze pokud je $v_0 > 0$ a $v_f > 0$, obrázek 3.33 (vpravo). V tomto případě může dojít ke křížení pod hodnotami v_0 a v_f , a příslušné řešení v sekvenci $u_-(t)$ bude vždy protínat rovinu s - v na nižší hladině rychlosti než existující řešení v sekvenci $u_+(t)$. V ostatních případech, v důsledku tvarů křivek γ a vztahu (3.75), ke křížení γ_{0-} a γ_{f-} dojít nemůže, a řešení v sekvenci $u_+(t)$ je jedinným a zároveň t-optimálním řešením. Tyto vlastnosti jsou shrnuty v následující tabulce.

$s_f - \gamma_0(v_f) > 0$			
v_0	v_f	počet řeš. v $u_+(t)$	počet řeš. v $u_-(t)$
>0	>0	1	0,1,2
>0	<0	1	0
<0	>0	1	0
<0	<0	1	0

Pro řízení systému z *konstantní rychlosti do konstantní rychlosti*, tedy z $\mathbf{x}_0=[s_0, v_0, 0]$ do $\mathbf{x}_f=[s_f, v_f, 0]$ dostaneme následující řešení soustav (3.41). Řešení jsou speciálním typem za předpokladu dosazení $a_0=0, a_f=0$ do obecných řešení v předchozí kapitole.

ŘEŠENÍ SOUSTAVY F_1 PRO PŘÍPAD $a_0=0, a_f=0$.

$$s1t1 = \frac{AM}{BM}$$

$$s1t2 = \frac{-2 AM^2 + 2 BM (VM - v_0)}{2 AM BM}$$

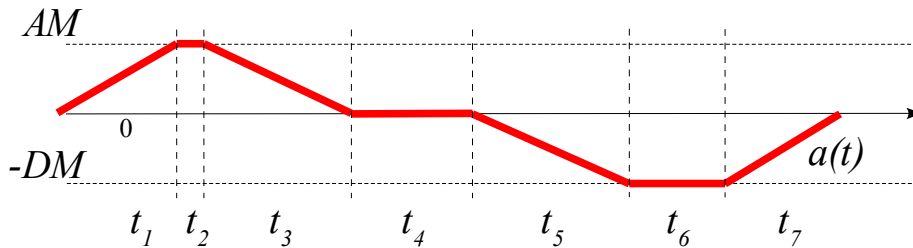
$$s1t3 = \frac{AM}{BM}$$

$$s1t4 = \frac{12 AM^2 BM DM (-VM - v_0) + 12 BM^2 DM (-VM^2 + v_0^2) + AM (24 BM^2 DM (-s_0 + s_f) + 12 BM DM^2 (-VM - v_f) + 12 BM^2 (-VM^2 + v_f^2))}{24 AM BM^2 DM VM}$$

$$s1t5 = \frac{DM}{BM}$$

$$s1t6 = \frac{-2 DM^2 + 2 BM (VM - v_f)}{2 BM DM}$$

$$s1t7 = \frac{DM}{BM}$$



Obrázek 3.37: Typický tvar trajektorie zrychlení pro soustavu $F_1|_{a_0=0, a_f=0}$.

ŘEŠENÍ SOUSTAVY F_2 PRO PŘÍPAD $a_0=0, a_f=0$.

$$s2t1 = s1t1$$

$$s2t2 = s1t2$$

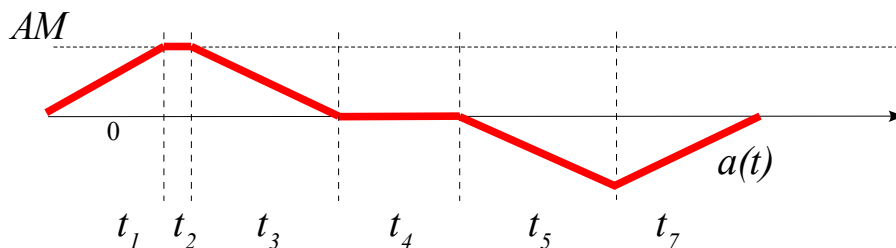
$$s2t3 = s1t3$$

$$s2t4 = \frac{24 AM BM^2 (-s_0 + s_f) + 12 AM^2 BM (-VM - v_0) + 12 BM^2 (-VM^2 + v_0^2) + 24 AM BM (-VM - v_f) \sqrt{BM (VM - v_f)}}{24 AM BM^2 VM}$$

$$s2t5 = \frac{\sqrt{BM (VM - v_f)}}{BM}$$

$$s2t6 = 0$$

$$s2t7 = s2t5$$



Obrázek 3.38: Typický tvar trajektorie zrychlení pro soustavu $F_2|_{a_0=0, a_f=0}$.

ŘEŠENÍ SOUSTAVY F_3 PRO PŘÍPAD $a_0=0, a_f=0$.

$$p_0 = 12 AM^2 DM^2 + 24 AM DM^3 + 12 DM^4 + 12 BM^2 (2 AM (s_0 - s_f) - v_0^2 + v_f^2) + 12 BM (4 AM DM v_f + 2 DM^2 v_f + AM^2 (v_0 + v_f))$$

$$p_1 = 24 BM (AM + DM) \left(\frac{AM DM}{2} + DM^2 + BM v_f \right)$$

$$p_2 = 12 BM^2 DM (AM + DM)$$

$$s_{3t6} = \text{roots} [p_0 + s_{3t6} (p_1 + p_2 s_{3t6}) = 0, s_{3t6}]$$

$$s_{3t1} = s_{1t1}$$

$$s_{3t2} = \frac{2(-AM^2 + DM^2) + 2 BM (DM s_{3t6} - v_0 + v_f)}{2 AM BM}$$

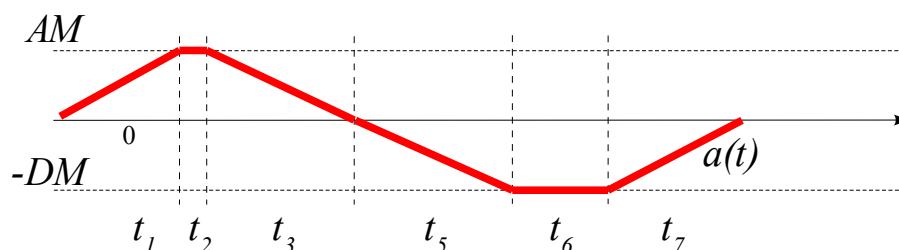
$$s_{3t3} = s_{1t3}$$

$$s_{3t4} = 0$$

$$s_{3t5} = s_{1t5}$$

$$s_{3t6} = \text{roots} [p_0 + s_{3t6} (p_1 + p_2 s_{3t6}) = 0, s_{3t6}]$$

$$s_{3t7} = s_{1t7}$$



Obrázek 3.39: Typický tvar trajektorie zrychlení pro soustavu $F_3|_{a_0=0, a_f=0}$.

ŘEŠENÍ SOUSTAVY F_4 PRO PŘÍPAD $a_0=0, a_f=0$.

$$p_0 = 24 AM BM^2 (s_0 - s_f) + 12 AM^2 BM (v_0 + v_f) + 12 BM^2 (-v_0^2 + v_f^2)$$

$$p_1 = 48 AM BM^2 v_f$$

$$p_2 = 12 BM^2 (AM^2 + 2 BM v_f)$$

$$p_3 = 24 AM BM^3$$

$$p_4 = 12 BM^4$$

$$r_{4t7} = \text{roots} [p_0 + p_1 s_{4t7} + p_2 s_{4t7}^2 + p_3 s_{4t7}^3 + p_4 s_{4t7}^4 = 0, s_{4t7}]$$

$$s_{4t1} = s_{1t1}$$

$$s_{4t2} = \frac{-2 AM^2 + 2 BM^2 s_{4t7}^2 + 2 BM (-v_0 + v_f)}{2 AM BM}$$

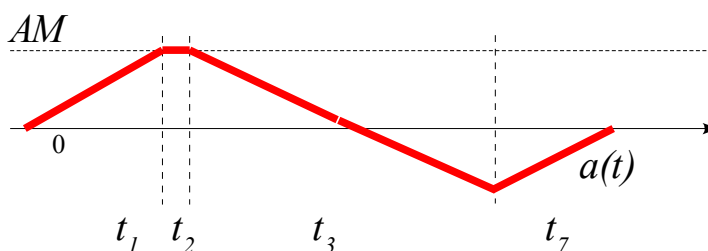
$$s_{4t3} = s_{1t3} + s_{4t7}$$

$$s_{4t4} = 0$$

$$s_{4t5} = 0$$

$$s_{4t6} = 0$$

$$r_{4t7} = \text{roots} [p_0 + p_1 s_{4t7} + p_2 s_{4t7}^2 + p_3 s_{4t7}^3 + p_4 s_{4t7}^4 = 0, s_{4t7}]$$



Obrázek 3.40: Typický tvar trajektorie zrychlení pro soustavu $F_4|_{a_0=0, a_f=0}$.

ŘEŠENÍ SOUSTAVY F_5 PRO PŘÍPAD $a_0=0, a_f=0$.

$$s5t1 = s5t3$$

$$s5t2 = 0$$

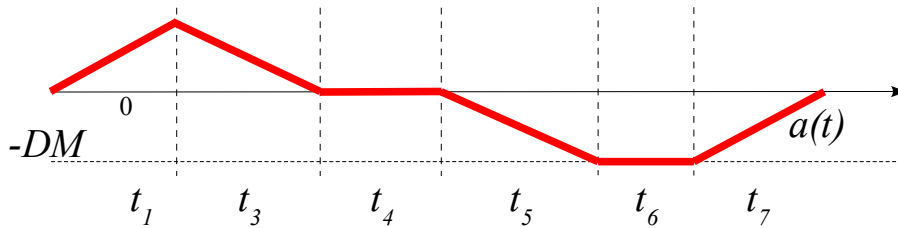
$$s5t3 = \frac{\sqrt{BM (VM-v_0)}}{BM}$$

$$s5t4 = \frac{24 BM^2 DM (-s_0+s_f) + 24 BM DM (-VM-v_0) \sqrt{BM (VM-v_0)} + 12 BM DM^2 (-VM-v_f) + 12 BM^2 (-VM^2+v_f^2)}{24 BM^2 DM VM}$$

$$s5t5 = s1t5$$

$$s5t6 = s1t6$$

$$s5t7 = s1t7$$



Obrázek 3.41: Typický tvar trajektorie zrychlení pro soustavu $F_5|_{a_0=0, a_f=0}$.

ŘEŠENÍ SOUSTAVY F_6 PRO PŘÍPAD $a_0=0, a_f=0$.

$$s6t1 = s5t1$$

$$s6t2 = 0$$

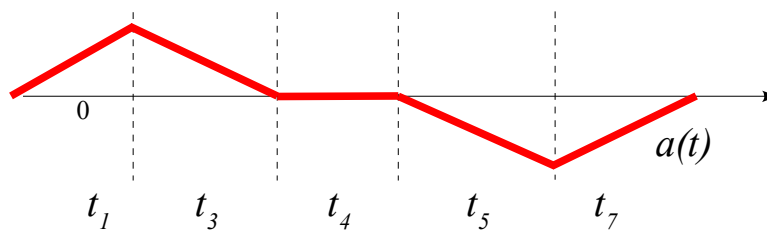
$$s6t3 = s5t3$$

$$s6t4 = \frac{-12 BM^2 (s_0-s_f) - 12 BM^2 t_3 (VM+v_0) - 12 BM^2 t_5 (VM+v_f)}{12 BM^2 VM}$$

$$s6t5 = s2t5$$

$$s6t6 = 0$$

$$s6t7 = s2t7$$



Obrázek 3.42: Typický tvar trajektorie zrychlení pro soustavu $F_6|_{a_0=0, a_f=0}$.

ŘEŠENÍ SOUSTAVY F_7 PRO PŘÍPAD $a_0=0, a_f=0$.

$$p_0 = 24 \text{ BM}^2 \text{ DM} (s_0 - s_f) + 12 \text{ BM DM}^2 (v_0 + v_f) + 12 \text{ BM}^2 (v_0^2 - v_f^2)$$

$$p_1 = 48 \text{ BM}^2 \text{ DM} v_0$$

$$p_2 = 12 \text{ BM}^2 (\text{DM}^2 + 2 \text{ BM} v_0)$$

$$p_3 = 24 \text{ BM}^3 \text{ DM}$$

$$p_4 = 12 \text{ BM}^4$$

$$s_7t_1 = \text{roots} [p_0 + p_1 s_7t_1 + p_2 s_7t_1^2 + p_3 s_7t_1^3 + p_4 s_7t_1^4 = 0, s_7t_1]$$

$$s_7t_1 = \text{roots} [p_0 + p_1 s_7t_1 + p_2 s_7t_1^2 + p_3 s_7t_1^3 + p_4 s_7t_1^4 = 0, s_7t_1]$$

$$s_7t_2 = 0$$

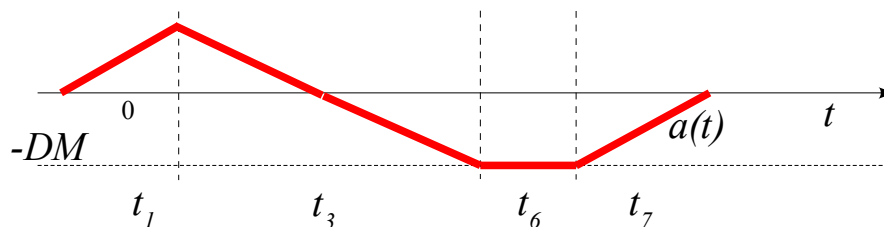
$$s_7t_3 = \frac{\text{DM}}{\text{BM}} + s_7t_1$$

$$s_7t_4 = 0$$

$$s_7t_5 = 0$$

$$s_7t_6 = \frac{-2 \text{ DM}^2 + 2 \text{ BM}^2 s_7t_1^2 + 2 \text{ BM} (v_0 - v_f)}{2 \text{ BM DM}}$$

$$s_7t_7 = s_1t_7$$



Obrázek 3.43: Typický tvar trajektorie zrychlení pro soustavu $F_7|_{a_0=0, a_f=0}$.

ŘEŠENÍ SOUSTAVY F_8 PRO PŘÍPAD $a_0=0, a_f=0$.

$$p_0 = 72 \text{ BM}^3 (\text{BM} s_0^2 - 2 \text{ BM} s_0 s_f + \text{BM} s_f^2 + (v_0 - v_f) (v_0 + v_f)^2)$$

$$p_1 = 288 \text{ BM}^3 (s_0 - s_f) v_f$$

$$p_2 = 72 \text{ BM}^2 (v_0 - v_f)^2$$

$$p_3 = 144 \text{ BM}^2 (s_0 - s_f)$$

$$p_4 = 72 \text{ BM} (-v_0 + v_f)$$

$$\text{dm} = \text{roots} [p_0 + \text{dm} (p_1 + \text{dm} (p_2 + \text{dm} (p_3 + \text{dm} p_4))) = 0, \text{dm}]$$

$$a_m = \frac{\text{BM dm} (\text{dm} s_0 - \text{dm} s_f + (v_0 - v_f)^2) (v_0 - v_f) - \text{dm}^3 (v_0 - v_f)^2 + \text{BM}^2 (s_0 - s_f) (\text{dm} s_0 - \text{dm} s_f + 2 (v_0 - v_f) v_f)}{\text{BM} (\text{BM} (s_0 - s_f)^2 + (v_0 - v_f)^2 (v_0 + v_f))}$$

$$s_8t_1 = \frac{a_m}{\text{BM}}$$

$$s_8t_2 = 0$$

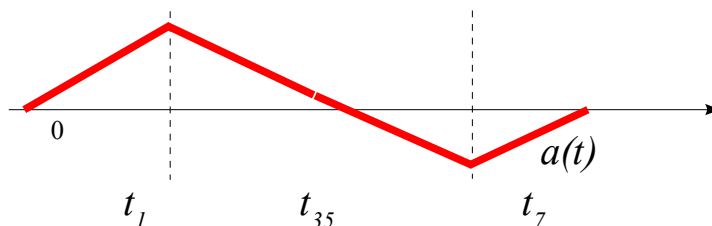
$$s_8t_{35} = \frac{a_m + \text{dm}}{\text{BM}}$$

$$s_8t_4 = 0$$

$$s_8t_5 = 0$$

$$s_8t_6 = 0$$

$$s_8t_7 = \frac{\text{dm}}{\text{BM}}$$



Obrázek 3.44: Typický tvar trajektorie zrychlení pro soustavu $F_8|_{a_0=0, a_f=0}$.

3.5.4 POHYB Z KLIDU DO KLIDU

Řízení systému z klidu $\mathbf{x}_0=[s_0,0,0]$ do klidu $\mathbf{x}_f=[s_f,0,0]$ je speciálním případem předchozích obecnějších pohybů za předpokladu, že platí $a_0=0, a_f=0, v_0=0, v_f=0$. Následující řešení soustav (3.41) musí být až na volnost počáteční polohy s_0 totožné s odvozením téhož v kapitole 3.3.3. Můžeme ho tedy považovat za přímou kontrolu správnosti řešení.

ŘEŠENÍ SOUSTAVY F_1 PRO PŘÍPAD $a_0=0, a_f=0, v_0=0, v_f=0$.

$$\begin{aligned} s1t1 &= AM / BM \\ s1t2 &= VM / AM - AM / BM \\ s1t3 &= AM / BM \\ s1t4 &= \frac{s_f - s_0}{VM} - \frac{(AM + DM)(AM DM + BM VM)}{2 AM BM DM} \\ s1t5 &= DM / BM \\ s1t6 &= VM / DM - DM / BM \\ s1t7 &= DM / BM \end{aligned}$$

ŘEŠENÍ SOUSTAVY F_2 PRO PŘÍPAD $a_0=0, a_f=0, v_0=0, v_f=0$.

$$\begin{aligned} s2t1 &= AM / BM \\ s2t2 &= VM / AM - AM / BM \\ s2t3 &= AM / BM \\ s2t4 &= -\sqrt{VM / BM} - AM / (2 BM) - VM / (2 AM) + (s_f - s_0) / VM \\ s2t5 &= \sqrt{VM / BM} \\ s2t6 &= 0 \\ s2t7 &= \sqrt{VM / BM} \end{aligned}$$

ŘEŠENÍ SOUSTAVY F_3 PRO PŘÍPAD $a_0=0, a_f=0, v_0=0, v_f=0$.

$$\begin{aligned} s3t1 &= AM / BM \\ s3t2 &= -\frac{2 AM + DM}{2 BM} + \sqrt{\left(\frac{AM DM}{2 BM}\right)^2 + \frac{2 AM DM (s_f - s_0)}{AM + DM}} \\ s3t3 &= AM / BM \\ s3t4 &= 0 \\ s3t5 &= DM / BM \\ s3t6 &= -\frac{AM + 2 DM}{2 BM} + \sqrt{\left(\frac{AM DM}{2 BM}\right)^2 + \frac{2 AM DM (s_f - s_0)}{AM + DM}} \\ s3t7 &= DM / BM \end{aligned}$$

ŘEŠENÍ SOUSTAVY F_4 PRO PŘÍPAD $a_0=0, a_f=0, v_0=0, v_f=0$.

$$\begin{aligned} dm_{1,2} &= -AM / 2 + 1 / 2 \sqrt{(AM^2 \pm 4 BM \sqrt{(2 AM (s_f - s_0))})} \\ s4t1 &= AM / BM \\ s4t2 &= \frac{dm_{1,2}^2}{AM BM} - AM / BM \\ s4t3 &= AM / BM \\ s4t4 &= 0 \\ s4t5 &= dm_{1,2} / BM \\ s4t6 &= 0 \\ s4t7 &= dm_{1,2} / BM \end{aligned}$$

ŘEŠENÍ SOUSTAVY F_5 PRO PŘÍPAD $a_0=0, a_f=0, v_0=0, v_f=0$.

$$s5t1 = \sqrt{VM/BM}$$

$$s5t2 = 0$$

$$s5t3 = \sqrt{VM/BM}$$

$$s5t4 = -\sqrt{VM/BM} - DM/(2BM) - VM/(2DM) + (s_f - s_0)/VM$$

$$s5t5 = DM/BM$$

$$s5t6 = VM/DM - DM/BM$$

$$s5t7 = DM/BM$$

ŘEŠENÍ SOUSTAVY F_6 PRO PŘÍPAD $a_0=0, a_f=0, v_0=0, v_f=0$.

$$s6t1 = \sqrt{VM/BM}$$

$$s6t2 = 0$$

$$s6t3 = \sqrt{VM/BM}$$

$$s6t4 = (s_f - s_0)/VM - 2\sqrt{VM/BM}$$

$$62t5 = \sqrt{VM/BM}$$

$$62t6 = 0$$

$$62t7 = \sqrt{VM/BM}$$

ŘEŠENÍ SOUSTAVY F_7 PRO PŘÍPAD $a_0=0, a_f=0, v_0=0, v_f=0$.

$$am_{1,2} = -DM/2 + 1/2\sqrt{(DM^2 \pm 4BM\sqrt{(2DM(s_f - s_0))})}$$

$$s7t1 = am_{1,2}/BM$$

$$s7t2 = 0$$

$$s7t3 = am_{1,2}/BM$$

$$s7t4 = 0$$

$$s7t5 = DM/BM$$

$$s7t6 = \frac{am_{1,2}^2}{DM BM} - DM/BM$$

$$s7t7 = DM/BM$$

ŘEŠENÍ SOUSTAVY F_8 PRO PŘÍPAD $a_0=0, a_f=0, v_0=0, v_f=0$.

$$s8t1 = \sqrt[3]{\frac{s_f - s_0}{2BM}}$$

$$s8t2 = 0$$

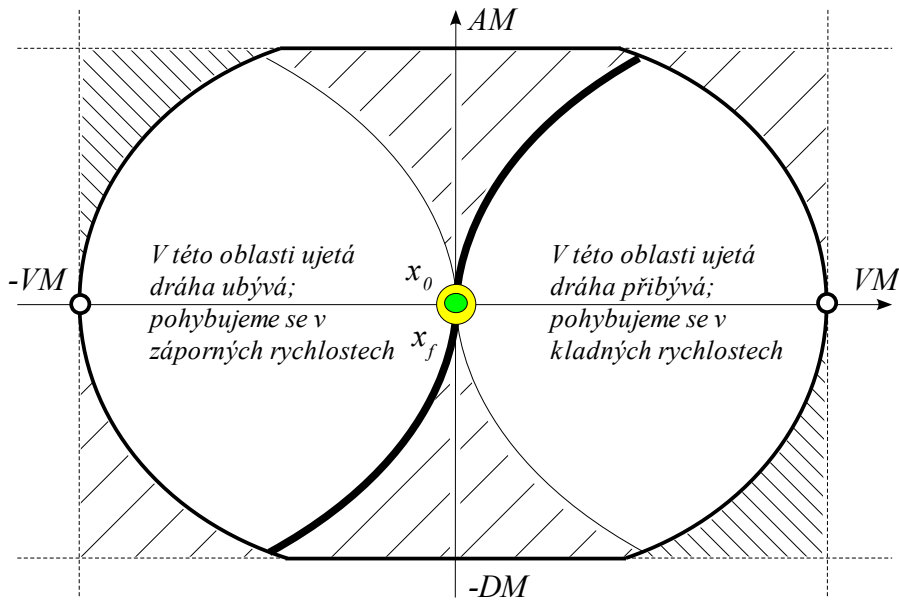
$$s8t3 = \sqrt[3]{\frac{s_f - s_0}{2BM}}$$

$$s8t4 = 0$$

$$s8t5 = \sqrt[3]{\frac{s_f - s_0}{2BM}}$$

$$s8t6 = 0$$

$$s8t7 = \sqrt[3]{\frac{s_f - s_0}{2BM}}$$



Obrázek 3.45: Stavový podprostor v - a pro řetězec 3 integrátorů s omezeními, s vyznačením zakázaných oblastí pro počáteční a koncový stav v případě pohybu z klidu do klidu.

Poznámka: Pro tento typ úlohy můžeme samozřejmě také apriori rozhodnout, kterou sekvencí řízení použít. Pokud je $s_f > s_0$, potom bude použito $u_+(t)$ řízení. V opačném případě řízení $u_-(t)$.

Rozdělující varieta dělí stavový podprostor s - v jednoznačně na dvě souvislé oblasti. Pokud bychom dále zjednodušili úlohu a nastavili $AM = DM$, $s_0 = 0$, $s_f = SM$, dostali bychom původní řešení algoritmu BAVS.

3.5.5 STANDARDNÍ APLIKACE ALGORITMU *GBAVS* V PLÁNOVÁNÍ POHYBU

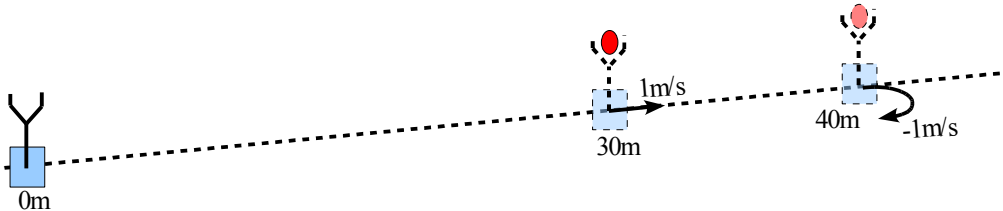
Většina plánovacích algoritmů pracuje s klidovým počátečním a koncovým stavem, přičemž umožňuje napojování dílčích pohybů pomocí tzv. blendingu. Jde o zavedený způsob přechodnapojení na další plánovanou trajektorii pohybu podle normy PLCopen motion control [81], který je využíván pro přepřelánování trajektorie během pohybu. Omezení mohou být během přesunu do jisté míry porušena. Algoritmy zpravidla neumí pracovat s nenulovými hodnotami zrychlení, a tak je možné využít blendingu pouze během konstantního pohybu.

Algoritmus *GBAVS* umožňuje libovolné plánování pohybu, přepřelánování během vykonávání pohybu, napojování dílčích pohybů, řetězení do složitějších struktur a také časovou synchronizaci v případě víceosých strojů. Pro názornost demonstrujeme dvě základní možnosti využití tohoto algoritmu.

PLÁNOVÁNÍ POHYBU V JEDNÉ OSE (SINGLE-AXIS MOTION PLANNING)

Možnost plánování trajektorií pohybu a jejich hladké napojování v poloze i rychlosti je hlavní předností třetího řádu, zrovna tak jako v případě využití spline křivek. Nastíňme v následujícím příkladu jakým způsobem můžeme postupovat při plánování.

Příklad. Uvažujme robotickou ruku, která se pohybuje po kolejové dráze. Na začátku pohybu se robot nachází v poloze 0m. Cílem robotické ruky je přemístit předmět z jednoho místa ve vzdálenosti 30m do jiného místa ve vzdálenosti 40m, a poté se vrátit zpět. Vlastností ruky je, že předmět je možné uchopit při konstantní rychlosti pohybu po kolejích do 1m/s, přičemž maximální rychlost v kladném směru je například omezena stoupáním, tzn. na 4 m/s, v záporném pak klesáním na 5 m/s. Pohyb tedy rozdělíme do třech dílčích pohybů, ve kterých budou nastaveny specifická omezení, ale požadujeme, aby jednotlivé pohyby na sebe navzájem navazovaly.



Obrázek 3.46: Naznačení klíčových parametrů podél přesunu systému.

Prvním dílčím pohybem je přesun robota k předmětu a jeho uchopení. Robot je v počátku v klidu a konec toho pohybu je okamžik uchopení na poloze 30m při rychlosti 1m/s a nulovém zrychlení. Předpokládejme, že robot může v nezatíženém stavu zrychlovat a zpomalovat podle následujících omezení.

POČÁTEK – UCHOPENÍ:

$$\mathbf{x}_0 = [0 \ 0 \ 0]^T \rightarrow \mathbf{x}_{f1} = [30 \ 1 \ 0]^T; \quad AM = 1, VM = 4, DM = 2, BM = |1|$$

Druhým dílčím pohybem je přesun předmětu do okamžiku reverzace polohy k návratu. Robot je v zatíženém stavu, a tudíž jeho zrychlování i brzdění je na nižší úrovni. Pro plynulejší náběh na tato omezení je omezena i strmost vstupu BM . Počáteční stav je dán koncovým stavem předchozího pohybu. Koncový pak faktem, že systém se bude vracet zpět (nulová rychlost), ale s co nejrychlejší reverzací (minimum z omezení na zrychlení v druhém a třetím pohybu).

UCHOPENÍ – POLOŽENÍ:

$$\mathbf{x}_{f1} = [30 \ 1 \ 0]^T \rightarrow \mathbf{x}_{f2} = [40 \ 0 \ -1]^T; \quad AM = 0.5, VM = 4, DM = 1.5, BM = |0.5|$$

Třetím dílčím pohybem je již návrat zpět do počátku v nezatíženém stavu. Systém může opět zrychlovat a zpomalovat s hodnotami jako v prvním pohybu. Protože systém bude couvat, je nutné zaměnit omezení na zrychlení a zpomalení v algoritmu. Podrobné vysvětlení v poznámce na konci kapitoly.

POLOŽENÍ – POČÁTEK:

$$\mathbf{x}_{f2} = [40 \ 0 \ -1]^T \rightarrow \mathbf{x}_0 = [0 \ 0 \ 0]^T; \quad AM = 2, VM = 5, DM = 1, BM = |1|.$$

Nezávislými výpočty jednotlivých sekvencí řízení pomocí algoritmu *GBAVS* a jejich zřetězením dostaneme výslednou řídicí sekvenci

$$\begin{bmatrix} u \\ t_{1..21} \end{bmatrix} = \begin{bmatrix} 1 & 0 & -1 & 0 & -1 & 0 & 1 & 0.5 & 0 & -0.5 & 0 \\ 1 & 3 & 1 & 2.8349 & 1.7321 & 0 & 1.7321 & 1 & 1.4481 & 3.5393 & 0 \\ -0.5 & 0 & 0.5 & -1 & 0 & 1 & 0 & 1 & 0 & -1 & \\ 0 & 0 & 0.5393 & 0 & 4.5 & 1 & 2.7583 & 2 & 0.5 & 2 & \end{bmatrix},$$

která průchodem generátoru 3. řádu generuje trajektorie pohybu, obrázek 3.47. Protože každá ze tří dílčích částí je t-optimální trajektorií vzhledem ke svým omezením, potom platí, že i výsledná trajektorie je t-optimální trajektorií [4]. Celkový přesun je tedy vzhledem k daným omezením t-optimálním přesunem a trvá $\tau_{21}=30.5840932\dots s$.

Pro zajímavost se podívejme na stejný problém generování trajektorie, tentokrát ale pomocí generátoru 2. řádu. Strmost náběhu na omezení zrychlení a zpomalení je již nekonečná, ostatní omezení zůstávají zachována.

POČÁTEK – UCHOPENÍ:

$$\mathbf{x}_0=[0 \ 0 \ 0]^T \rightarrow \mathbf{x}_{f1}=[30 \ 1 \ 0]^T; \quad AM=1, VM=4, DM=2$$

UCHOPENÍ – POLOŽENÍ:

$$\mathbf{x}_{f1}=[30 \ 1 \ 0]^T \rightarrow \mathbf{x}_{f2}=[40 \ 0 \ -1]^T; \quad AM=0.5, VM=4, DM=1.5$$

POLOŽENÍ – POČÁTEK:

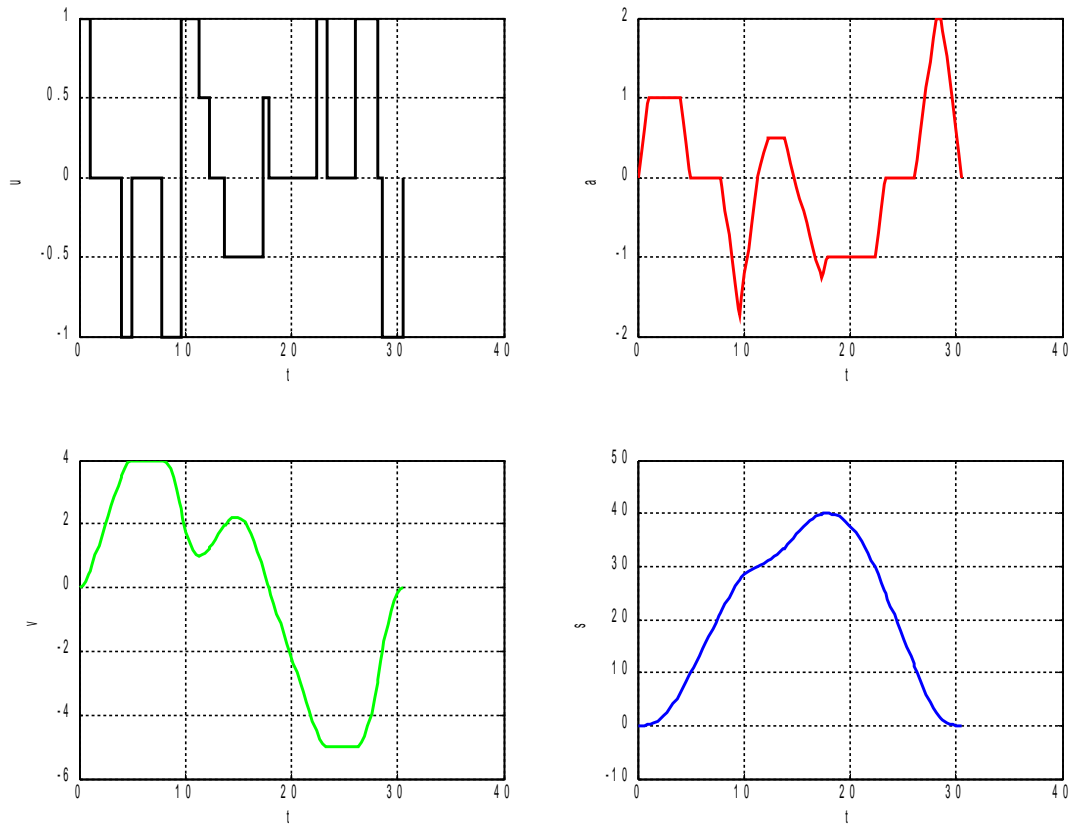
$$\mathbf{x}_{f2}=[40 \ 0 \ -1]^T \rightarrow \mathbf{x}_0=[0 \ 0 \ 0]^T; \quad AM=1, VM=5, DM=2$$

Nezávislými výpočty jednotlivých sekvencí řízení a jejich zřetěžením dostaneme výslednou řídicí sekvenci

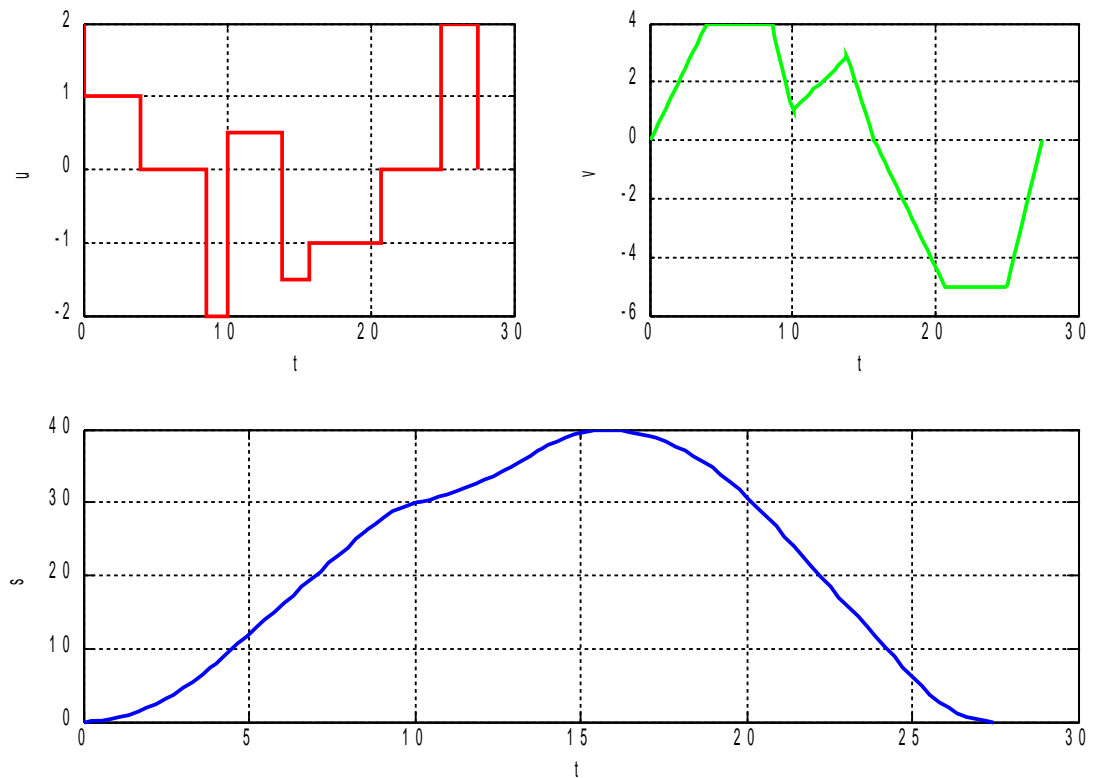
$$\begin{bmatrix} u \\ t_{1..15} \end{bmatrix} = \begin{bmatrix} 2 & 1 & 0 & -2 & -1 & 1.5 & 0.5 & 0 & -1.5 & -0.5 \\ 0 & 4 & 4.5625 & 1.5 & 0 & 0 & 3.7446 & 0 & 1.9149 & 0 \\ -2 & -1 & 0 & 2 & 1 \\ 0 & 5 & 4.25 & 2.5 & 0 \end{bmatrix},$$

která průchodem generátoru 2. řádu generuje následující trajektorie s celkovou dobou přesunu $\tau_{15}=27.4719\dots s$.

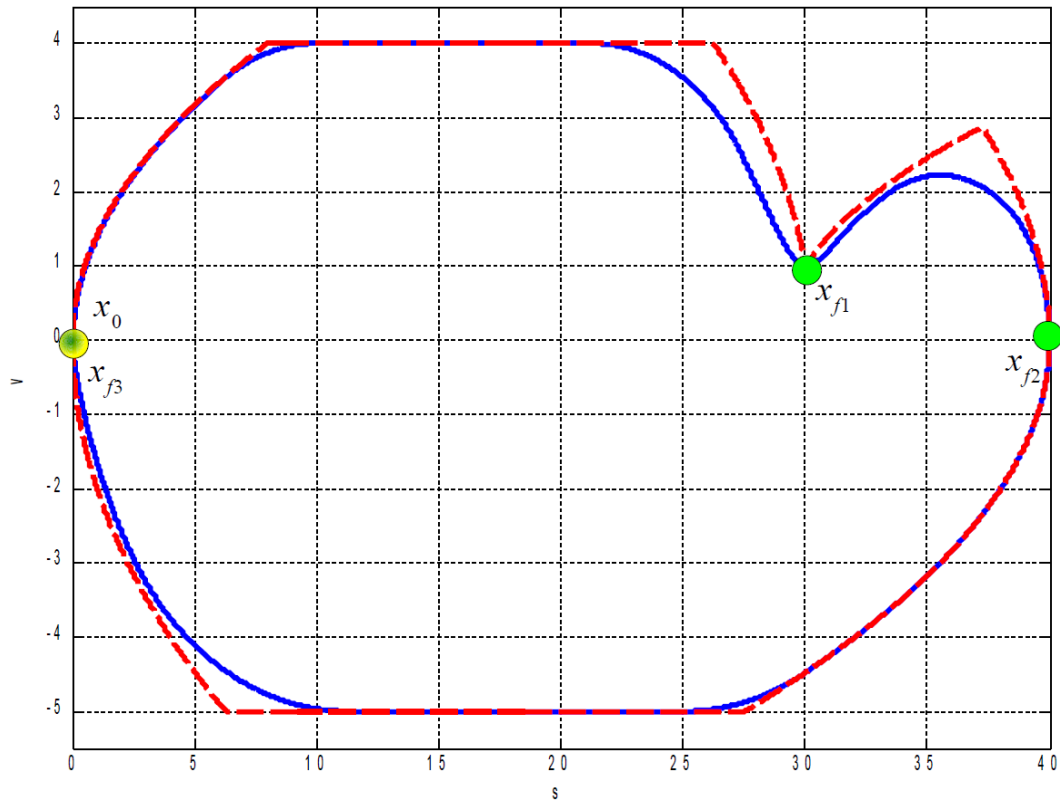
Z grafů 3.47,3.48 je vidět, že systém v obou případech minimalizuje časovou náročnost pohybu. V každém okamžiku se snaží najíždět na maximální rychlost maximální možným zrychlením. V místě uchopení objektu \mathbf{x}_{f1} zpomaluje na „zachytávací“ rychlost, poté zpět zrychluje. V místě položení předmětu \mathbf{x}_{f2} systém mění svůj směr s maximálním zrychlením. Poté najíždí na maximální rychlost a na té setrvává do poslední chvíle, z které je ještě schopen zpomalit a zastavit zpět v počátku. Srovnáním dob trvání pohybů a tvaru trajektorií v rovině s-v vidíme klíčové rozdíly, obrázek 3.49.



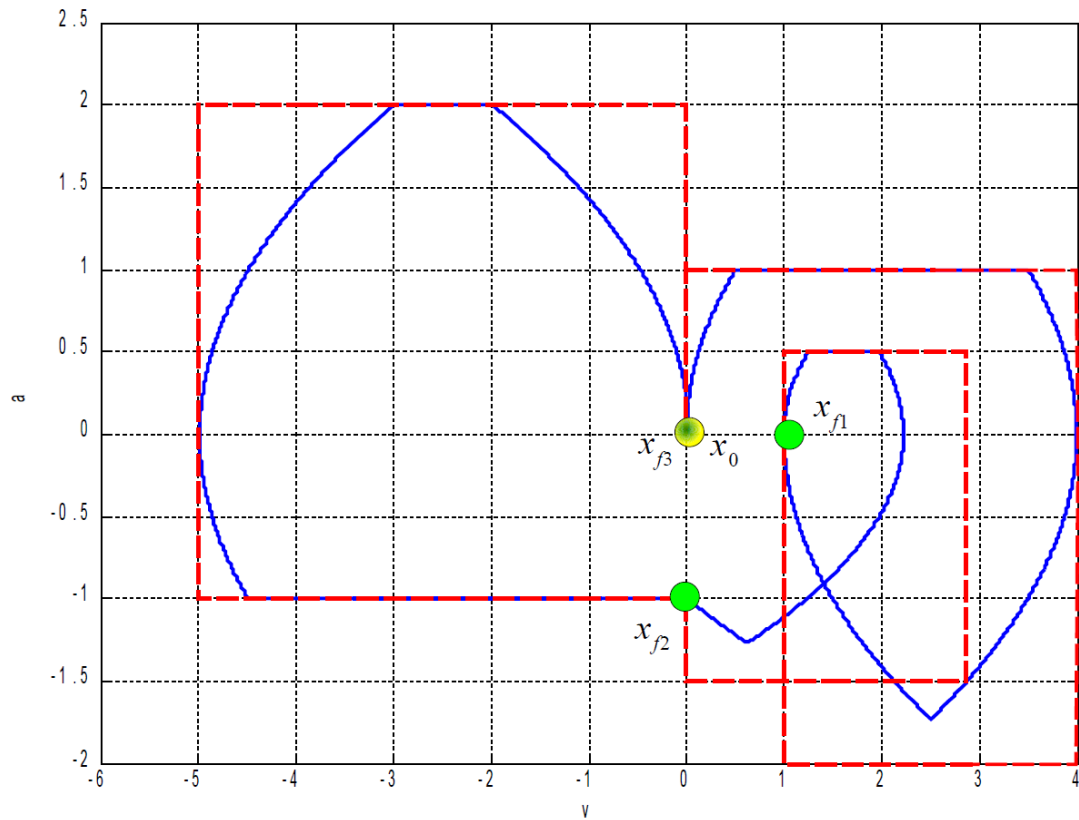
Obrázek 3.47: Časové průběhy vstupu a jednotlivých stavů systému 3. řádu.



Obrázek 3.48: Časové průběhy vstupu a jednotlivých stavů systému 2. řádu.



Obrázek 3.49: Trajektorie systému 2. (čárkovaně) a 3. řádu v podprostoru s - v .



Obrázek 3.50: Trajektorie systému 2. (čárkovaně) a 3. řádu v podprostoru v - a .

POHYB Z BODU DO BODU SE SYNCHRONIZACÍ (MULTI-AXIS MOTION PLANNING)

V úlohách, kde nezáleží na tvaru trajektorie přesunu, je problém plánování velmi často převeden na plánování každé dílčí osy zvlášť, ale tak, aby všechny osy ukončili svůj pohyb současně. Typickým příkladem jsou obráběcí stroje, osazovací stroje, nebo roboty pro přesun materiálu typu *pick and place*. Naplánovaný pohyb každé osy musí být přizpůsoben nejpomalejší ose. Pro pohyb z klidu do klidu lze využít *měřítkování* faktorem k a nalézt převodní vztahy pro omezení

$$k = \frac{t_f}{\bar{t}_f}, \quad (3.76)$$

$$\bar{B}M = BM k^3, \bar{A}M = AM k^2, \bar{V}M = VM k,$$

kde t_f je původní čas přesunu, \bar{t}_f je výsledný čas přesunu a řídicí strategie je totožná s použitím modifikovaných omezení $\bar{B}M$, $\bar{A}M$, $\bar{V}M$. Bohužel tato strategie pracuje pouze pro úlohy s pohybem z *klidu do klidu*. Následující dedukce stručně vysvětluje proč tomu tak je.

Předpokládejme, že aktuální stav generátoru pohybu musí splňovat rovnici

$$x(t) = x(\bar{t}(t)), \quad (3.77)$$

kde \bar{t} je modifikovaný čas, daný substitucí $\bar{t} = t/k$. Výpočtem první derivace

$$\dot{x}(t) = \frac{\partial x(\bar{t})}{\partial \bar{t}} \frac{d\bar{t}}{dt} \rightarrow \dot{x}(t) = \dot{x}(\bar{t})/k \quad (3.78)$$

zjistíme, že rychlost stavu je závislá na faktoru k . To znamená, že jediný společný počáteční a koncový stav rychlosti pro $\dot{x}(t)$ a $\dot{x}(\bar{t})$, který nebude porušen faktorizací, je stav nulový. Omezení na rychlost VM musí být ovlivněno faktorem k .

Výpočtem druhé derivace (3.77)

$$\ddot{x}(t) = \frac{\partial \dot{x}(\bar{t})}{\partial \bar{t}} \frac{d\bar{t}}{dt} \rightarrow \ddot{x}(t) = \ddot{x}(\bar{t})/k^2 \quad (3.79)$$

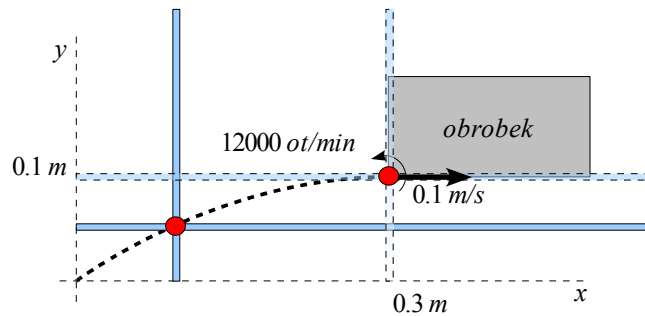
zjistíme funkční závislost i druhé derivace na faktoru k . Opět dostaneme omezení na x_0, x_f ve tvaru nulových stavů zrychlení. Omezení na zrychlení AM je ovlivněno faktorem k^2 . Podobně bychom dostali závislost derivace zrychlení a omezení BM na faktoru k^3 , jak tomu je v (3.76).

V případě pohybu z obecného stavu do obecného stavu již není možné tyto vztahy použít. Navíc by mohl nastat případ, kdy například transformované omezení na rychlost $\bar{V}M$ by vyšlo nižší než je počáteční nebo koncová rychlost, což je z hlediska řešitelnosti úlohy nepřijatelné. Můžeme ale modifikovat velikost *Jerku* (BM). To je navíc z praktického hlediska nejvýhodnější, protože tím snížíme strmost náběhu zrychlení a zvýšíme plynulost pohybu. Ostatní omezení není třeba modifikovat, ale je to samozřejmě možné. Snížení hodnoty BM za účelem prodloužení času pohybu není triviální úlohou a lze nalézt pomocí iteračního algoritmu využívajícího plánovač *GBAVS*. To umožňuje naplánovat pohyb každé dílčí osy víceosého stroje, jehož podmínky na pohyb se v čase mění a jehož nezávislé osy chceme z nějakého důvodu synchronizovat. Popřípadě v průběhu pohybu přeplánovat a synchronizovat například plynulé zastavení všech os. Výhodou obecné funkčnosti je také stálá znalost „hranic“ pohybu, kterou je možné za běhu vyčíslit a určit, zda stroj za daných podmínek pohybu dokáže zastavit ve svém pracovním prostoru, nebo kolik prostoru bude potřebovat pro změnu směru apod.

Poznamenejme, že tento postup obecně nezachovává tvar řídicí funkce a řešení nemusí existovat. To nastane v případě, kdy snížení *Jerku* nebo i jiných omezení „zpomalí“ systém natolik, že není například možné dožadovaného koncového stavu bez překročení omezení. Je možné uměle snížit jiné omezení a zkusit najít vyšší hodnotu *Jerku*, ale ani to se nemusí povést.

Poznámka. Při nezávislém plánování každé osy je třeba brát v potaz, že se jedná o nekoordinovaný pohyb. Není předem jasné, jakým způsobem se bude koncová část mezi body pohybovat a hrozí případná kolize.

Příklad. Uvažujme planární obráběcí frézku s třemi nezávislými stupni volnosti, tj. dva posuvy a natočení frézy. Chceme, aby nástroj najel na požadovanou polohu s požadovanou rychlostí v ose x , $v_x = 0.1$ m/s a přitom se již otáčel předepsanou rychlostí pro úběr frézovaného materiálu, například 12000 ot/min.



Obrázek 3.51: Základní struktura úlohy s vyznačením klíčových dat a předpokládané trajektorie systému v pracovní rovině.

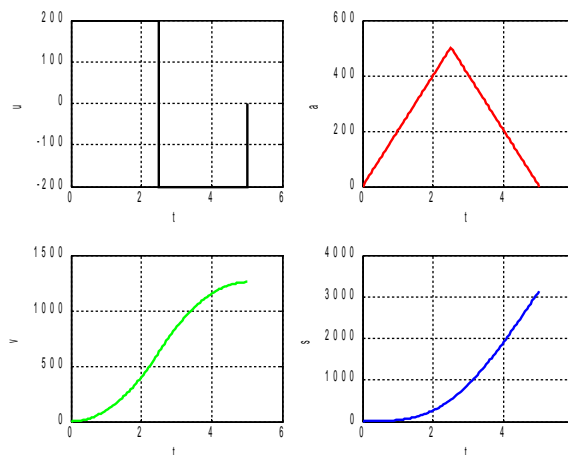
Vstupní data pro generátor pohybu mohou vypadat následovně:

osa x : $BM = 5, AM = 1, DM = AM, VM = 0.2, x_0 = [0, 0, 0], x_f = [0.3, 0.1, 0]$

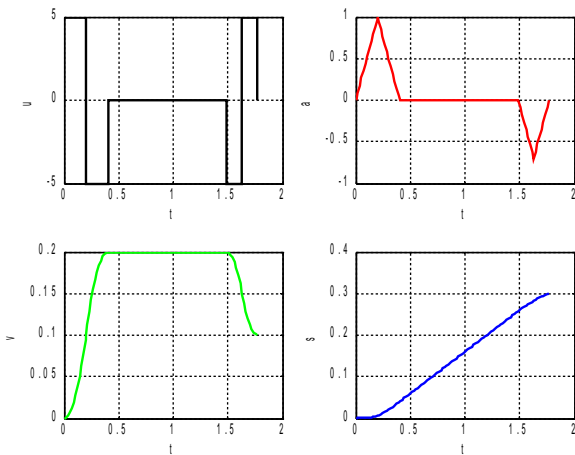
osa y : $BM = 5, AM = 1, DM = AM, VM = 0.2, x_0 = [0, 0, 0], x_f = [0.1, 0, 0]$

osa ϕ : $BM = 200, AM = 1000, DM = AM, VM = 1500, x_0 = [0, 0, 0], x_f = [-, 1256(12000 \text{ ot/min}), 0]$

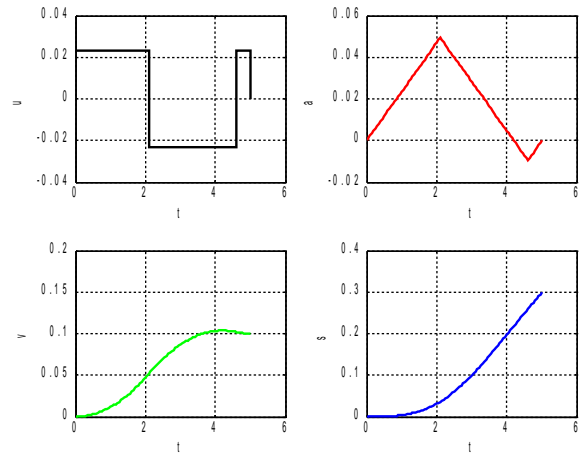
Protože nás nezajímá natočení frézy, pouze její aktuální rychlost, použijeme pro plánování jejího pohybu generátor 2.řádu $GAVS$. Dostáváme čas přesunu $tf_\phi = 5.01s$. Pro osu x a y použijeme generátor 3.řádu $GBAVS$ a dostáváme časy přesunu $tf_x = 1.77s$ a $tf_y = 0.9s$. Vidíme, že je nutné pro synchronizaci os prodloužit čas pohybu posuvů vzhledem k pohybu frézy. Iteračním algoritmem využívajícím bisekci a algoritmus $GBAVS/GAVS$ dostáváme následující modifikované hodnoty $\bar{B}M_x = 0.0235, \bar{B}M_y = 0.0253$. Následující grafy ukazují průběh plánovaných trajektorií před a po synchronizaci.



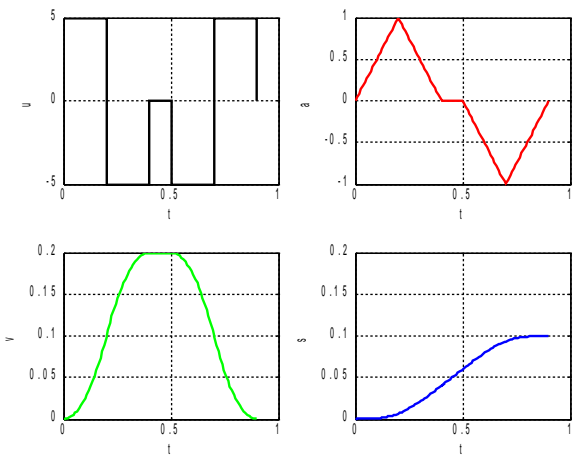
Obrázek 3.52. Trajektorie pohybu obráběcí frézy.



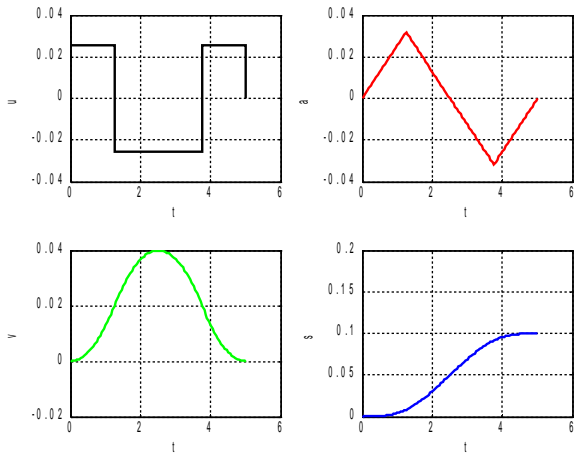
Obrázek 3.54. Trajektorie pohybu x-posuvu před časovou synchronizací.



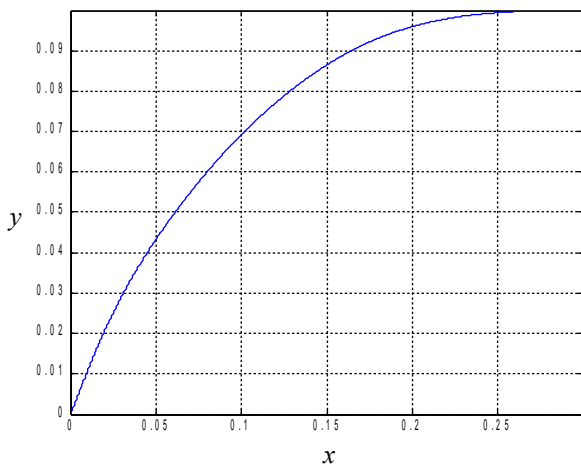
Obrázek 3.53. Trajektorie pohybu x-posuvu po časové synchronizaci.



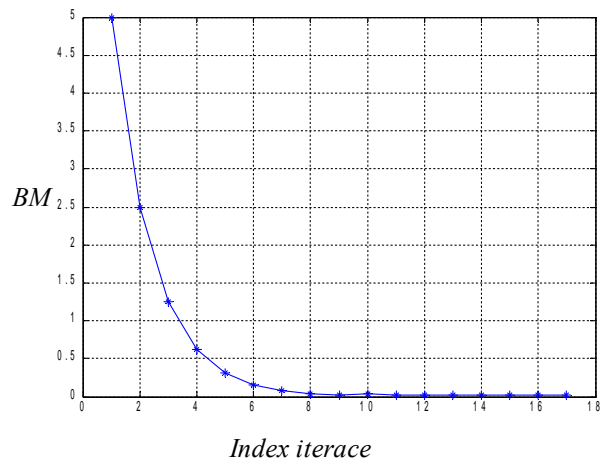
Obrázek 3.57. Trajektorie pohybu y-posuvu před časovou synchronizací.



Obrázek 3.58. Trajektorie pohybu y-posuvu po časové synchronizaci.



Obrázek 3.55. Výsledná trajektorie pohybu obráběcí frézy v rovině x-y.



Obrázek 3.56. Vývoj parametru BM pro x-posuv pomocí metody bisekce. Přesnost koncového času byla nastavena na 10^{-2} .

4 PLÁNOVÁNÍ POHYBU PODÉL SPECIFIKOVANÉ CESTY

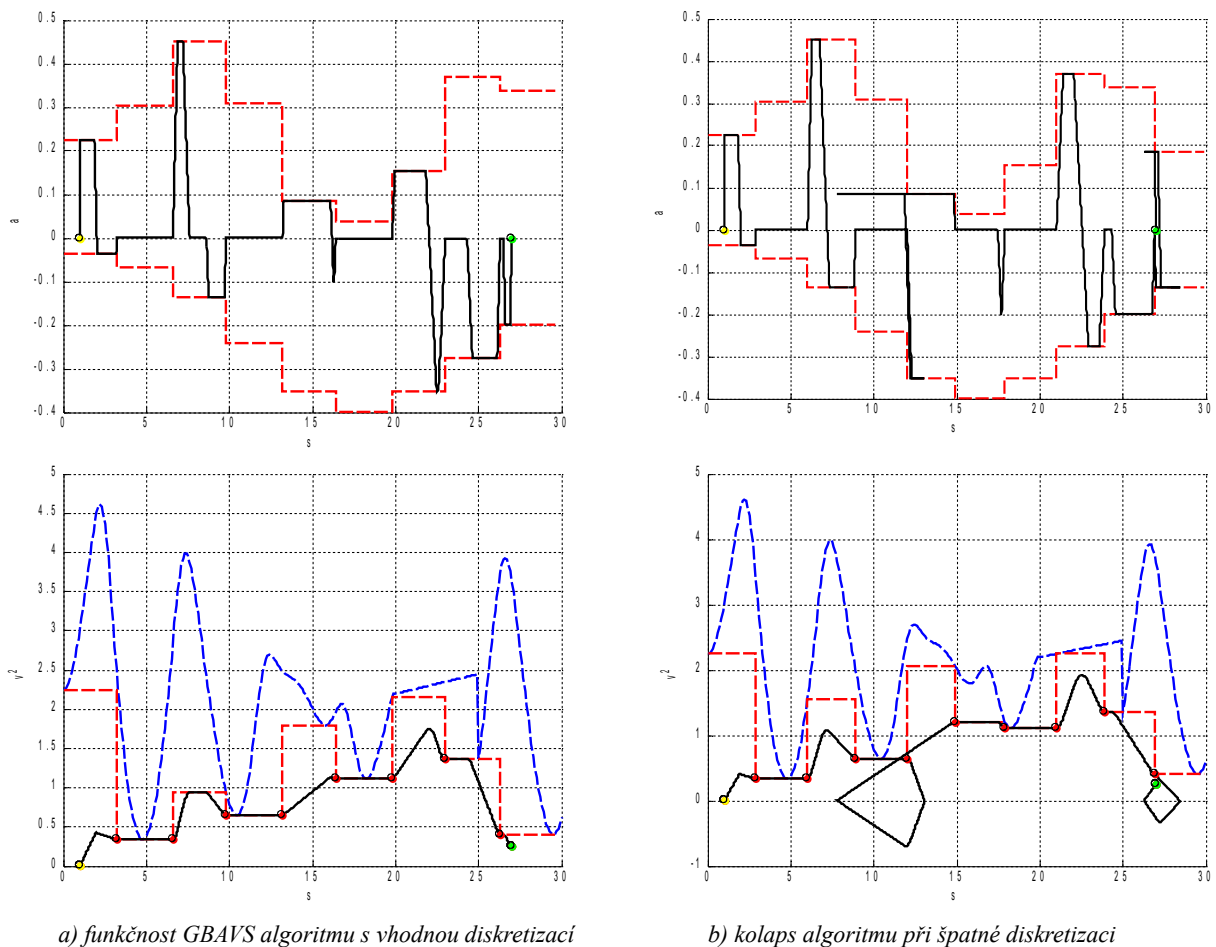
Mnoho úloh v robotice požaduje, aby robot nebo jeho pohyblivé části sledovali zadanou křivku v pracovním prostoru robota, nebo v jeho kloubových souřadnicích. Typickým příkladem je CNC obrábění nebo pohyb mobilního robota v prostoru s překážkami nebo dalšími roboty. Cesta, kterou má robot vykonat, je koordinovaným pohybem, definována posloupností bodů nebo třeba analytickou funkcí v pracovním prostoru. Profil rychlosti nemusí být předem definován, pouze jsou známé limity jednotlivých aktuátorů, případně další omezení na pohyb v prostoru robota. Možnosti pohybu robota jsou zcela určeny pohybovými rovnicemi popisující dynamiku a kinematickými vazbami vzhledem k specifikované cestě. Matematický popis i jednoduchých robotů je téměř vždy nelineární. Geometrie cesty (křivost) se projeví na dynamice stroje například ve formě setrvačnosti, odstředivých sil, koriolisových sil apod. Na kinematice pak formou nelineárních transformací pohybových (fázových) omezení, možností výskytu kinematických a dynamických singularit. Plánovací algoritmus musí proto respektovat komplexní chování systému podél specifikované cesty.

V úvodu bylo na motivačním příkladu planárního robota naznačeno, že i pro tak jednoduché systémy dostáváme v důsledku kinematických nelineárních transformací, kino-dynamických a fázových omezení nelineární a nespojitě omezující křivky. V kapitole 3.3 byly nastíněny standardní postupy využívající numerické integrace pohybových rovnic v prostoru generátoru trajektorie [13],[46],[93],[113]. Tyto postupy se využívají z pravidla pro offline plánování strojů s mnoha pracovními cykly.

Dalším možným postupem je diskretizace křivek omezení a převedení na po částech konstantní omezující funkce. Pro každý její konstantní úsek bychom pak mohli použít algoritmy typu *GBAVS*, nebo podobné techniky [55],[56]. Jednoduché zřetězení dílčích výpočtů algoritmu *GBAVS* vyžaduje dopředu sestavit množinu počátečních a koncových stavů pro každý úsek, podobně jako v příkladu z kapitoly 3.5.5. Ty jsou nastaveny podle diskretizovaných omezení na hranicích každého úseku tak, aby vzájemně navazovaly a tvořily uzlové body pohybu, obrázek 4.1a). Bohužel diskretizaci není možné provést libovolně. Obecně může nastat případ, že diskretizovaná omezení společně s uzlovými body nevedou na řešení v dané sekvenci řízení, viz obrázek 4.1b) nebo řešení neexistuje vůbec.⁹ Mohlo by docházet k překročení omezení jako v [44] i přesto, že je zřejmé, že řešení existuje. Zpravidla je nutné zvětšit krok diskretizace.

Nicméně pro generátor 2.řádu je možné nalézt obecný algoritmus, který využívá tento postup, ale není zatížen výše zmíněnými problémy, viz dále.

⁹ Připomeňme, že omezení mohou nabývat libovolné strmosti, a nelze tedy využít žádné apriorní informace k volbě délky kroku diskretizace.



Obrázek 4.1: Plánování pohybu pro diskretizovaná omezení GBAVS algoritmem.

Kapitola 4 stručně nastiňuje zmíněný problém plánování s obecným tvarem kino-dynamických omezení. V první části bude ukázán vliv kinematiky a dynamiky na kino-dynamická omezení. Dále budou navrženy tři analytické techniky pro generování trajektorie s různou strukturou omezení. V uvedených technikách se bude jednat pouze o generátor 2.řádu, který využívá analytických postupů a jehož výpočetní náročnost je velmi nízká. Výpočetní algoritmus neobsahuje žádné výpočetní smyčky ani cykly. Délka výpočtu je závislá pouze na hustotě diskretizace omezujících funkcí. Výsledky jsou v závěru nasazeny na jednoduchý příklad planárního manipulátoru.

Pro 3.řád je prozatím nutné využít vhodné diskretizace zmíněné výše nebo numerických metod přímé integrace, která výrazným způsobem zvýší časovou náročnost výpočtu.

4.1 KINEMATIKA A DYNAMIKA SYSTÉMU PODÉL SPECIFIKOVANÉ CESTY

V této kapitole uvedeme základní vztahy nutné pro pochopení dalších kapitol. Soustředíme se na mechanické systémy se sériovým spojením jednotlivých pohyblivých částí.

Předpokládejme sériový manipulátor s n stupni volnosti. Konfigurační prostor systému je $n \times 1$,

vektor kloubových souřadnic označíme \mathbf{q} . Pohybové rovnice pro netlumený systém lze například pomocí Lagrangeovy metody [76] vyjádřit obecně ve tvaru

$$M(\mathbf{q})\ddot{\mathbf{q}} + \dot{\mathbf{q}}^T C(\mathbf{q})\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) = \boldsymbol{\tau}, \quad (4.1)$$

kde $M(\mathbf{q})$ je pozitivně definitní symetrická matice $n \times n$, kde $C(\mathbf{q})$ je tenzor $n \times n \times n$ odstředivých a koriolisových koeficientů, $\mathbf{g}(\mathbf{q})$ je $n \times 1$ vektor gravitačních koeficientů a $\boldsymbol{\tau}$ je $n \times 1$ vektor vstupních sil a momentů působících na systém. Předpokládáme tedy, že počet stupňů volnosti je roven počtu akčních členů.

Uvažujme vektorovou funkci $\mathbf{p}(s)$, která předepisuje cestu v pracovním prostoru stroje. Skalární parametr s je přirozenou parametrizací délky křivky. V následujícím textu pro jednoduchost předpokládáme, že přirozená parametrizace existuje. Momentový a silový rozsah aktuátorů je dán jeho konstantními limitními hodnotami

$$T_{min}^i \leq \tau_i \leq T_{max}^i, \quad i = 1 \dots n. \quad (4.2)$$

Dále předpokládejme, že existují konstantní omezení na maximální rychlost a zrychlení jednotlivých aktuátorů

$$|\dot{q}_i| \leq V_{max}^i, \quad i = 1 \dots n \quad (4.3)$$

$$A_{min}^i \leq \ddot{q}_i \leq A_{max}^i, \quad i = 1 \dots n \quad (4.4)$$

kteřá mohou vyplývat z dalších požadavků a omezení na konstrukci stroje, pracovní prostor apod. Problém nalezení trajektorie pohybu po dané cestě lze formulovat následujícím způsobem.

Problém 4.1. Nalezněte řídicí sekvenci vstupních sil a momentů $\boldsymbol{\tau}(s)$, která přesune efektor systému (4.1) podél cesty a bude respektovat jeho kino-dynamické vlastnosti a omezení (4.2), (4.3) a (4.4).

4.1.1 KINEMATIKA PODÉL SPECIFIKOVANÉ CESTY V PRACOVNÍM PROSTORU

Uvažujme přímou kinematiku robota $\mathbf{r}(\mathbf{q})$, která vztahuje prostor kloubových souřadnic a pracovní prostor robota. Obecně můžeme přímou kinematickou úlohu zapsat ve tvaru

$$\mathbf{p} = \mathbf{r}(\mathbf{q}): Q \rightarrow X, \quad (4.5)$$

kde \mathbf{q} jsou zobecněné kloubové souřadnice stroje, \mathbf{p} jsou souřadnice koncového efektoru nebo klíčové části stroje v souřadnicích pracovního prostoru.

Pomocí Jakobiánu vektorové funkce $\frac{\partial \mathbf{r}}{\partial \mathbf{q}}$ dostáváme rychlostní závislost

$$\dot{\mathbf{p}} = \frac{\partial \mathbf{r}}{\partial \mathbf{q}} \dot{\mathbf{q}}. \quad (4.6)$$

Využitím Hessiánu vektorové funkce $\frac{\partial^2 \mathbf{r}}{\partial \mathbf{q}^2}$ dostáváme závislost zrychlení ve tvaru

$$\ddot{\mathbf{p}} = \dot{\mathbf{q}}^T \frac{\partial^2 \mathbf{r}}{\partial \mathbf{q}^2} \dot{\mathbf{q}} + \frac{\partial \mathbf{r}}{\partial \mathbf{q}} \ddot{\mathbf{q}}. \quad (4.7)$$

Jestliže je cesta v konfiguračním nebo pracovním prostoru stroje přirozeně parametrizovatelná, potom můžeme vyjádřit závislosti rychlostí a zrychlení na přirozené parametrizaci cesty s . Kinematika podél cesty je tedy parametrizována skalárním parametrem s , jež definuje pohyb po křivce.

Parametrizací přímé kinematické úlohy v pracovním prostoru X dostaneme

$$\mathbf{p}(s) = \mathbf{r}(q). \quad (4.8)$$

Postupným derivováním dostaneme závislosti rychlostí a zrychlení

$$\frac{\partial \mathbf{p}}{\partial s} \dot{s} = \frac{\partial \mathbf{r}}{\partial \mathbf{q}} \dot{\mathbf{q}} \quad (4.9)$$

$$\frac{\partial^2 \mathbf{p}}{\partial s^2} \dot{s}^2 + \frac{\partial \mathbf{p}}{\partial s} \ddot{s} = \dot{\mathbf{q}}^T \frac{\partial^2 \mathbf{r}}{\partial \mathbf{q}^2} \dot{\mathbf{q}} + \frac{\partial \mathbf{r}}{\partial \mathbf{q}} \ddot{\mathbf{q}}, \quad (4.10)$$

kde $\frac{\partial^2 \mathbf{r}}{\partial \mathbf{q}^2}$ je Hessian vektorové funkce.

Vyjádřením rychlosti kloubových souřadnic z (4.9) dostaneme zobrazení $(s, \dot{s}) \rightarrow \dot{\mathbf{q}}(s, \dot{s})$

$$\dot{\mathbf{q}} = \left(\frac{\partial \mathbf{r}}{\partial \mathbf{q}} \right)^{-1} \frac{\partial \mathbf{p}}{\partial s} \dot{s}. \quad (4.11)$$

Podobně pro zrychlení kloubových souřadnic dostaneme z (4.10), s využitím (4.11), zobrazení $(s, \dot{s}, \ddot{s}) \rightarrow \ddot{\mathbf{q}}(s, \dot{s}, \ddot{s})$.

$$\ddot{\mathbf{q}} = \left(\frac{\partial \mathbf{r}}{\partial \mathbf{q}} \right)^{-1} \left(\frac{\partial \mathbf{p}}{\partial s} \right) \ddot{s} + \left(\frac{\partial \mathbf{r}}{\partial \mathbf{q}} \right)^{-1} \left[\frac{\partial^2 \mathbf{p}}{\partial s^2} - \left[\left(\frac{\partial \mathbf{r}}{\partial \mathbf{q}} \right)^{-1} \left(\frac{\partial \mathbf{p}}{\partial s} \right) \right]^T \frac{\partial^2 \mathbf{r}}{\partial \mathbf{q}^2} \left[\left(\frac{\partial \mathbf{r}}{\partial \mathbf{q}} \right)^{-1} \left(\frac{\partial \mathbf{p}}{\partial s} \right) \right] \right] \dot{s}^2 \quad (4.12)$$

Pro lepší orientaci označíme některé členy

$$\left(\frac{\partial \mathbf{r}}{\partial \mathbf{q}} \right)^{-1} = \mathbf{I}r_q, \quad \left(\frac{\partial \mathbf{p}}{\partial s} \right) = p_s, \quad \frac{\partial^2 \mathbf{p}}{\partial s^2} = p_{ss}, \quad \frac{\partial^2 \mathbf{r}}{\partial \mathbf{q}^2} = r_{qq}$$

Potom vztahy (4.11) a (4.12) přejdou na tvar

$$\dot{\mathbf{q}} = \mathbf{I}r_q p_s \dot{s} \quad (4.13)$$

$$\ddot{\mathbf{q}} = \mathbf{I}r_q p_s \ddot{s} + \mathbf{I}r_q \left[p_{ss} - [\mathbf{I}r_q p_s]^T r_{qq} [\mathbf{I}r_q p_s] \right] \dot{s}^2, \quad (4.14)$$

který dále přeznačením $\mathbf{G}(s) = \mathbf{I}r_q p_s$, $\mathbf{K}(s) = \mathbf{I}r_q \left[p_{ss} - [\mathbf{I}r_q p_s]^T r_{qq} [\mathbf{I}r_q p_s] \right]$ zjednodušíme na

$$\dot{\mathbf{q}} = \mathbf{G}(s) \dot{s}, \quad (4.15)$$

$$\ddot{\mathbf{q}} = \mathbf{G}(s) \ddot{s} + \mathbf{K}(s) \dot{s}^2. \quad (4.16)$$

Dostáváme závislosti zobecněných souřadnic na skalárním parametru s podél specifikované cesty definované v pracovním prostoru X .

4.1.2 KINEMATIKA PODÉL SPECIFIKOVANÉ CESTY V KONFIGURAČNÍM PROSTORU

Podobně bude vypadat i parametrizace v konfiguračním prostoru kloubových souřadnic Q . Uvažujme, že existuje parametrizace $\mathbf{q} = \mathbf{f}(s)$. Potom existují vztahy pro rychlostní závislost a závislost zrychlení ve tvaru

$$\dot{\mathbf{q}} = \frac{\partial \mathbf{f}}{\partial s} \dot{s}, \quad (4.17)$$

$$\ddot{\mathbf{q}} = \frac{\partial^2 \mathbf{f}}{\partial s^2} \dot{s}^2 + \frac{\partial \mathbf{f}}{\partial s} \ddot{s}. \quad (4.18)$$

Dostáváme závislosti zobecněných souřadnic na skalárním parametru s podél specifikované cesty definované v konfiguračním prostoru Q . Můžeme samozřejmě dosadit zpět do (4.6) a (4.7) a získat přímou kinematiku v pracovním prostoru

$$\mathbf{p} = \mathbf{r}(\mathbf{f}(s)), \quad (4.19)$$

$$\dot{\mathbf{p}} = \frac{\partial \mathbf{r}}{\partial \mathbf{q}} \frac{\partial \mathbf{f}}{\partial s} \dot{s}, \quad (4.20)$$

$$\ddot{\mathbf{p}} = \left[\left(\frac{\partial \mathbf{f}}{\partial s} \right)^T \frac{\partial^2 \mathbf{r}}{\partial \mathbf{q}^2} \frac{\partial \mathbf{f}}{\partial s} + \frac{\partial \mathbf{r}}{\partial \mathbf{q}} \frac{\partial^2 \mathbf{f}}{\partial s^2} \right] \dot{s}^2 + \frac{\partial \mathbf{r}}{\partial \mathbf{q}} \frac{\partial \mathbf{f}}{\partial s} \ddot{s}. \quad (4.21)$$

4.1.3 DYNAMIKA SYSTÉMU PODÉL SPECIFIKOVANÉ CESTY

Pomocí (4.17),(4.18) lze vyjádřit pohybové rovnice systému parametrizované podél specifikované cesty v konfiguračním prostoru. Dosazením do (4.1) dostaneme

$$M(\mathbf{f}(s)) \frac{\partial \mathbf{f}}{\partial s} \ddot{s} + \left[M(\mathbf{f}(s)) \frac{\partial^2 \mathbf{f}}{\partial s^2} + \left(\frac{\partial \mathbf{f}}{\partial s} \right)^T C(\mathbf{f}(s)) \frac{\partial \mathbf{f}}{\partial s} \right] \dot{s}^2 + \mathbf{g}(\mathbf{f}(s)) = \boldsymbol{\tau} \quad (4.22)$$

↓ přeznačením získáme

$$\mathbf{M}_Q(s) \ddot{s} + \mathbf{C}_Q(s) \dot{s}^2 + \mathbf{g}_Q(s) = \boldsymbol{\tau}.$$

Pomocí (4.15),(4.16) lze vyjádřit pohybové rovnice systému parametrizované podél specifikované cesty v pracovním prostoru. Opět dosazením do (4.1)

$$M \mathbf{I} r_q p_s \ddot{s} + \left(M \mathbf{I} r_q [p_{ss} - [\mathbf{I} r_q p_s]^T r_{qq} [\mathbf{I} r_q p_s]] + [\mathbf{I} r_q p_s]^T C \mathbf{I} r_q p_s \right) \dot{s}^2 + \mathbf{g}(r^{-1}(\mathbf{p}(s))) = \boldsymbol{\tau} \quad (4.23)$$

a substitucí

$$\begin{aligned} \mathbf{M}_X(s) &= M \mathbf{G}(s) \\ \mathbf{C}_X(s) &= M \mathbf{K}(s) + \mathbf{G}(s)^T C \mathbf{G}(s) \\ \mathbf{g}_X(s) &= \mathbf{g}(r^{-1}(\mathbf{p}(s))) \end{aligned} \quad (4.24)$$

dostaneme pohybové rovnice systému parametrizované podél specifikované cesty v pracovním prostoru skalárním parametrem s

$$\mathbf{M}_X(s) \ddot{s} + \mathbf{C}_X(s) \dot{s}^2 + \mathbf{g}_X(s) = \boldsymbol{\tau}. \quad (4.25)$$

Obecný zápis pohybových rovnic systému podél specifikované cesty budeme uvažovat ve tvaru

$$\mathbf{M}(s) \ddot{s} + \mathbf{C}(s) \dot{s}^2 + \mathbf{g}(s) = \boldsymbol{\tau}. \quad (4.26)$$

Poznámka. Všimněme si, že pro netlumený systém, popsany pohybovými rovnicemi (4.26), dostáváme lineární rovnice vzhledem k (\ddot{s}, \dot{s}^2) , parametrizované podél cesty parametrem s . Je tedy možné, pro danou polohu s , kreslit křivky momentových omezení ve tvaru přímek

$$M_i \ddot{s} + C_i \dot{s}^2 + g_i = T_{min}^i, \quad M_i \ddot{s} + C_i \dot{s}^2 + g_i = T_{max}^i, \quad i = 1 \dots n. \quad (4.27)$$

Ty budou v rovině (\ddot{s}, \dot{s}^2) definovat mnohoúhelník, ve kterém se musí pohybovat stav systému, aby nedošlo k porušení omezení.

4.1.4 OMEZENÍ A JEJICH VLIV NA PROBLÉM PLÁNOVÁNÍ

V předchozí kapitole byl odvozen vztah pro kinematiku a dynamiku systému parametrizovanou podél specifikované cesty. Využitím vztahů (4.15),(4.16) a (4.26) zjistíme, jaký vliv má pohyb podél cesty na tvar omezení (4.2),(4.3) a (4.4). V důsledku kinematických a dynamických transformací omezení nezůstanou konstantní, což bylo naznačeno na motivačním příkladu v úvodu. Pro jednoduchost bude ve vzorcích vypuštěna explicitní závislost na s .

DYNAMICKÁ ZÁVISLOST OMEZENÍ

Z (4.26) a (4.2) dostaneme

$$T_{min}^i \leq M_i \ddot{s} + C_i \dot{s}^2 + g_i \leq T_{max}^i, \quad i = 1..n$$

a můžeme omezení na zrychlení přeformulovat do tvaru

$$L_i(s, \dot{s}) \leq \ddot{s} \leq U_i(s, \dot{s}), \quad (4.28)$$

kde

$$L_i(s, \dot{s}) = \begin{cases} (T_{min}^i - C_i \dot{s}^2 - g_i) / M_i, & \text{if } M_i > 0 \\ (T_{max}^i - C_i \dot{s}^2 - g_i) / M_i, & \text{if } M_i < 0 \end{cases} \quad (4.29)$$

a

$$U_i(s, \dot{s}) = \begin{cases} (T_{max}^i - C_i \dot{s}^2 - g_i) / M_i, & \text{if } M_i > 0 \\ (T_{min}^i - C_i \dot{s}^2 - g_i) / M_i, & \text{if } M_i < 0 \end{cases} \quad (4.30)$$

Jestliže existují na cestě body pro které platí $M_i = 0$ pro nějaké i , potom omezující rovnice přejdou na tvar

$$T_{min}^i \leq C_i(s) \dot{s}^2 + g_i(s) \leq T_{max}^i \quad (4.31)$$

a tvoří *kritické body* omezující rychlost

$$\dot{s}^2 \leq V_i^\tau(s), \quad (4.32)$$

kde

$$V_i^\tau(s) = \begin{cases} (T_{max}^i - g_i) / C_i, & \text{if } C_i > 0 \\ (T_{min}^i - g_i) / C_i, & \text{if } C_i < 0 \end{cases} \quad (4.33)$$

KINEMATICKÁ ZÁVISLOST OMEZENÍ

Z kinematiky lze podobně získat další nerovnice pro omezující křivky. Z (4.16) a (4.4) dostaneme

$$A_{min}^i \leq G_i \ddot{s} + K_i \dot{s}^2 \leq A_{max}^i, \quad i = 1..n, \quad (4.34)$$

$$D_i(s, \dot{s}) \leq \ddot{s} \leq H_i(s, \dot{s}), \quad (4.35)$$

kde

$$D_i(s, \dot{s}) = \begin{cases} (A_{min}^i - K_i \dot{s}^2) / G_i, & \text{if } G_i > 0 \\ (A_{max}^i - K_i \dot{s}^2) / G_i, & \text{if } G_i < 0 \end{cases} \quad (4.36)$$

a

$$H_i(s, \dot{s}) = \begin{cases} (A_{max}^i - K_i \dot{s}^2) / G_i, & \text{if } G_i > 0 \\ (A_{min}^i - K_i \dot{s}^2) / G_i, & \text{if } G_i < 0 \end{cases} \quad (4.37)$$

Podobně, jestliže existují na cestě body pro které platí $G_i = 0$ pro nějaké i , potom omezující rovnice (4.34) přejdou na tvar

$$A_{min}^i \leq K_i \dot{s}^2 \leq A_{max}^i, \quad i = 1..n \quad (4.38)$$

a tvoří *kritické body* omezující rychlost

$$\dot{s}^2 \leq V_i^a(s), \quad (4.39)$$

kde

$$V_i^a(s) = \begin{cases} A_{max}^i / K_i, & \text{if } K_i > 0 \\ A_{min}^i / K_i, & \text{if } K_i < 0 \end{cases} \quad (4.40)$$

Poslední rovnice omezení na rychlost podél cesty plynou přímo z (4.3) a (4.15)

$$\dot{s}^2 \leq V_i^v(s), \quad (4.41)$$

kde

$$V_i^v(s) = (V_{max}^i / G_i)^2, \quad \text{if } G_i \neq 0. \quad (4.42)$$

KINO-DYNAMICKÉ OMEZENÍ

Limitní křivky zrychlení jsou potom dány minimem/maximem omezujících křivek (4.29),(4.30) a (4.36), (4.37)

$$\begin{aligned} DM(s, \dot{s}) &= \max\{L_i(s, \dot{s}), D_i(s, \dot{s})\} \\ AM(s, \dot{s}) &= \min\{U_i(s, \dot{s}), H_i(s, \dot{s})\} \end{aligned} \quad (4.43)$$

Ze vztahu (4.28),(3.35) a (4.43) vyplývá ještě jedno důležité omezení na rychlost. Musí platit, že $DM(s, \dot{s}) \leq AM(s, \dot{s})$, a tedy bude existovat limitní křivka v prostoru (s, \dot{s}) , pro kterou bude platit

$$V_i^{AD}(s): DM(s, \dot{s}) = AM(s, \dot{s}). \quad (4.44)$$

Výsledná *limitní křivka rychlosti* je potom minimem omezujících křivek (4.33),(4.40),(4.44), a (4.42)

$$VM(s) = \sqrt{\min\{V_i^{AD}(s), V_i^r(s), V_i^a(s), V_i^v(s)\}}. \quad (4.45)$$

Poznámka. Jestliže se stav systému nachází na $VM(s)$ a strmost této křivky je nižší než strmost $AM(s, \dot{s}), DM(s, \dot{s})$, potom není možné použít mezní hodnoty zrychlení jako vstup systému. Takový bod je právě *singulárním bodem*, který byl již popsán v kapitole 3. Omezení na zrychlení je zcela specifikováno strmostí limitní křivky rychlosti $VM(s)$.

4.2 PARAMETRICKÉ PLÁNOVÁNÍ S VYLOUČENÍM ČASU

V kapitole 3 jsme se zabývali časově optimálním plánováním přesunu systému s konstantními omezeními. Jednalo se o plánování takových úloh, které lze na tento problém převést, jako například pohyb kolejových vozidel s po částech konstantními omezeními, řízení výtahů, dopravníků apod. Plánovací algoritmus hledal řešení v časové oblasti. V případě obecných robotických systémů je možné určitým způsobem přeformulovat úlohu plánování a získat vstupní parametry pro již dříve zmíněné techniky.

Standardní formulací t-optimální úlohy v robotice je požadavek nalezení minima funkcionálu (3.3) podél cesty systému, definované například sekvencí poloh efektoru systému v pracovním prostoru stroje a časových značek, $p(t)$. Mnohdy je ale cesta vyjádřena bez časových značek a lze ji pouze parametrizovat skalárním parametrem s , $p(s)$. Časové značky jsou neznámou a vyplynou až z řešení úlohy plánování pohybu podél této cesty. Potom je užitečné přeformulovat kritérium (3.3) následujícím způsobem.

Předpokládejme, že s je přirozenou parametrizací délky cesty. Potom můžeme vyjádřit časovou změnu jako $dt = 1/\dot{s} ds$ a integrací obou stran rovnice dostaneme závislost času na rychlosti podél cesty

$$t - t_0 = \int_{s_0}^s \frac{1}{\dot{s}} ds. \quad (4.46)$$

V t-optimálním řízení hledáme tedy takovou křivku $1/\dot{s}(s)$, jejíž integrál nabude nejnižší možné hodnoty. Aby toto kritérium šlo vyčíslit i pro hodnotu $\dot{s} = 0$, je užitečné přeformulovat kritérium t-optimality a hledat maximum převrácené hodnoty kritéria

$$\min_{u(\cdot)} \int_0^{t_f} 1 dt \Rightarrow \max_{u(\cdot)} \int_{s_0}^{s_f} \dot{s} ds. \quad (4.47)$$

Znamená to, že hledáme takové řízení, které ve fázové rovině (s, \dot{s}) bude maximalizovat plochu pod křivkou stavu. Problém 4.1 pak můžeme přeformulovat do tvaru

Problém 4.2. Minimalizujte čas, který potřebuje systém

$$\ddot{s} = u(t) \quad (4.48)$$

k přesunu z počátečního do koncového stavu podléhajícího omezením

$$DM(s, \dot{s}) \leq \dot{s} \leq AM(s, \dot{s}), \quad (4.49)$$

$$\dot{s} \leq VM(s), \quad (4.50)$$

definovaným v kapitole 4.1.4.

Možnost převedení problému 4.1 na problém 4.2 je klíčová pro využití generátorů pohybu založených na řetězci integrátorů a t-optimální strategii plánování. To umožňuje pro diskretizovaná omezení plánovat pohyb bez nutnosti numerického řešení pohybových rovnic systému, a tím tento proces plánování mnohonásobně urychlit, mnohdy až k výpočetní hranici reálného času.

4.3 PLÁNOVÁNÍ PODÉL SPECIFIKOVANÉ CESTY SYSTÉMEM DRUHÉHO ŘÁDU

Jestliže nám úloha dovoluje skokově měnit zrychlení systému, potom je možné použít řetězec dvou integrátorů jako generátoru trajektorií pro polohu a rychlost systému podél specifikované cesty. Například pro nejrychlejší přesun systému mezi požadovanými stavy je nutné využít plný výkon aktuátorů. Zrychlení je přímo úměrné vstupujícímu proudu do akčního členu, proto je v případě t-optimality nutné ho skokově měnit mezi mezními hodnotami. Řetězec dvou integrátorů s omezeným vstupem je tedy nejnižší struktura generátoru trajektorie. Další využití generátoru 2.řádu může nastat u systémů, které obsahují suché tření. Zde nastává například problém s rozjezdem, kde v případě pomalého náběhu vstupní síly dochází k nespecifikovanému rozjezdu až po překonání tření. I přes hladké průběhy žádaných veličin dochází ke skokovým změnám v nespecifikovaných intervalech. Proto je spíše vhodné působit přímo skokovým zrychlením a alespoň specifikovat kdy a za jakých podmínek k rozjezdu a zastavení dojde.

4.3.1 ZÁKLADNÍ POPIS A DEFINICE ÚLOHY

Dána struktura robota nebo stroje, přímá i zpětná kinematická úloha a dynamický popis. Nechť je dána i cesta pohybu stroje nebo koncového efektoru robota, a tato cesta je dostatečně hladkou křivkou, aby ji bylo možné projet bez zastavení¹⁰. Předpokládejme, že trajektorie přesunu systému po dané cestě existuje. Neboli, že omezení jsou taková, že je možné nalézt křivky polohy, rychlosti a zrychlení, které převádí počáteční stav do koncového stavu a respektují daná omezení na celém rozsahu pohybu. Dále bez ztráty obecnosti předpokládáme takové počáteční a koncové podmínky, které zajistí, že vývoj stavu generátoru bude v kladných hodnotách rychlosti. Úkolem generátoru pohybu je naplánovat průjezd po dané cestě tak, aby pohyb respektoval všechna omezení, případně minimalizoval dané kritérium kvality. Dostáváme tak formulaci problému 4.2.

Následující text řeší výše zmíněný problém pro tři základní závislosti omezení na parametru s a \dot{s} .

4.3.2 PLÁNOVACÍ ALGORITMUS – TRIANGLE ALGORITHM

Předpokládejme, že úloha vede na plánování s konstantními omezeními na zrychlení a s proměnným omezením na rychlost podél cesty ($VM(s)$, AM , DM). Uvažujme řetězec dvou integrátorů jako generátor křivek polohy, rychlosti a zrychlení. Ve zjednodušeném zápisu dostaneme popis generátoru

$$\begin{aligned}\dot{s} &= v \\ \dot{v} &= u,\end{aligned}\tag{4.51}$$

s omezeními na stavy ve tvaru

$$\begin{aligned}s_{Min} &\leq s \leq s_{Max} \\ -VM(s) &\leq v(s) \leq VM(s) \\ -DM &\leq u \leq AM\end{aligned}\tag{4.52}$$

kde vstup u budeme podle potřeby také značit písmenem a a nazývat ho zrychlením. Dále předpokládáme, že omezení $VM(s)$ je definováno posloupností bodů podél cesty, získaných například vhodnou diskretizací. Dostáváme stejné zadání úlohy jako v kapitole 3.4 s tím rozdílem, že omezení na rychlost podél cesty není konstantní.

¹⁰ V opačném případě by bylo nutné cestu rozdělit na spojité úseky a naplánovat je odděleně s požadavkem shodných navazujících počátečních a koncových stavů.

TRANSFORMACE STAVOVÉHO PROSTORU GENERÁTORU

Protože budeme plánování pohybu provádět ve fázovém prostoru, je třeba zjistit, jak se změni vztahy pro popis pohybu systému (4.51), jestliže přejdeme z prostoru časového do prostoru fázového. Integrací první a druhé diferenciální rovnice (4.51)

$$\begin{aligned} v &= v_0 + ut \\ s &= s_0 + v_0 t + u/2 t^2, \quad u = \text{konst.} \end{aligned} \quad (4.53)$$

a vyloučením času dostaneme popis integrálních křivek v prostoru (s, v)

$$v^2 - v_0^2 = 2u(s - s_0), \quad u = \text{konst.}, \quad \text{směrnice} = \frac{u}{v}. \quad (4.54)$$

Z popisu je zřejmé, že systém se bude pro konstantní hodnotu vstupu pohybovat ve fázovém prostoru po parabole a tedy stále měnit svůj směr se směrnicí u/v . Abychom usnadnili výpočet a především další náhled na problém plánování ve fázovém prostoru, zavedeme transformaci

$$\bar{v} = T(v) = v^2, \quad (4.55)$$

kde předpokládáme, že platí $v(s) \geq 0$. Potom se v prostoru (s, \bar{v}) změni tvar integrálních křivek (4.54) na přímky se sklonem $2u$.

$$\bar{v} - \bar{v}_0 = 2u(s - s_0), \quad u = \text{konst.}, \quad \text{směrnice} = 2u, \quad \bar{v}_0 = v_0^2 \quad (4.56)$$

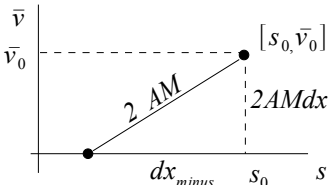
Můžeme tak přeformulovat problém 4.2 do tvaru

Problém 4.3. Předpokládejme, že po celou dobu pohybu systému platí $v(s) \geq 0$. Potom v prostoru (s, \bar{v}) naplánujte takovou křivku rychlosti $\bar{v}(s)$, která bude pod hraniční křivkou $\bar{V}M(s) = VM^2(s)$ s počátkem v bodě $\bar{x}_0 = [s_0, \bar{v}_0]$, $\bar{v}_0 = v_0^2$ a koncem v bodě $\bar{x}_f = [s_f, \bar{v}_f]$, $\bar{v}_f = v_f^2$, jejíž strmost bude v rozsahu $-2DM \leq 2u(s) \leq 2AM$ a tato křivka splňuje kritérium (4.47).

Transformace (4.55) je klíčová pro další možnosti plánování. Její výhodou je, že pro konstantní hodnotu u dostáváme konstantní směr pohybu stavu systému v prostoru (s, \bar{v}) . Je tedy jednoduché a na pohled zřejmé, kterým směrem se bude systém pohybovat po celou dobu působení konstantního vstupu u . Pokud tedy bude například hraniční křivka rychlosti zadána posloupností bodů, potom přímkové spojení těchto bodů v prostoru (s, \bar{v}) může být pro určitou konstantní hodnotu vstupu přímo trajektorií systému.

Shrňme základní vlastnosti transformace, vhodné pro geometrický přístup k plánování pohybu. Uvažujme transformovaný počáteční stav $\bar{x}_0 = [s_0, \bar{v}_0]$. Pro konstantní hodnoty vstupu u dostáváme následující trajektorie systému:

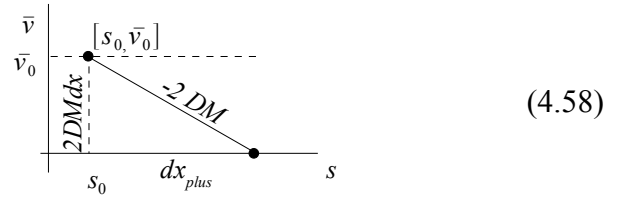
- $u = 0$: Trajektorie systému je přímka rovnoběžná s osou s , vycházející z bodu \bar{x}_0 .
- $u > 0$: Trajektorie systému je přímka s konstantním sklonem $2u$, vycházející z bodu \bar{x}_0 . Nejbližším bodem s nulovou rychlostí, ze kterého je možné dosáhnout stav \bar{x}_0 , je pro maximální zrychlení AM bod

$$[s_0 - dx_{\text{minus}}, 0], \quad dx_{\text{minus}} = \frac{\bar{v}_0}{2AM} \quad (4.57)$$


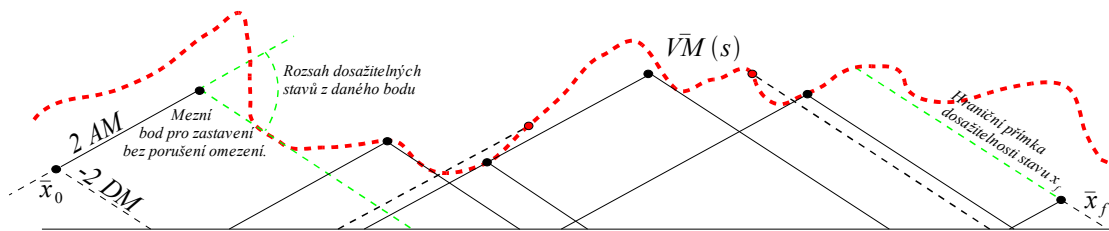
- $u < 0$: Trajektorie systému je přímka s konstantním sklonem $2u$, vycházející z bodu \bar{x}_0 . Nejbližším bodem s nulovou rychlostí, kterého je možné dosáhnout ze stavu \bar{x}_0 , je pro

maximální zpomalení DM bod

$$[s_0 + dx_{plus}, 0], \quad dx_{plus} = \frac{\bar{v}_0}{2DM}$$



Z těchto vlastností vyplývá níže uvedený obrázek 4.2 i název algoritmu. Stav systému v prostoru (s, \bar{v}) musí ležet v každém okamžiku na nebo pod omezující křivkou $V\bar{M}(s)$ a maximální rozběhové a doběhové přímky musí taktéž ležet pod nebo na dotyku omezující křivky. Pokud budeme v každém okamžiku maximalizovat rychlost maximálním možným zrychlením, potom dostaneme t-optimální průběh pro transformovaný systém, a tedy i pro systém původní.



Obrázek 4.2: Základní idea možnosti plánování pohybu v transformovaném prostoru (s, \bar{v}) , včetně naznačení některých vlastností pohybu.

ZÁKLADNÍ STRUKTURA ALGORITMU

Předpokládejme, že omezující křivka $V\bar{M}(s)$ je dána posloupností dvojic bodů (s_i, \bar{v}_i) , $i=1..n$. Na intervalu poloh podél cesty (s_0, s_f) vytvoříme s pomocí vztahů (4.57) a (4.58) dva vektory

$$DX_{minus} = [s_0 - dx(\bar{v}_0)_{minus}, s_j - dx(\bar{v}_j)_{minus}, s_f - dx(\bar{v}_f)_{minus}], \quad j=1..m \quad (4.59)$$

$$DX_{plus} = [s_0 + dx(\bar{v}_0)_{plus}, s_j + dx(\bar{v}_j)_{plus}, s_f + dx(\bar{v}_f)_{plus}], \quad (4.60)$$

kde $s_0 \leq s_j \leq s_f$ a \bar{v}_j je příslušný prvek k s_j .

Z prvního vektoru odstraníme všechny prvky, které by porušovaly vlastnost rostoucí posloupnosti. Takové příslušné body na omezující křivce jsou nedosažitelné z pohledu předchozího stavu, viz. obrázek 4.2. Získaný redukovaný vektor DX_{minus}^r definuje červené body pro vzorovou křivku $V\bar{M}(s)$ na obrázku 4.3. Podobně, ale tentokrát odzadu, odstraníme všechny prvky druhého vektoru, které by porušovaly vlastnost klesající posloupnosti. Takové příslušné body na omezující křivce jsou nedosažitelné z pohledu následujícího stavu. Pro vzorovou křivku $V\bar{M}(s)$ bychom získali redukovaný vektor DX_{plus}^r , který definuje zelené body na obrázku 4.4. Průnikem redukovaných vektorů $DX^r = DX_{minus}^r \cap DX_{plus}^r$ dostaneme příslušné body na omezující křivce $\bar{x} = \{\bar{x}_1, \bar{x}_2, \dots\}$, kde i -tý bod \bar{x}_i je tvořen dvojicí $DX^r(i)$ a příslušnou rychlostí $v(DX^r(i))$. Tyto body jsou dosažitelné přípustným řízením u , obrázek 4.5.

Nyní máme počáteční a koncový stav a sekvenci přípustných bodů na omezující křivce rychlosti, $\{\bar{x}_0, \bar{x}, \bar{x}_f\}$. Protože pracujeme v transformovaném prostoru (s, \bar{v}) , víme, že t-optimální trajektorie bude tvořena vhodným přímkovým spojením těchto bodů. Uvažujme dva body (s, \bar{v}) a $(s^+, \bar{v}(s^+))$ ze sekvence přípustných bodů, kde index “+” znamená, že bod je podle ujeté dráhy dále vpravo vzhledem k bodu (s, \bar{v}) . Pro index “-” bude situace opačná. Jestliže jsou body na omezující křivce přímo sousedními, potom je výsledná trajektorie mezi těmito body přímo přímkovým spojením.

Směrnice

$$2u = \frac{\bar{v}(s^+) - \bar{v}(s)}{s^+ - s} \quad (4.61)$$

a tedy i velikost zrychlení je zaručena předchozím výběrem přípustných bodů na omezující křivce. Jestliže body sousední nejsou, potom je mezi nimi nutné postupovat podle kritéria optimality a snažit se maximalizovat rychlost. Pro bod $(s^-, \bar{v}(s^-))$ určíme směrnicí u^+ podle pravého sousedního bodu z původní omezující křivky $\bar{V}\bar{M}(s)$, maximálně však $2AM$. Pro bod $(s^+, \bar{v}(s^+))$ určíme směrnicí u^- podle levého sousedního bodu z původní omezující křivky $\bar{V}\bar{M}(s)$, minimálně však $-2DM$. Vyřešíme soustavu pro nalezení průsečíku přímk

$$\begin{aligned} (\bar{v}(s^+) - \bar{v}(s)) &= 2u^+(s^+ - s) \\ (\bar{v}(s) - \bar{v}(s^-)) &= 2u^-(s - s^-), \end{aligned} \quad (4.62)$$

kde (s, \bar{v}) je hledaný průsečík. Získáme tak nový bod trajektorie (s, \bar{v}) .

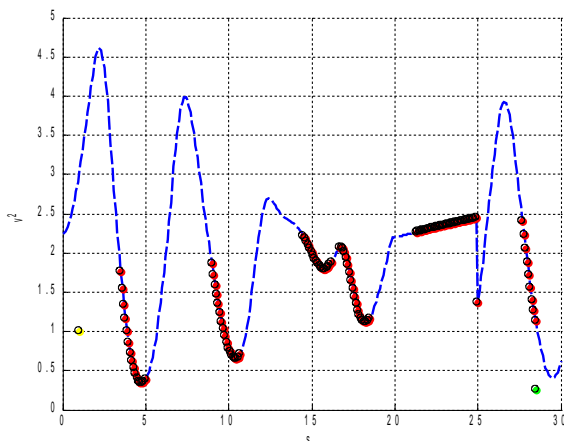
POSTUP ŘEŠENÍ

Rozdělíme problém na tři části a vyřešíme pohyb v těchto částech

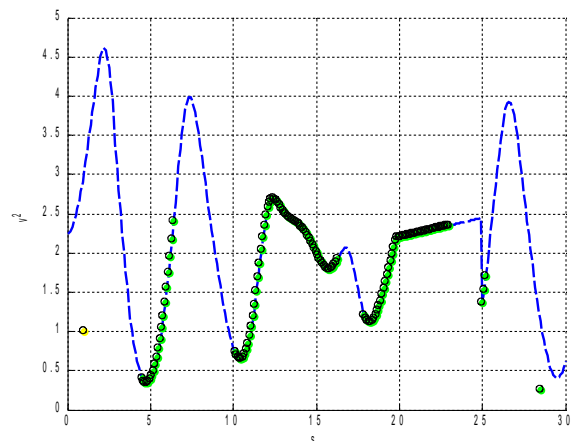
- Nájezd na omezující křivku $\bar{V}\bar{M}(s)$.
Najdeme průsečík rozjezdové přímky (se směrnicí $2AM$) s přímkou, která přichází na první dosažitelný bod omezující křivky se směrnicí danou tímto dosažitelným bodem a bodem předchozím, minimálně však $-2DM$.
- Pohyb po omezující křivce $\bar{V}\bar{M}(s)$.
Postupujeme po omezující křivce podle kritéria optimality, podle vztahů (4.61) nebo (4.62)
- Dojezd do koncového stavu.
Najdeme průsečík dojezdové přímky (se směrnicí $-2DM$) s přímkou, která vychází z posledního dostupného bodu omezující křivky se směrnicí danou tímto bodem a bodem následujícím, maximálně však $2AM$.

Vyřešením pohybu ve všech částech a jejich zřetězením dostáváme výslednou trajektorii stavu systému dvou integrátorů v prostoru (s, \bar{v}) , obrázek 4.6. Zpětnou transformací dostaneme množinu trajektorií pro rychlost a zrychlení vzhledem k poloze na specifikované cestě, $\{v(s), u(s)\}$.

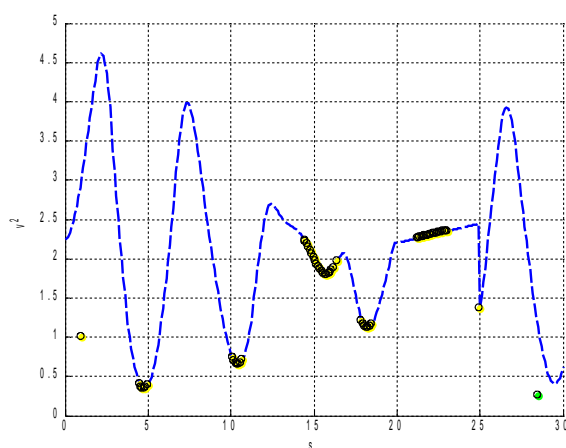
Věta 4.1. *Výše zmíněný postup výpočtu řeší problém 4.3, pro daná omezení splňuje podmínky časové optimality a kritérium (4.47).*



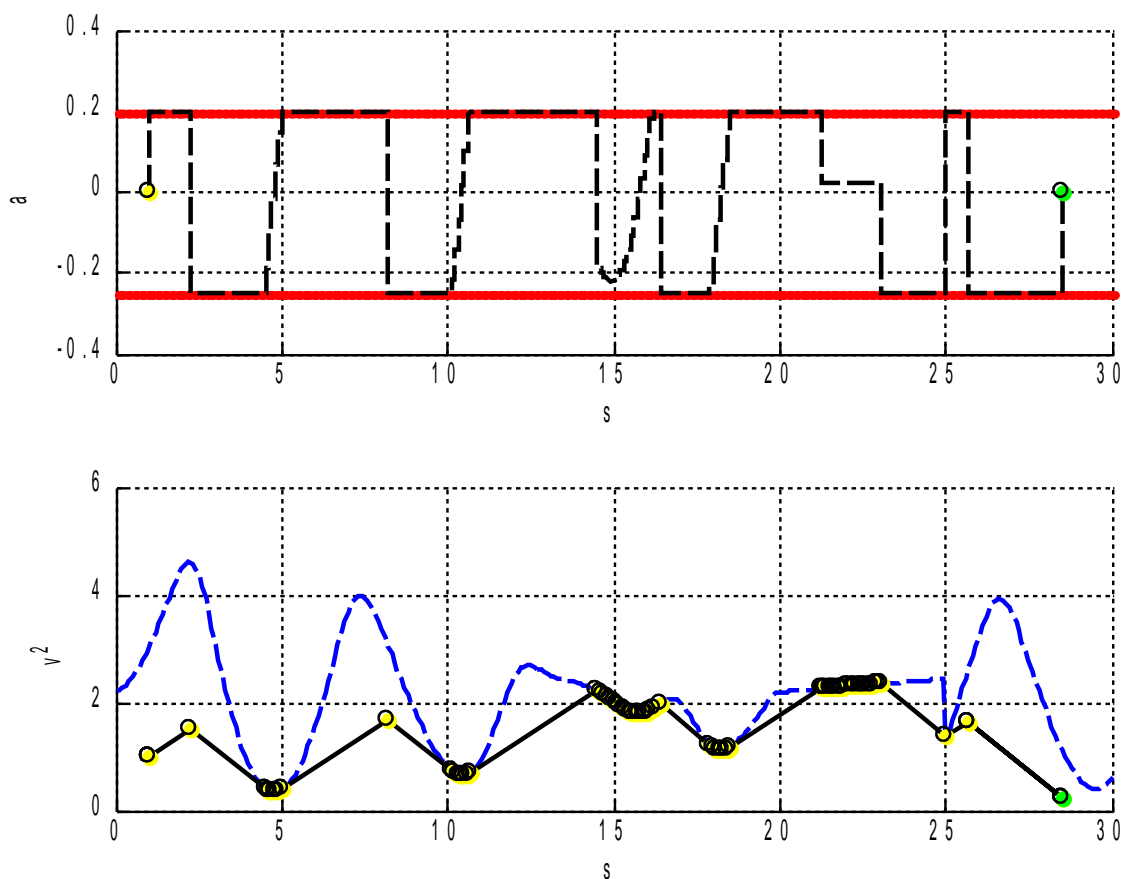
Obrázek 4.3: Příklad nalezení dosažitelných bodů na omezující křivce rychlosti vzhledem k předchozímu stavu.



Obrázek 4.4: Příklad nalezení dosažitelných bodů na omezující křivce rychlosti vzhledem k následujícímu stavu.



Obrázek 4.5: Průnik dosažitelných bodů z obrázku 4.3 a 4.4. Tyto body je možné přípustným řízením dosáhnout a také opustit bez porušení omezení.



Obrázek 4.6: Příklad výsledné trajektorie systému v prostoru (s, \bar{v}) .

4.3.3 PLÁNOVACÍ ALGORITMUS $GAVS(s)$

V případě, že úloha vede na plánování s omezeními, jejichž velikost je závislá na aktuální poloze systému ($VM(s), AM(s), DM(s)$), potom problém 4.3 přejde do tvaru

Problém 4.4. Předpokládejme, že po celou dobu pohybu systému platí $v(s) \geq 0$. Potom v prostoru (s, \bar{v}) naplánujte takovou křivku rychlosti $\bar{v}(s)$, která bude pod hraniční křivkou $V\bar{M}(s) = VM^2(s)$ s počátkem v bodě $\bar{x}_0 = [s_0, \bar{v}_0]$, $\bar{v}_0 = v_0^2$ a koncem v bodě $\bar{x}_f = [s_f, \bar{v}_f]$, $\bar{v}_f = v_f^2$, jejíž strmost bude v povoleném rozsahu $-2DM(s) \leq 2u(s) \leq 2AM(s)$ a tato křivka splňuje kritérium (4.47).

Řešení problému již nelze nalézt pomocí předchozího algoritmu, protože strmost rozjezdových a dojezdových křivek se podél cesty mění. Obecné řešení by bylo možné nalézt přímou integrací pohybových rovnic společně s metodou střelby, jako například v [13],[93]. Přesnost tohoto výpočtu bude nepřímo úměrná výpočetnímu času. Můžeme ale provést diskretizaci omezení *metodou nejhoršího případu* a využít předchozí postupy plánování.

DISKRETIZACE OMEZENÍ

Diskretizací metodou nejhoršího případu se rozumí taková aproximace obecné funkce funkcí po částech konstantní, která leží pod nebo na hranici původní funkce. To v případě diskretizace limitních křivek omezení zajistí, že pohyb po aproximované křivce neporuší daná omezení. Prvním krokem algoritmu plánování je tedy sestavení posloupnosti poloh mezi počátečním a koncovým stavem, mezi kterými jsou daná omezení konstantní

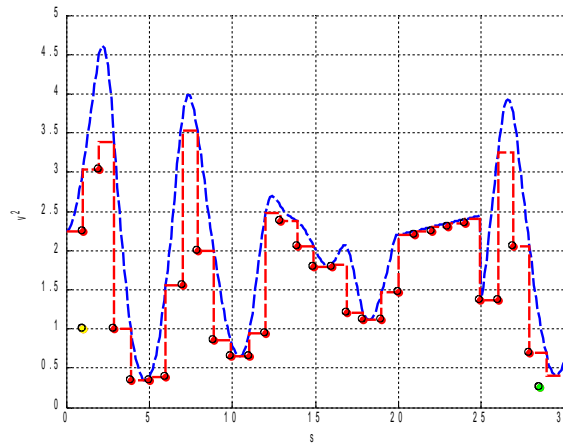
$$\begin{aligned} & [s_0, s_1, s_2, \dots, s_f] \\ & [V\bar{M}_1, V\bar{M}_2, V\bar{M}_3, \dots, V\bar{M}_{sf-1}] \\ & [AM_1, AM_2, AM_3, \dots, AM_{sf-1}] \\ & [DM_1, DM_2, DM_3, \dots, DM_{sf-1}]. \end{aligned} \quad (4.63)$$

POSTUP ŘEŠENÍ

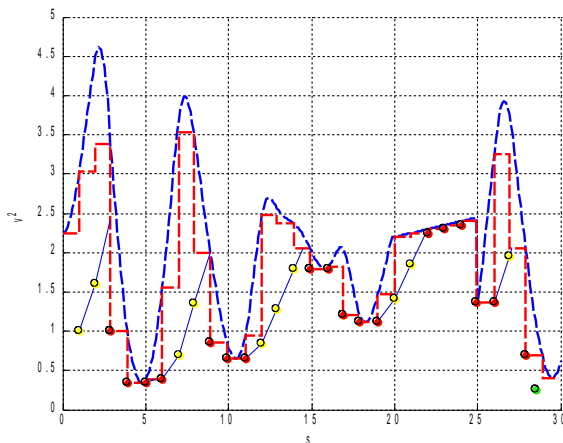
- krok 1: Diskretizace křivek omezení a rozdělení problému na dílčí úseky, ve kterých jsou omezení konstantní.
- krok 2: Nalezení spodní množiny bodů na diskretizované limitní křivce rychlosti. (s_i, \bar{v}_i) , $i=0, 1, 2, \dots, f$, obrázek 4.7.
- krok 3: Procházení množiny z s_0 do s_f a maximalizace rychlosti podle t-optimální strategie: Jestliže je možné zrychlovat na daném úseku i z předešlého dosaženého stavu, potom lze nalézt nový dosažitelný stav. Jestliže je tento stav na hranici úseku vyšší než (s_{i+1}, \bar{v}_{i+1}) , potom přejde výpočet na tento bod, jinak je vytvořen nový mezní bod (s_{ni}, \bar{v}_{ni}) , ze kterého se postupuje dále, obrázek 4.8.
- krok 4: Procházení množiny z s_f do s_0 a maximalizace rychlosti s uvažováním záporného času podle t-optimální strategie: Jestliže je možné zrychlovat z předešlého dosaženého stavu na daném úseku i , potom lze nalézt nový dosažitelný stav. Jestliže je tento stav na hranici úseku vyšší než (s_{i-1}, \bar{v}_{i-1}) , potom přejde výpočet na tento bod, jinak je vytvořen nový mezní bod (s_{ni}, \bar{v}_{ni}) , ze kterého se postupuje dále, obrázek 4.9.
- krok 5: Výsledkem je sekvence bodů (žluté body na obrázku 4.9) na hranicích jednotlivých úseků, jejichž složka rychlosti je maximální možná, kterou lze příslušným řízením dosáhnout. Výpočtem t-optimálního přechodu každého úseku mezi těmito body

s respektováním příslušných omezení (4.63) a jejich zřetěžením dostaneme výslednou trajektorii zrychlení a rychlosti, které přesunou systém podél specifikované cesty, viz obrázek 4.10. Po zpětné transformaci může být tento výpočet proveden *GAVS* algoritmem z kapitoly 3.4.1.

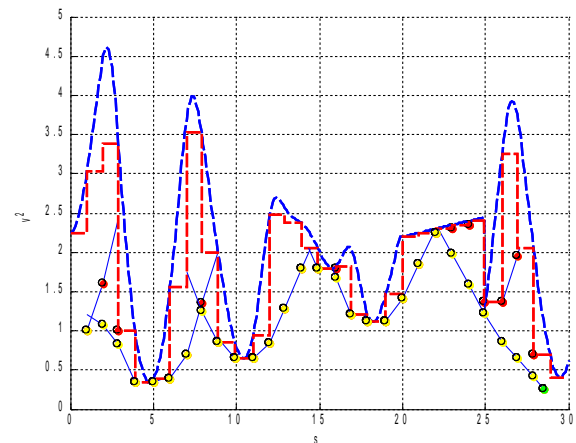
Věta 4.2. *Výše zmíněný postup výpočtu řeší problém 4.4. Pro diskretizovaná omezení splňuje podmínky časové optimality a kritérium (4.47).*



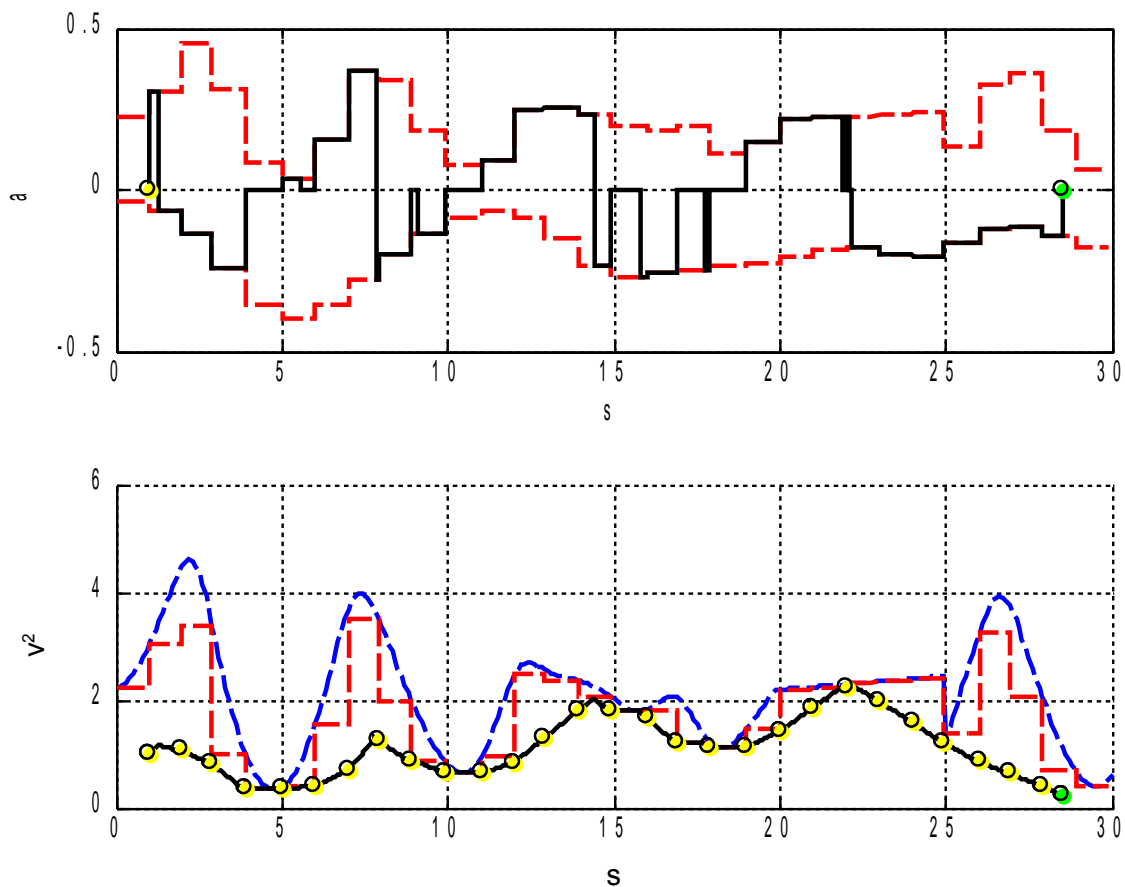
Obrázek 4.7: Nalezení spodní množiny bodů (s_i, \bar{v}_i) .



Obrázek 4.8: Maximalizace rychlosti podle příslušných omezení na zrychlení.



Obrázek 4.9: Minimalizace rychlosti podle příslušných omezení na zrychlení.



Obrázek 4.10: Výsledná trajektorie zrychlení a rychlosti podél specifikované cesty respektující příslušná diskretizovaná omezení.

4.3.4 PLÁNOVACÍ ALGORITMUS $GAVS(s,v)$

Obecně vede úloha řízení pohybu mechatronických systémů na problém plánování s omezeními, jejichž velikost je závislá na aktuální poloze systému a rychlosti ($VM(s)$, $AM(s,v)$, $DM(s,v)$), viz kapitola 4.1. Potom je třeba zobecnit problém 4.3 do tvaru

Problém 4.5. Předpokládejme, že po celou dobu pohybu systému platí $v(s) \geq 0$. Potom v prostoru (s, \bar{v}) naplánujte takovou křivku rychlosti $\bar{v}(s)$, která bude pod hraniční křivkou $\bar{VM}(s) = VM^2(s)$ s počátkem v bodě $\bar{x}_0 = [s_0, \bar{v}_0]$, $\bar{v}_0 = v_0^2$ a koncem v bodě $\bar{x}_f = [s_f, \bar{v}_f]$, $\bar{v}_f = v_f^2$, jejíž strmost bude v povoleném rozsahu $-2DM(s, \dot{s}) \leq 2u(s, \dot{s}) \leq 2AM(s, \dot{s})$ a tato křivka splňuje kritérium (4.47).

Problém 4.5 je opět možné řešit přímou integrací pohybových rovnic společně s metodou střílby. Pokud ale chceme najít rychlé aproximativní řešení, je vhodné využít diskretizace křivek a variet omezení metodou nejhoršího případu a využít předchozí postupy plánování.

DISKRETIZACE OMEZENÍ

Diskretizací v tomto případě dostaneme vektor poloh a pomocný vektor rychlostí

$$\begin{bmatrix} s_0, s_1, s_2, \dots, s_f \\ 0, \bar{d}s_1, \bar{d}s_2, \bar{d}s_3, \dots, \bar{d}s_k \end{bmatrix}, \quad (4.64)$$

kde k je počet vzorků diskretizace a $\bar{d}s_k$ musí být vyšší než $\max(V\bar{M}_i)$.

Vektoru poloh potom přísluší diskretizovaná limitní křivka rychlosti

$$\begin{bmatrix} s_0, s_1, s_2, \dots, s_f \\ V\bar{M}_1, V\bar{M}_2, V\bar{M}_3, \dots, V\bar{M}_{sf-1} \end{bmatrix}. \quad (4.65)$$

Diskretizací limitních variet zrychlení na množině danou vektory (4.64) dostaneme matice

$$\begin{bmatrix} AM_{1,1} & AM_{1,2} & AM_{1,3} & \dots & AM_{1,sf-1} \\ AM_{2,1} & AM_{2,2} & AM_{2,3} & \dots & AM_{2,sf-1} \\ AM_{3,1} & AM_{3,2} & AM_{3,3} & \dots & AM_{3,sf-1} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ AM_{k,1} & AM_{k,2} & AM_{k,3} & \dots & AM_{k,sf-1} \end{bmatrix} \quad (4.66)$$

$$\begin{bmatrix} DM_{1,1} & DM_{1,2} & DM_{1,3} & \dots & DM_{1,sf-1} \\ DM_{2,1} & DM_{2,2} & DM_{2,3} & \dots & DM_{2,sf-1} \\ DM_{3,1} & DM_{3,2} & DM_{3,3} & \dots & DM_{3,sf-1} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ DM_{k,1} & DM_{k,2} & DM_{k,3} & \dots & DM_{k,sf-1} \end{bmatrix} \quad (4.67)$$

Nyní je třeba ještě otestovat platnost vztahu (4.44), případně snížit příslušnou hodnotu $V\bar{M}_i$ tak, aby vztah platil. Následující postup řešení je složitější obdobou předchozího postupu. Již neplatí, že nové uzlové body budou vždy na hranici změny $V\bar{M}_i$, ale mohou být i na hranicích změny omezení zrychlení. To do značné míry zkomplikuje výpočetní algoritmus, protože počet uzlových bodů již není dán délkou vektoru poloh. Princip výpočtu je ale stále stejný.

POSTUP ŘEŠENÍ

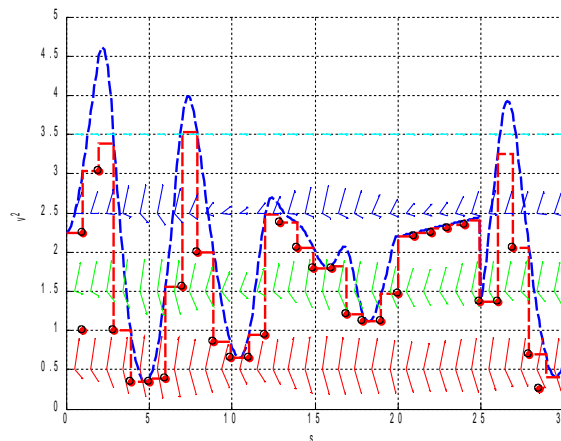
- krok 1: Diskretizace křivek omezení a rozdělení problému na dílčí úseky, ve kterých jsou omezení konstantní.
- krok 2: Nalezení spodní množiny bodů na diskretizované limitní křivce rychlosti. (s_i, v_i) , $i=0, 1, 2, \dots, f$, obrázek 4.11.
- krok 3: Procházení množiny z s_0 do s_f a maximalizace rychlosti podle t-optimální strategie: Jestliže je možné zrychlovat na daném úseku ij z předešlého dosaženého stavu, potom lze nalézt nový dosažitelný stav na hranici změny omezení rychlosti nebo zrychlení. Jestliže je tento stav na hranici úseku vyšší než (s_{i+1}, \bar{v}_{i+1}) , potom přejde výpočet na tento bod, jinak je vytvořen nový mezní bod (s_{ni}, \bar{v}_{ni}) , ze kterého se postupuje dále, obrázek 4.12.
- krok 4: Procházení množiny z s_f do s_0 a maximalizace rychlosti s uvažováním záporného času podle t-optimální strategie: Jestliže je možné zrychlovat z předešlého dosaženého stavu na daném úseku ij , potom lze nalézt nový dosažitelný stav na hranici změny

omezení rychlosti nebo zrychlení. Jestliže je tento stav na hranici úseku vyšší než (s_{i-1}, \bar{v}_{i-1}) , potom přejde výpočet na tento bod, jinak je vytvořen nový mezní bod (s_{ni}, \bar{v}_{ni}) , ze kterého se postupuje dále, obrázek 4.13.

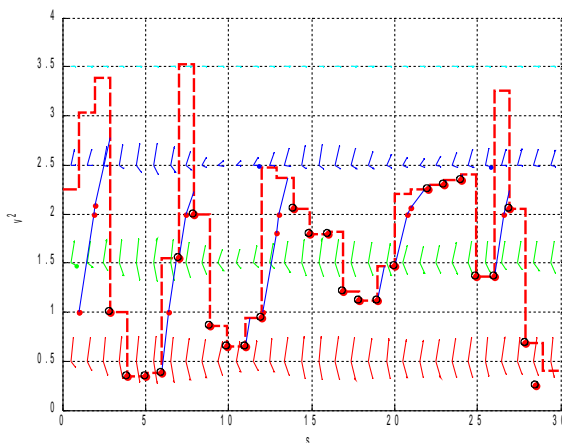
krok 5: Výsledkem je sekvence bodů (žluté body na obrázku 4.13) na hranicích jednotlivých úseků, jejichž složka rychlosti je maximální možná, kterou lze příslušným řízením dosáhnout. Výpočtem t-optimálního přechodu mezi těmito body na každém úseku s respektováním příslušných omezení (4.65),(4.66),(4.67) a jejich zřetězením dostaneme výslednou trajektorii zrychlení a rychlosti, které přesunou systém podél specifikované cesty, viz obrázek 4.14. Po zpětné transformaci může být tento výpočet opět proveden *GAVS* algoritmem z kapitoly 3.4.1.

Věta 4.3. *Výše zmíněný postup výpočtu řeší problém 4.5. Pro diskretizovaná omezení splňuje podmínky časové optimality a kritérium (4.47).*

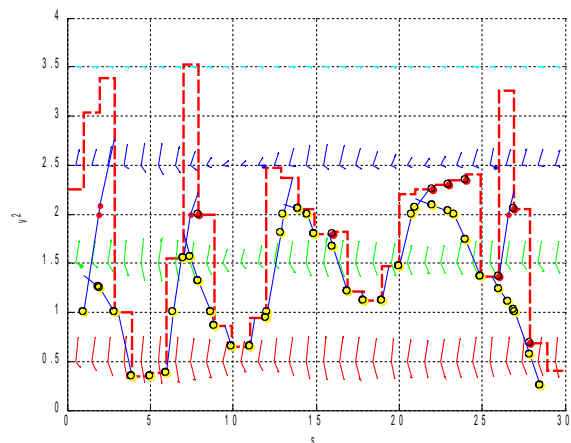
Poznámka. Závěrem je vhodné upozornit, že algoritmus *GAVS* pro výpočet přesunu systému na každém dílčím úseku lze výrazně zjednodušit. Protože se pohybujeme stále v kladných hodnotách rychlosti a vzhledem k systematickému umístění uzlových (mezních) bodů, bude řešení existovat vždy ve strategii $u_+(t)$. Navíc je možné jednotlivý přesun systému v daném úseku spočítat v transformovaném prostoru $(s-\bar{v})$ a ještě tak zjednodušit a zrychlit výpočet.



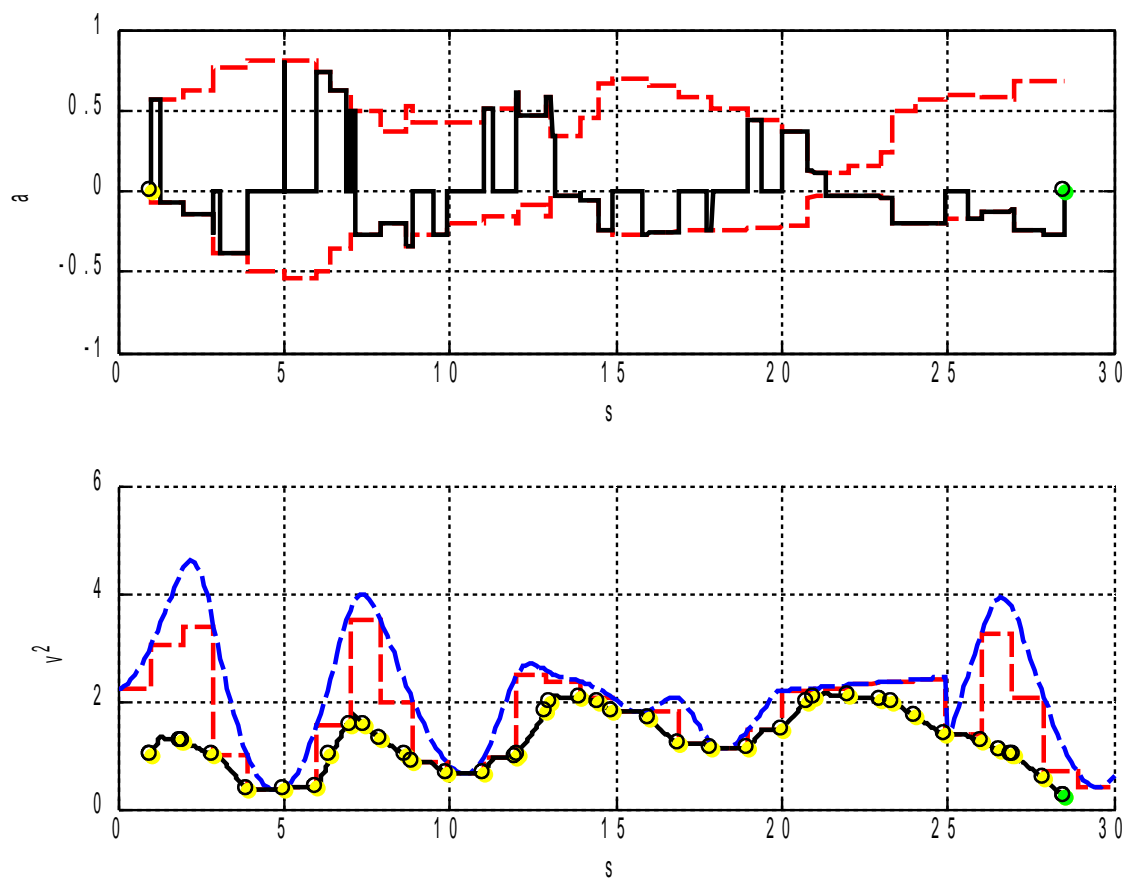
Obrázek 4.11: Diskretizace limitní křivky rychlosti a variety zrychlení (naznačeno směrovými vektory), nalezení bodů (s_i, \bar{v}_i) .



Obrázek 4.12: Maximalizace rychlosti podle příslušných omezení na zrychlení.



Obrázek 4.13: Minimalizace rychlosti podle příslušných omezení na zrychlení.



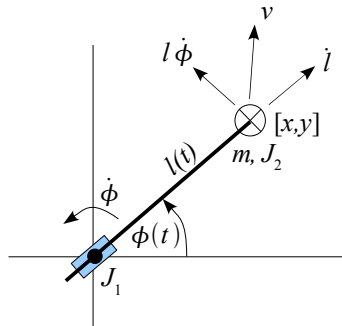
Obrázek 4.14: Výsledná trajektorie zrychlení a rychlosti podél specifikované cesty respektující příslušná diskretizovaná omezení.

4.4 PLANÁRNÍ MANIPULÁTOR - ILUSTRÁČNÍ PŘÍKLAD

Následující příklad shrnuje poznatky čtvrté kapitoly. Jeho cílem je ukázat plánování pohybu podél specifikované cesty pro konkrétní problém s využitím představených metod. Jednoduchost vybraného mechanického systému neubírá nic na obecnosti zvolených postupů řešení. Pouze umožňuje snáze ilustrovat dané metody a porovnat získané výsledky s přirozenou intuicí.

4.4.1 ZADÁNÍ ÚLOHY

Uvažujme planární mechanismus s dvěma stupni volnosti. K pevné inerciální soustavě je připevněn rotační kloub, na něm je pak připevněn prizmatický kloub, zajišťující posuv koncového efektoru ve směru natočení rotačního kloubu. Konfigurační prostor Q je definován jako $[l, \phi] \in S^1 \times \mathbb{R}^1$. Pracovní prostor manipulátoru X je definován na euklidovské rovině \mathbb{R}^2 jako $[x, y] \in \mathbb{R}^2$; $[x, y]^T = \mathbf{r}(l, \phi)$.



Obrázek 4.15: Základní geometrie a parametry manipulátoru.

KINEMATIKA MANIPULÁTORU

Přímé kinematické zobrazení je dáno jako $\mathbf{r} : S^1 \times \mathbb{R}^1 \rightarrow \mathbb{R}^2$; $(l, \phi) \mapsto \mathbf{r}(l, \phi)$,

$$\mathbf{r}(\mathbf{q}) = \begin{bmatrix} l \cos(\phi) \\ l \sin(\phi) \end{bmatrix}, \quad (4.68)$$

kde $\mathbf{q} = [l, \phi]^T$ je vektor zobecněných souřadnic. Vektor rychlosti koncového efektoru lze vyjádřit pomocí jednotkových směrových vektorů jako $\mathbf{v} = \dot{l} \mathbf{e}_l + l \dot{\phi} \mathbf{e}_\phi$. Dále uvažujme, že existují omezení aktuátorů ve formě kinematických omezení na maximální přípustné rychlosti a zrychlení akčních členů

$$|\dot{\phi}| \leq \dot{\phi}_{max}, \quad |\dot{l}| \leq \dot{l}_{max} \quad (4.69)$$

$$|\ddot{\phi}| \leq \ddot{\phi}_{max}, \quad |\ddot{l}| \leq \ddot{l}_{max}. \quad (4.70)$$

DYNAMIKA MANIPULÁTORU

Dynamické chování mechanismu získáme například pomocí lagrangeových rovnic ve tvaru

$$\frac{d}{dt} \frac{\partial L(\dot{\mathbf{q}}, \mathbf{q})}{\partial \dot{\mathbf{q}}} - \frac{\partial L(\dot{\mathbf{q}}, \mathbf{q})}{\partial \mathbf{q}} = \boldsymbol{\tau}, \quad (4.71)$$

kde $\boldsymbol{\tau}$ je vektor vnějšího silového a momentového působení a L je Lagrangian definovaný jako

rozdíl kinetické a potenciální energie

$$L(\dot{\mathbf{q}}, \mathbf{q}) = T(\dot{\mathbf{q}}, \mathbf{q}) - V(\mathbf{q}). \quad (4.72)$$

Uvažujeme rovinný manipulátor, proto vyjádříme pouze kinetické energie jednotlivých prvků mechanismu. Pro rotační kloub dostáváme

$$T_1 = \frac{1}{2} J_1 \dot{\phi}^2. \quad (4.73)$$

Pro prizmatický kloub dostáváme

$$T_2 = \frac{1}{2} J_2 \dot{\phi}^2 + \frac{1}{2} m \mathbf{v}^2 = \frac{1}{2} J_2 \dot{\phi}^2 + \frac{1}{2} m l^2 \dot{\phi}^2 + \frac{1}{2} m \dot{l}^2. \quad (4.74)$$

Celková energie je dána součtem dílčích energií jednotlivých prvků mechanismu. Derivováním a dosazením do (4.71) dostaneme pohybové rovnice mechanismu

$$\begin{aligned} m\ddot{l} - m l \dot{\phi}^2 &= \tau_1 \\ (J_1 + J_2 + m l^2) \ddot{\phi} + 2 m l \dot{l} \dot{\phi} &= \tau_2 \end{aligned} \quad (4.75)$$

kteřé přepíšeme do standardního maticového zápisu (4.1)

$$\begin{bmatrix} m & 0 \\ 0 & J_1 + J_2 + m l^2 \end{bmatrix} \begin{bmatrix} \ddot{l} \\ \ddot{\phi} \end{bmatrix} + \underbrace{\begin{bmatrix} \dot{l} & \dot{\phi} \end{bmatrix} \begin{bmatrix} 0 & 0 \\ 0 & -m l \\ 0 & m l \\ m l & 0 \end{bmatrix}}_{\text{zjednodušený zápis tenzorového součinu } \dot{\mathbf{q}}^T \mathbf{C} \dot{\mathbf{q}}} \begin{bmatrix} \dot{l} \\ \dot{\phi} \end{bmatrix} = \begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix}. \quad (4.76)$$

Předpokládejme, že silový a momentový rozsah aktuátorů je dán jeho konstantními mezními hodnotami ve tvaru

$$\begin{aligned} T_{min}^1 &\leq \tau_1 \leq T_{max}^1 \\ T_{min}^2 &\leq \tau_2 \leq T_{max}^2 \end{aligned} \quad (4.77)$$

PROBLÉM PLÁNOVÁNÍ

Předpokládejme, že manipulátor se má pohybovat po cestě tvaru kružnice se středem v bodě $[x_0, y_0]$ a poloměru r . Úkolem je naplánovat pohyb po této cestě tak, aby nebylo porušeno žádné omezení (4.69), (4.70), (4.77) a přitom byl minimalizován čas přesunu. Druhým úkolem bude nalézt maximální možnou konstantní rychlost pod limitní křivkou rychlosti a porovnat časy jednotlivých přesunů. Grafy budou vykresleny pro následující parametry, definované ve standardních jednotkách SI.

parametry omezení: $-T_{min}^1 = T_{max}^1 = 10, \quad \ddot{l}_{max} = 5, \quad \dot{l}_{max} = 2$
 $-T_{min}^2 = T_{max}^2 = 10, \quad \ddot{\phi}_{max} = 5, \quad \dot{\phi}_{max} = 1.1$

kinematické parametry: $r = 1.1, x_0 = 0.5, y_0 = 1.5, s_0 = 0, v_0 = 0, s_f = 2\pi r, v_f = 0.5$

dynamické parametry: $m = 3, J_1 = 0.1, J_2 = 0.1$

4.4.2 KINEMATIKA MANIPULÁTORU PODÉL SPECIFIKOVANÉ CESTY

POLOHOVÉ ZÁVISLOSTI

Protože specifikovaná cesta je tvaru kružnice, existuje přirozená parametrizace skalárním parametrem s , který definuje posuv po dané trajektorii. Dostáváme zobrazení $s \rightarrow \mathbf{p}(s) = [x(s), y(s)]^T$ ve tvaru

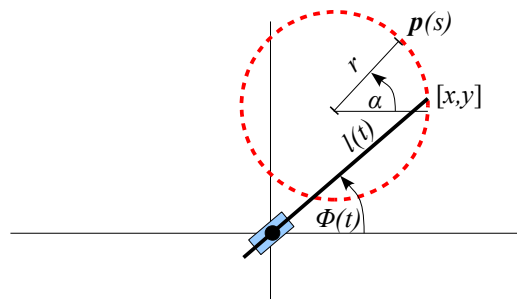
$$\begin{aligned} x(s) &= x_0 + r \cos s/r, & s/r &= \alpha \\ y(s) &= y_0 + r \sin s/r \end{aligned}$$

a přímou kinematickou závislost podle vztahu (4.8) ve tvaru

$$\begin{aligned} l \cos(\phi) &= x_0 + r \cos s/r \\ l \sin(\phi) &= y_0 + r \sin s/r. \end{aligned} \quad (4.78)$$

Vyjádřením závislosti zobecněných souřadnic $\mathbf{q}(s) = [l(s), \phi(s)]^T$ na parametru s dostaneme zobrazení $s \rightarrow \mathbf{q}(s)$ a vztahy

$$\begin{aligned} l(s) &= \sqrt{x_0^2 + y_0^2 + 2rx_0 \cos(s/r) + r^2 + 2ry_0 \sin(s/r)} \\ \operatorname{tg} \phi(s) &= \frac{y_0 + r \sin(s/r)}{x_0 + r \cos(s/r)} \end{aligned} \quad (4.79)$$



Obrázek 4.16: Kinematika manipulátoru a specifikovaná cesta v pracovním prostoru.

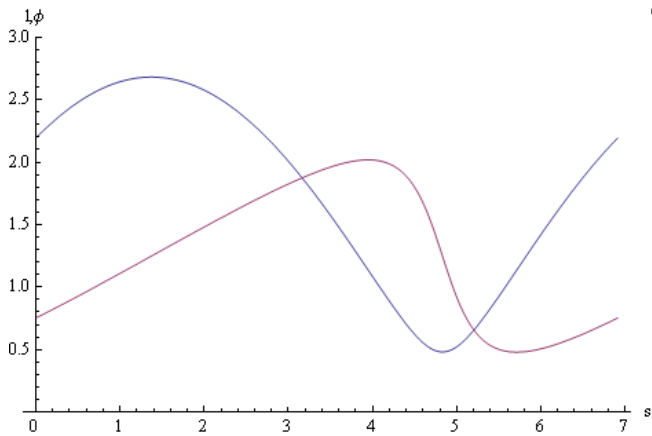
ZÁVISLOSTI RYCHLOSTÍ

Tečná rychlost podél trajektorie je dána vztahem

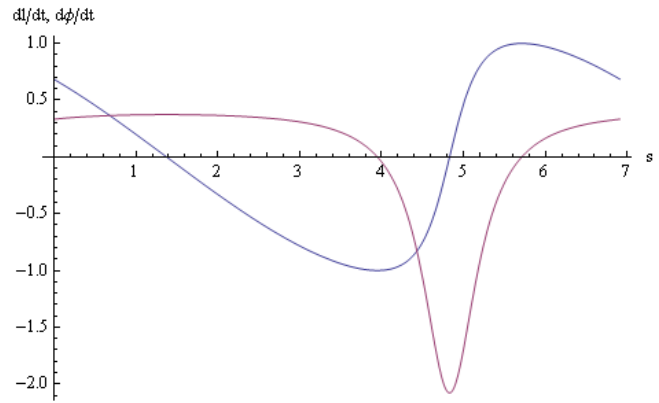
$$\frac{ds}{dt} = \dot{s}, \quad \dot{s} = r \dot{\alpha}$$

Z (4.15) vyjádříme rychlostní závislost na parametrizaci s .

$$\begin{pmatrix} \dot{l} \\ \dot{\phi} \end{pmatrix} = \mathbf{G} \dot{s} = \begin{pmatrix} -\sin(s/r - \phi) \\ \frac{\cos(s/r - \phi)}{l} \end{pmatrix} \dot{s} \quad (4.80)$$



Obrázek 4.18: Průběh zobecněných souřadnic podél specifikované cesty pro dané parametry.



Obrázek 4.17: Diferenciál zobecněných souřadnic podél specifikované cesty.

ZÁVISLOSTI ZRYCHLENÍ

Podobně vyjádříme i závislost zrychlení na parametrizaci podle vztahu (4.16),

$$\ddot{\mathbf{q}} = \mathbf{G} \ddot{s} + \mathbf{K} \dot{s}^2$$

kde

$$\mathbf{G} = \begin{pmatrix} -\sin(s/r - \phi) \\ \frac{\cos(s/r - \phi)}{l} \end{pmatrix}$$

$$\mathbf{K} = \begin{pmatrix} \frac{2l \cos(2\phi) + (1-l) \cos(\frac{2s}{r}) + 2(2-l + (2+l) \cos(2\phi - \frac{2s}{r})) - (1+l) \cos(4\phi - \frac{2s}{r})}{8l^2} & \frac{\cos(\phi - \frac{s}{r})}{r} \\ \frac{\sin(\phi - \frac{s}{r}) \left[-l + (1+l)r \cos(\phi)^3 \cos(\frac{s}{r}) + \frac{1}{2}r(1+3l + (1+l) \cos(2\phi)) \sin(\phi) \sin(\frac{s}{r}) \right]}{l^2 r} & \end{pmatrix}$$

4.4.3 DYNAMIKA MANIPULÁTORU PODÉL SPECIFIKOVANÉ CESTY

Protože je cesta specifikována v pracovním prostoru, využijeme rovnici (4.25) a substituci (4.24),

$$\begin{aligned} \mathbf{M}_x &= \mathbf{M} \mathbf{G} \\ \mathbf{C}_x &= \mathbf{M} \mathbf{K} + \mathbf{G}^T \mathbf{C} \mathbf{G} \end{aligned} \quad (4.81)$$

k získání parametrizovaných pohybových rovnic

$$\mathbf{M}_x \ddot{s} + \mathbf{C}_x \dot{s}^2 = \boldsymbol{\tau}, \quad (4.82)$$

kde \mathbf{M} a \mathbf{C} jsou objekty specifikované v (4.76) a vektory \mathbf{G} , \mathbf{K} jsou specifikovány výše.

4.4.4 SESTROJENÍ LIMITNÍCH KŘIVEK OMEZENÍ

Podle navržených postupů plánování popsaných výše sestrojíme limitní křivky v prostoru generátoru, pod kterými budeme dále plánovat pohyb podél cesty.

Z omezení (4.77) vyplývá následující požadavek

$$\begin{aligned} T_{min}^1 &\leq M_x^1 \ddot{s} + C_x^1 \dot{s}^2 \leq T_{max}^1 \\ T_{min}^2 &\leq M_x^2 \ddot{s} + C_x^2 \dot{s}^2 \leq T_{max}^2. \end{aligned} \quad (4.83)$$

Řešením podle (4.28) dostaneme funkce $L_i(s, \dot{s}), U_i(s, \dot{s})$.

Z omezení (4.70) vyplývá následující požadavek

$$\begin{aligned} -\ddot{l}_{max} &\leq G_1 \ddot{s} + K_1 \dot{s}^2 \leq \ddot{l}_{max} \\ -\ddot{\phi}_{max} &\leq G_2 \ddot{s} + K_2 \dot{s}^2 \leq \ddot{\phi}_{max}. \end{aligned} \quad (4.84)$$

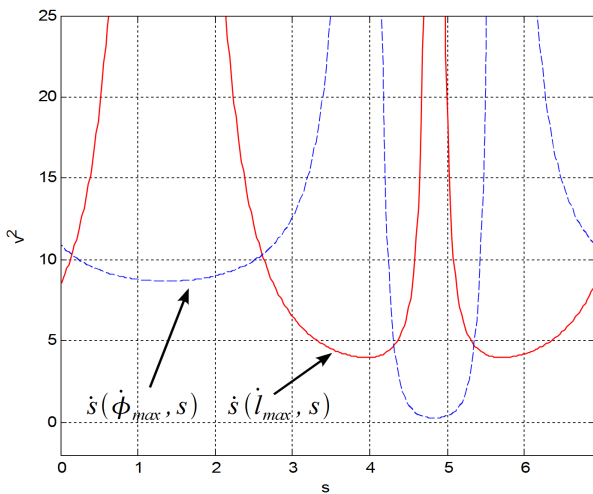
Řešením podle (4.36),(4.37) dostaneme funkce $D_i(s, \dot{s}), H_i(s, \dot{s})$. Z (4.43) potom pro konkrétní parametry dostaneme mezní křivky $AM(s, \dot{s}), DM(s, \dot{s})$ a oblast přípustného zrychlení, obrázek 4.25. Průsečík těchto křivek pak definuje limitní křivku $V_i^{AD}(s): DM(s, \dot{s})=AM(s, \dot{s})$, obrázek 3.22. Z praktického hlediska je rozumné, aby plánování pohybu probíhalo v kladných hodnotách zrychlení a záporných hodnotách brzdění. Proto je limitní křivka $V_i^{AD}(s): DM(s, \dot{s})=AM(s, \dot{s})$ modifikována tak, aby prostor pod ní splňoval následující, tvrdší podmínku

$$V_i^{AD}(s): \min\{DM(s, \dot{s})=AM(s, \dot{s}), DM(s, \dot{s})=0, AM(s, \dot{s})=0\} \quad (4.85)$$

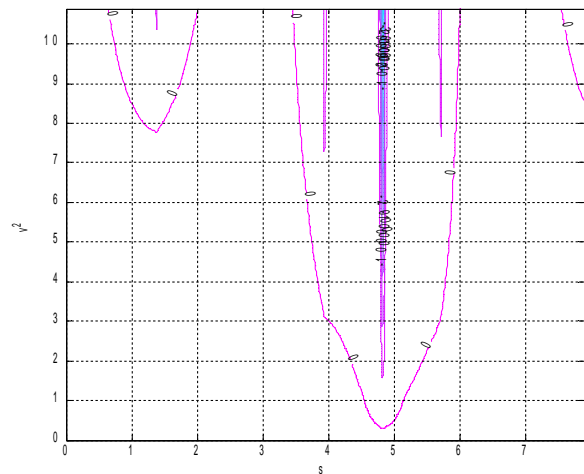
Z omezení (4.69) vyplývá následující požadavek

$$\begin{aligned} \dot{s}^2 &\leq \dot{l}_{max}/G_1 \\ \dot{s}^2 &\leq \dot{\phi}_{max}/G_2 \end{aligned} \quad (4.86)$$

Řešením podle (4.42) dostaneme funkce $V_i^v(s)$. Vyčíslením pro dané parametry dostaneme křivky na obrázku 4.19. Z (4.33) a (4.39) dostaneme ještě funkce $V_i^t(s), V_i^a(s)$ pro singulární body $M_x^i(s)=0, G_i(s)=0$. Obrázek 4.21 ukazuje kvadrát výsledné limitní křivky rychlosti $VM(s)$.



Obrázek 4.19: Zobrazení limitních hodnot rychlosti jednotlivých zobecněných souřadnic podél cesty tvaru kružnice pro dané parametry.



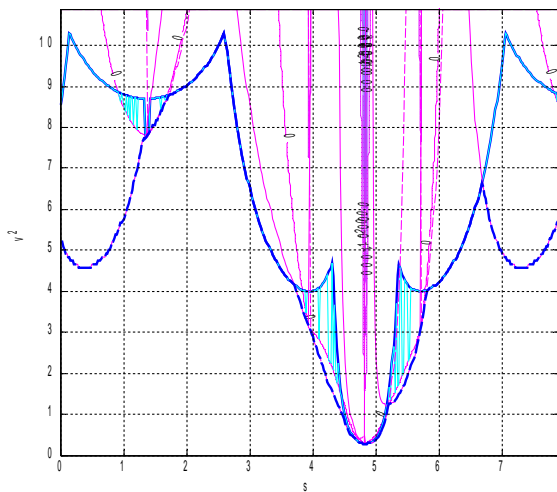
Obrázek 4.20: Limitní křivka $V_i^{AD}(s)$.

4.4.5 VÝPOČET PŘESUNU SYSTÉMU PODÉL CESTY GENERÁTOREM 2.ŘÁDU

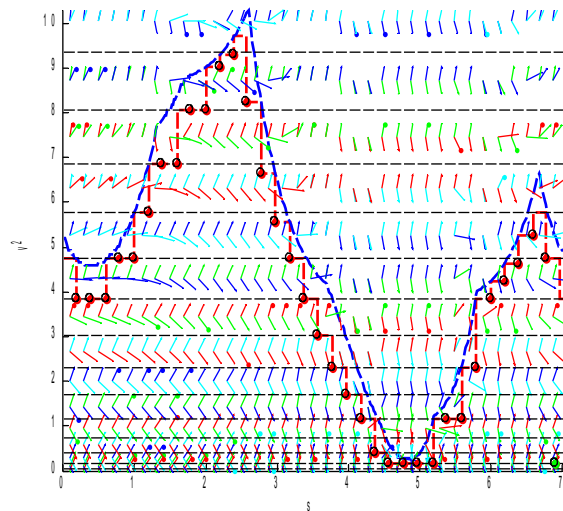
Z limitní křivky rychlosti vyplývá, že maximální možná konstantní rychlost je 0.4m/s. Můžeme tedy nastavit podmínky tak, aby manipulátor objel první kružnici co nejrychleji a pak pokračoval maximální možnou konstantní rychlostí. Dostaneme následující podmínky pro počáteční a koncový stav generátoru $\mathbf{x}_0 = [0 \ 0 \ 0]^T \rightarrow \mathbf{x}_f = [2\pi r \ 0.5 \ 0]^T$, $AM(s, \dot{s})$, $DM(s, \dot{s})$, $VM(s)$.

Diskretizací omezení dostaneme plánovací prostor generátoru, obrázek 4.22. Barevné vektory definují maximální a minimální strmost zrychlení pro každou diskretizovanou oblast. Pomocí výše uvedených metod plánování dostaneme výstupní data, která generují následující obrázky. Ty ukazují klíčové výstupy výpočtu obou přístupů. Pro srovnání je levý sloupec tvořen výstupy z algoritmu $GAVS(s)$, pravý výstupy z algoritmu $GAVS(s, v)$. Celkový čas přesunu systému je pro $GAVS(s, v)$ roven přibližně $t_f \approx 6,14 s$. U algoritmu $GAVS(s)$ kde byly limitní křivky zrychlení získány minimem hodnot zrychlení pod diskretizovanou křivkou rychlosti, dostaneme pomalejší přesun, přibližně $t_f \approx 6,7 s$.

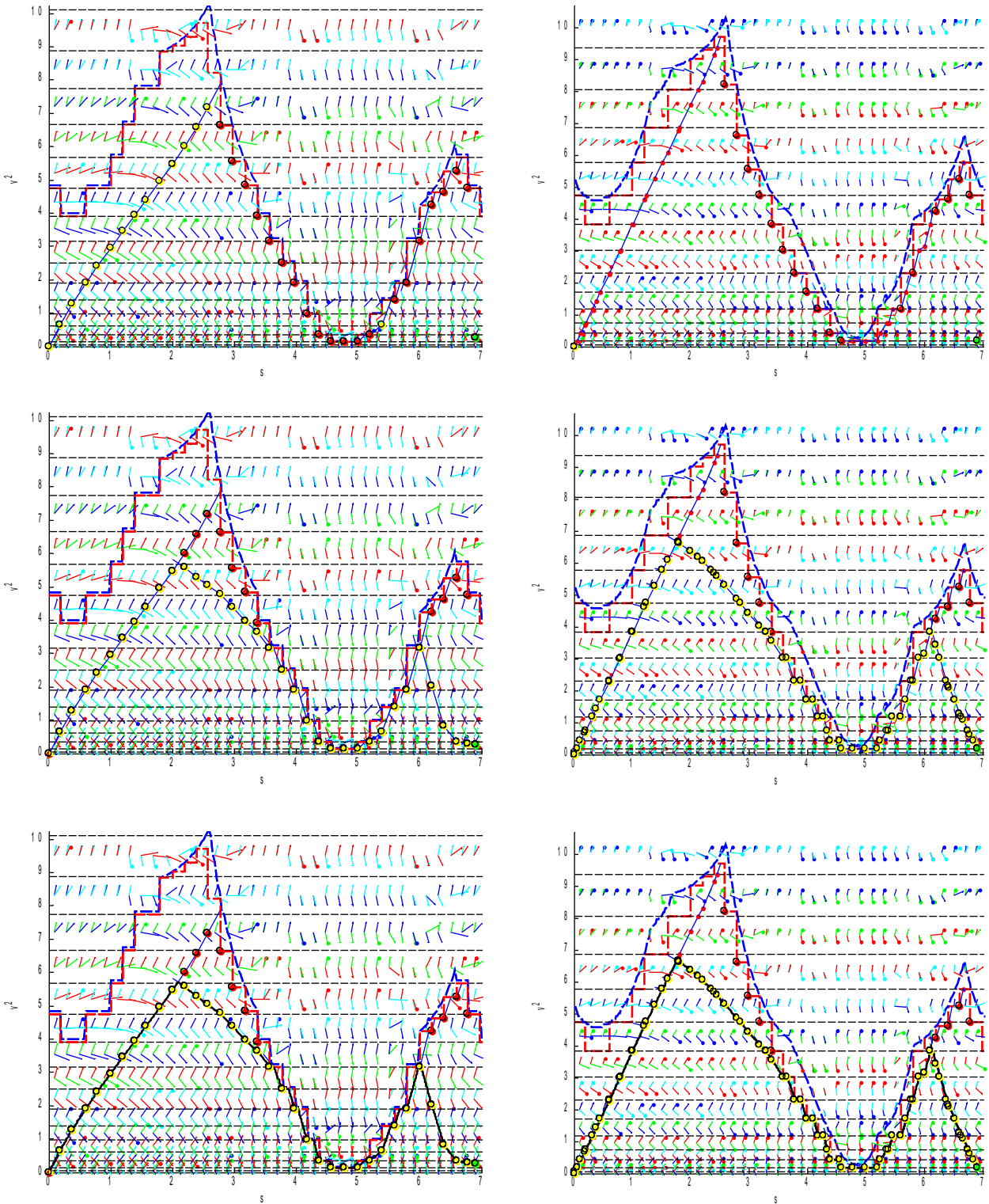
Nárůst času je velmi závislý na diskretizaci limitních křivek. Pokud bychom chtěli projet danou cestu maximální konstantní rychlostí, potom bychom dostali celkový čas přesunu $t_f \approx 18,3 s$. Z obrázku 4.21 je zřejmé, že tento nárůst času je silně závislý na tvaru limitní křivky rychlosti. Čím více se bude systém pohybovat blízko singularit, tím nižší bude minimum $VM(s)$.



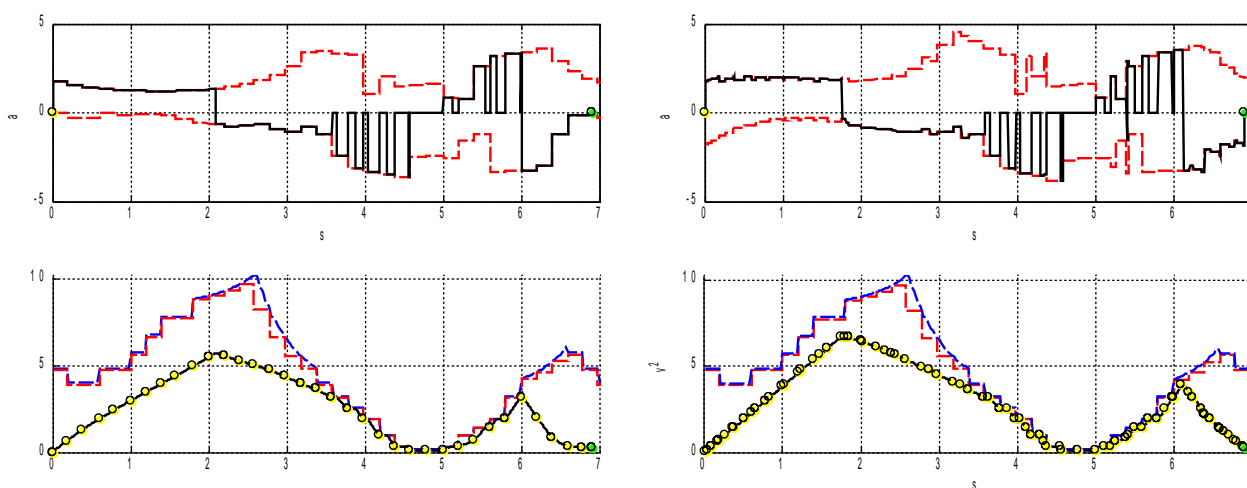
Obrázek 4.21: Minimem omezujících křivek rychlosti je limitní křivka rychlosti $VM^2(s)$.



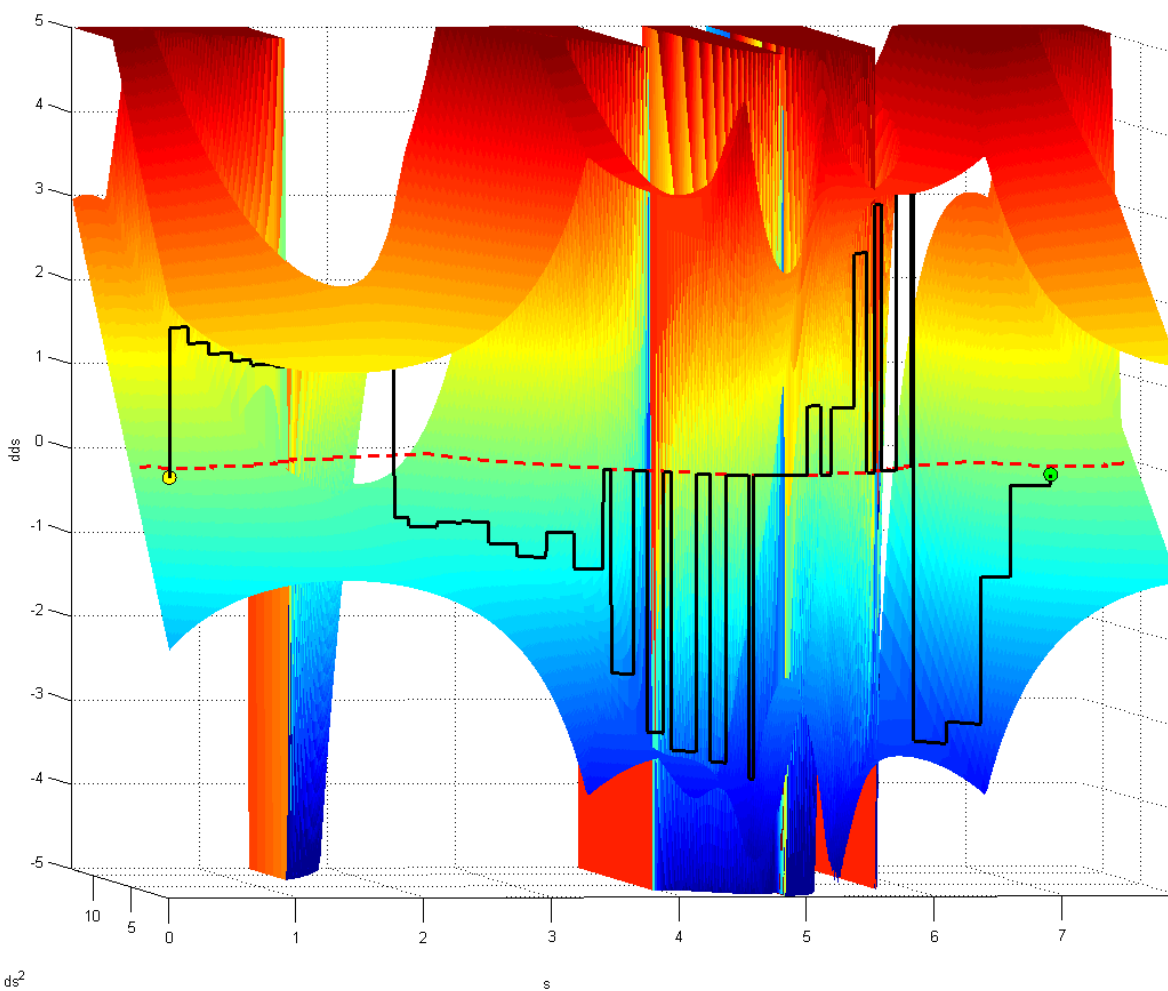
Obrázek 4.22: Diskretizace limitní křivky rychlosti a křivek zrychlení pro algoritmus $GAVS(s, v)$ nalezení bodů (s_i, \bar{v}_i) .



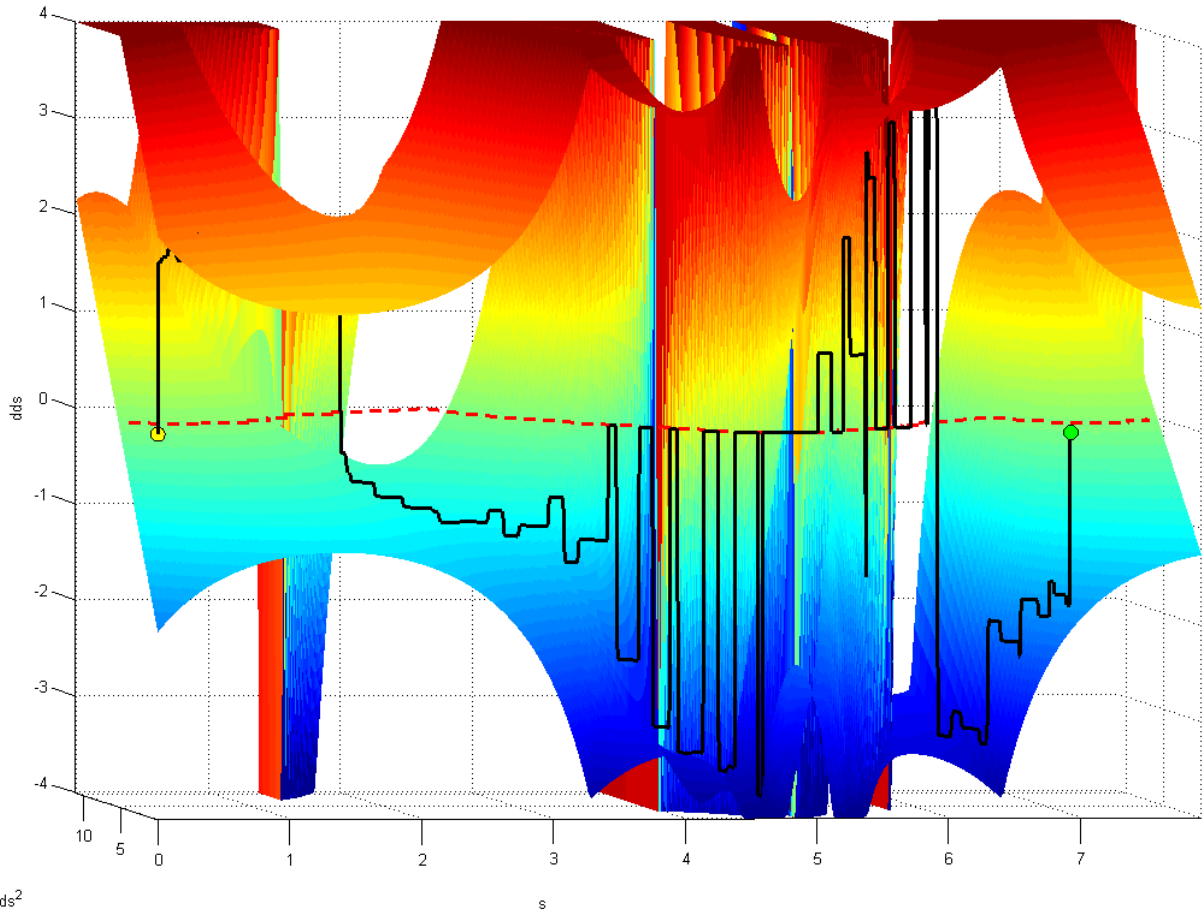
Obrázek 4.23: Grafické výstupy klíčových kroků 3, 4 a 5 výpočtu Algoritmu $GAVS(s)$ (vlevo) a $GAVS(s,v)$ (vpravo).



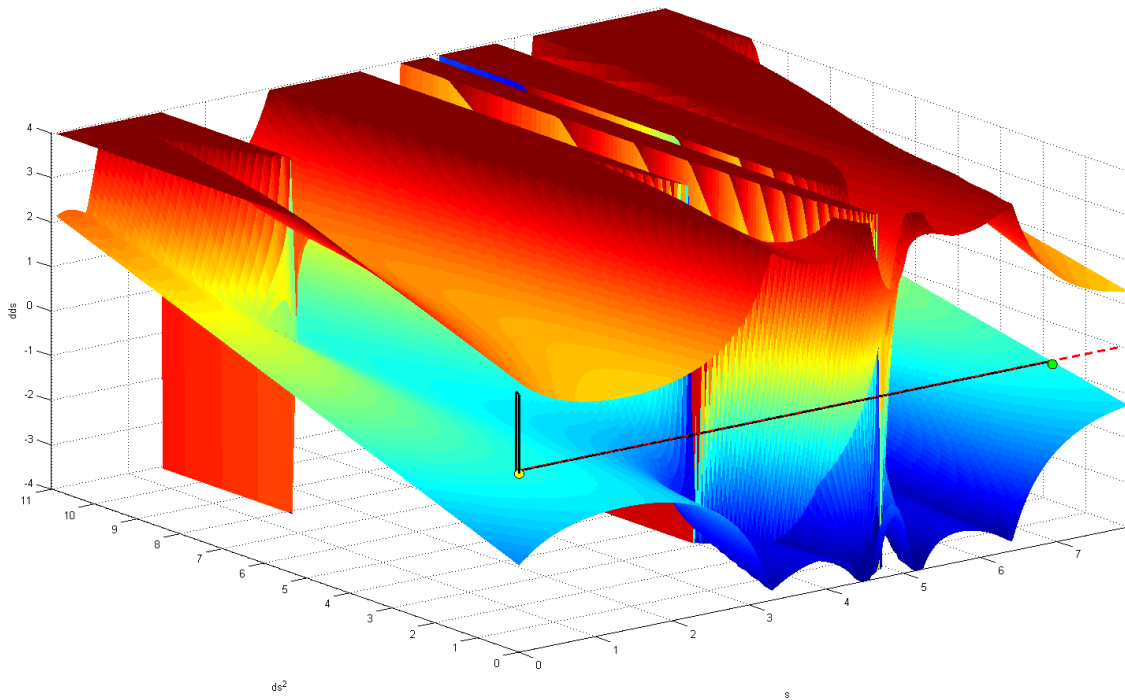
Obrázek 4.24: Výsledné trajektorie zrychlení a rychlosti algoritmu $GAVS(s)$ (vlevo) a $GAVS(s,v)$ (vpravo) v prostoru $(s-\bar{v})$.



Obrázek 4.25: $GAVS(s)$; pohyb stavu pod křivkami omezení podél specifikované cesty, $t_f \approx 6,7$ s.



Obrázek 4.26: $GAVS(s,v)$; pohyb stavu pod křivkami omezení podél specifikované cesty, $t_f \approx 6,14$ s.



Obrázek 4.27: Přesun maximální možné konstantní rychlosti podél specifikované cesty, $t_f \approx 18,3$ s.

ZÁVĚR

Řízení pohybu lze obecně rozdělit do tří úrovní, na plánování geometrické cesty stroje v konfiguračním nebo pracovním prostoru, generování trajektorií pohybu podél cesty a na zpětnovazební řízení/sledování generovaných trajektorií pohybu. Tato práce se věnuje druhé úrovni řízení pohybu. Zabývá se vývojem nových metod generování pohybu mechatronických systémů a technologických procesů. Hlavním kritériem při plánování pohybu je respektování pohybových omezení.

V úvodu je proveden základní rozbor úloh řízení pohybu. Nastíněny jsou standardní problémy generování trajektorií pohybu v jedné a více osách. Motivační příklad ukazuje vliv omezení na problém plánování pohybu. Dále je představena možnost využití teorie časově optimálního (t-opt.) řízení pro návrh generátorů trajektorií. Tato strategie umožňuje redukovat problém plánování pohybu systému s n stupni volnosti na problém řízení stavu generátoru pohybu s jedním stupněm volnosti. To umožňuje řešit i jinak zdánlivě neřešitelné problémy. Pro návrh optimálního řízení generátoru pohybu je využito hypotézy vyplývající z Pontryaginova principu maxima. T-optimální strategie řízení umožňuje dále transformovat problém plánování pohybu na problém hledání řešení soustav polynomiálních rovnic. Samotný výpočetní algoritmus pak využívá matematického nástroje z oblasti algebraické geometrie, který umožňuje obecně řešit polynomiální soustavy. Jedná se o techniku Gröbnerovy báze, jejíž kořeny, základní myšlenky a vlastnosti jsou v úvodu také představeny. Poslední část úvodu je věnována vytyčení cílů práce.

Teorie Gröbnerovy báze a její metody jsou hlavní náplní přípravné kapitoly. Kromě základních pojmů polynomiální algebry je zde zaveden pojem Gröbnerova báze jako speciální množina generátorů řešení původní soustavy polynomiálních rovnic. Pro nalezení této množiny je představen původní Buchbergerův algoritmus a jeho možné modifikace. Dále jsou ukázány některé možnosti využití Gröbnerovy báze v polynomiální algebře. Poslední část přípravné kapitoly je věnována aplikační působnosti této teorie v automatickém řízení. Z literatury vyplývá, že je možné tuto teorii využít například při vyšetřování říditelnosti a pozorovatelnosti systémů, při návrhu a parametrizaci výstupní zpětné vazby metodou přiřazení pólů, při popisu kinematiky robotů, pro řešení diofantické rovnice u 2-D systémů, pro obecnější popis vícerozměrných systémů a samozřejmě pro generování a plánování řídicích trajektorií.

Kapitola 3 se věnuje t-optimálnímu řízení systému řetězce integrátorů jako vhodného generátoru trajektorií pohybu. Úvodem je naznačeno několik standardních přístupů k řešení. Dále je představena základní struktura generátoru a uvedeno řešení úlohy generování přesunu z klidu do klidu. Generování obecného přesunu je možné pomocí GB převést na problém řešení soustav polynomiálních rovnic. Ukázalo se, že tímto způsobem je možné zcela obecně vyřešit t-optimální

problém řízení řetězce integrátorů s konstantními omezeními na všechny stavy. Tato strategie generování pohybu je hlavním výsledkem práce. S její pomocí je vytvořen obecný algoritmus *GBAVS/GAVS* pro generování trajektorií pohybu z libovolného počátečního stavu do libovolného koncového stavu. Struktura a základní kroky výpočtu jsou naznačeny. Předností algoritmu je, že nevyužívá žádné výpočetní smyčky ani cykly a obsahuje pouze explicitní vyčíslení vzorců.

Dále je ukázána možnost urychlení výpočtu pro plánování pohybu z konstantní rychlosti do konstantní rychlosti. Vlastnosti vyplývající z těchto předpokladů umožňují redukovat vyčíslení kořenů polynomiálních soustav na polovinu a dvojnásobně tak urychlit výpočet. Toto urychlení je dalším výsledkem práce. Závěr této kapitoly je věnován příkladům, které ukazují možnosti využití algoritmu *GBAVS* pro řízení pohybu v jedné ose a pro řízení nekoordinovaného pohybu víceosých strojů.

Kapitola 4 se zabývá posledním vytyčeným cílem a to problémem plánování podél specifikované cesty. Tento problém lze opět převést na problém řízení řetězce integrátorů, tentokrát ale s proměnnými omezeními. Z literatury víme, že tento přístup je využíván pro přímé numerické řešení pohybových rovnic a návrhu t-optimálního řešení. Bohužel tento přístup je velmi výpočetně náročný. Proto byl navržen postup, který využívá navržených generátorů *GBAVS/GAVS* a umožňuje pro diskretizovaná omezení generovat trajektorie pohybu s velmi nízkou výpočetní náročností. Opět není využito žádného cyklu ani výpočetní smyčky. Délka výpočtu je závislá pouze na množství diskretizovaných vzorků omezení. Tento přístup je dalším původním výsledkem této práce.

Problémem do budoucna je nalezení podobné strategie pro generátory vyšších řádů, které by zajistily spojitost generovaných trajektorií až do úrovně zrychlení, případně *Jerku*. Pro plánování pohybu s proměnnými omezeními je velmi obtížné nalézt jakékoli výpočetně nenáročné řešení, které bude plně respektovat pohybová omezení a zároveň bude generovat spojitě trajektorie pohybu. Podle dostupné literatury takové řešení není v současnosti známo, nebo plně nerespektuje pohybová omezení.

SHRnutí

Tato práce přináší základní poznatky z teorie řízení pohybu mechatronických systémů. Věnuje se druhé úrovni plánování pohybu, tedy generování řídicích trajektorií pohybu. Pro systémy s konstantními omezeními na pohybové veličiny pak navrhuje nové přístupy, založené na metodách algebraické geometrie, které umožňují analyticky řešit standardní problémy plánování pohybu v jedné ose. Také je ukázáno, jak tyto přístupy aplikovat pro napojování jednotlivých pohybů a jakým způsobem plánovat nekoordinovaný pohyb více os s časovou synchronizací. Pro systémy s proměnnými omezeními ukazuje možnost využití předešlých technik a pro nižší řád generátoru pohybu řeší obecný problém plánování podél specifikované cesty. Získané algoritmy jsou zcela obecné, nezávislé na tvaru omezení a složitosti systému, a jejich výpočetní náročnost závisí pouze na množství diskretizovaných vzorků omezení. Jejich hlavní předností je velmi nízká výpočetní náročnost a absence iteračních a dalších výpočetních cyklů.

RESUME

This thesis provides a basic knowledge of motion control theory of mechatronic systems. It deals with the second level of motion planning, i.e. generating the motion control trajectories. It proposes a new approaches for trajectory generation in single axis, based on the methods of algebraic geometry. This enables to solve motion planning problems for systems with constant constraints on motion variables and arbitrary initial and final state. It is also shown how to apply these techniques to linking individual movements to more complex motions and how to plan uncoordinated movement of multi-axis systems.

For systems with varying constraints it shows the possibility of using previous techniques and lower order generator to solve the general motion planning problem along the given path. Resulting algorithms are general, independent of the profile of constraints and the computational complexity depends only on the number of samples of limit curves. The main advantage is very low computational cost, analytical background and lack of iterative computational cycles.

*Logic will get you from A to B.
Imagination will take you everywhere.*

Albert Einstein

A NOVÉ SMĚRY V TEORII GRÖBNEROVY BÁZE

Největší pokrok, od doby kdy byl Buchbergerův algoritmus prvně publikován, nastal v oblasti efektivity výpočtu dané výběrem způsobu uspořádání členů, a snížením počtu S-polynomů které musí být uvažovány. V literatuře nalezneme následující ideje, které přispěly k dramatickému zrychlení původního algoritmu.

- Využití kritéria pro výběr pouze určitých S-polynomů, [23];
- Gröbner walk, [38];
- Přístup na základě lineární algebry, [34];

Všechny tyto přístupy nemění základní ideu algoritmické konstrukce a nepostradatelnou roli S-polynomů. Podívejme se stručně na nejzajímavější výsledky poslední doby.

ZRYCHLENÍ BUCHBERGEROVO ALGORITMU

Dramatické zrychlení algoritmu může být dosaženo vhodným výběrem S-polynomů, které je možné z redukce modulo $\overline{S(f_i, f_j)}^{G_i}$ vynechat. Tento přístup byl nazván řetězovým kritériem a popsán a dokázán v [20]. Kritérium říká, že redukce S-polynomu $S(f_i, f_j)$ může být vynechána, pokud existuje f_k takové, že hlavní monom dělí nejmenší společný násobek hlavních monomů f_i, f_j a současně S-polynom $S(f_k, f_i)$ a zároveň $S(f_k, f_j)$ byl již v algoritmu uvažován.

ALGORITMUS F4, F5

Buchbergerův algoritmus byl prvním algoritmem pro výpočet Gröbnerovy báze. Bohužel v některých případech, i pro soustavy nízkého řádu, je výpočet velmi časově náročný a někdy se řešení v rozumném čase nedočkáme. Jean-Charles Faugère našel postup, jakým lze urychlit výpočet redukce \overline{f}^G . Například tzv. problematický cyklus 9 byl prolomem právě prvním z Faugèrových algoritmů, F4 [34].

Jádrem tohoto algoritmu je myšlenka převést redukci polynomu na převod jakési vhodně uspořádané matice do redukované formy a následně odtud vyčíst redukovanou formu polynomu. To v zájmu co nejrychlejšího výpočtu není zajištěno Gaussovou eliminací, ale my se v následujícím příkladu podívejme, jak celý proces s využitím Gaussovy eliminace probíhá.

Uvažujme polynom $f_i = 2x^2 + xy - 2y$, který chceme redukovat množinou $\{g_1, g_2\} = \{x-1, y+2\}$. Pomocí dělicího algoritmu (kap. 2.2) s resp. uspořádání $x_{lex} > y$

$$\begin{array}{r}
2x^2 + xy - 2y : \{x-1, y+2\} \\
\underline{-2x(x-1)} \\
xy + 2x - 2y \\
\underline{-y(x-1)} \\
2x - y \\
\underline{-2(x-1)} \\
-y + 2 \\
\underline{1(y+2)} \\
4
\end{array}$$

dostaneme, že $f_i = (2x + y + 2)g_1 - g_2 + 4$ byl redukován množinou na hodnotu 4.

Nyní provedeme stejný výpočet Faugèrovým postupem pomocí Gaussovy eliminace. Uspořádáme polynomy f, g_1, g_2 do matice

$$\begin{array}{c}
x^2 \quad xy \quad x \quad y \quad 1 \\
\left[\begin{array}{ccccc}
1 & 0 & -1 & 0 & 0 \\
0 & 1 & 0 & -1 & 0 \\
0 & 0 & 1 & 0 & -1 \\
0 & 0 & 0 & 1 & 2 \\
2 & 1 & 0 & -2 & 0
\end{array} \right] = A_i
\end{array} \quad (1)$$

Gaussovou eliminací dostaneme

$$\begin{array}{c}
x^2 \quad xy \quad x \quad y \quad 1 \\
\left[\begin{array}{ccccc}
1 & 0 & -1 & 0 & 0 \\
0 & 1 & 0 & -1 & 0 \\
0 & 0 & 1 & 0 & -1 \\
0 & 0 & 0 & 1 & 2 \\
0 & 0 & 0 & 0 & 4
\end{array} \right] = \tilde{A}_i
\end{array}$$

kde v posledním řádku je vyjádření redukováného polynomu f_i . Klíčovou otázkou je, jak vytvořit matici A_i .

Víme, že v každém kroku dělicího algoritmu se snažíme eliminovat nejvyšší monom v dílčím výsledku h . Chceme, aby tento výpočet byl obsažen v matici. Tedy jestliže h je dílčí výsledek redukce a jeho hlavní monom může být dále redukován $h - tg_i$, potom matice A_i musí obsahovat řádek popisující tg_i . Počet sloupců je dán počtem monomů, které lze z dané množiny f_1, g_1, g_2 vygenerovat.

Dostáváme následující algoritmus, jehož průchodem dostaneme matici (1).

- 1) Necht' M je množina monomů z f .
- 2) Jestliže M je prázdná, přidej do A ještě f a ukonči algoritmus.
- 3) Vyjmi m -hlavní monom z M .
- 4) Pokud m nelze redukovat pomocí g_i jdi na krok 2.
- 5) Vyber monom t a g_i tak, že $m = LT(tg_i)$
- 6) Přidej tg_i do A , přidej monomy tg_i kromě m do M a jdi na krok 2.

Uvažujme, že chceme redukovat polynomy f_1, \dots, f_n . V příkladu jsme tvořili matici pro redukci jednoho polynomu f_i . Museli bychom tedy vytvořit matice pro všechny polynomy. Tímto způsobem postupuje i Buchbergerův algoritmus. Nicméně F4 využívá toho, že je možné tvořit

matici A , která obsahuje všechny řádky matic A_i (vyjma duplikátů), abychom následně redukovali několik polynomů najednou. Samozřejmostí F4 algoritmu je, že využívá nástroje na práci s řídkými maticemi, které jsou polynomy generovány. Detailní popis algoritmu lze nalézt např. v [34].

Faugère F5 algoritmus je vylepšením F4 algoritmu. Algoritmus nejprve spočte GB první dvojice generátorů. Poté přidává další generátory a tuto bázi využije k redukci velikosti matice A . Tato strategie umožňuje použít dvě nová kritéria, Faugèrem nazývána jako *signatures of polynomials*. Podrobný popis tohoto přístupu lze nalézt v [35].

GRÖBNER WALK ALGORITMUS

Jak jsme již uvedli, složitost a rychlost výpočtu Gröbnerovy báze je silně podmíněna výběrem uspořádání členů. Proto je v některých případech vhodné nalézt GB s takovým typem uspořádání (\langle_{grevlex}), jehož výpočetní náročnost je nízká, a potom nějakým způsobem přejít k požadovanému uspořádání, např. \langle_{lex} . Jedním z prvních algoritmů, pracujícím na tomto principu, je tzv. *FGLM algoritmus*, vymyšlený již zmíněným Faugèrem a jeho spolupracovníky Gianni, Lazard, Mora v roce 1993, viz [36]. *FGLM algoritmus* umí pracovat pouze s ideálem nulové dimenze. Tento deficit byl v roce 1997 odstraněn vývojem nového tzv. *Gröbner Walk algoritmu* [28]. Ten nejprve spočte GB s respektováním \langle_{grevlex} uspořádání, protože se ukázalo, že tento výpočet je nejméně časově náročný. Potom přejde pomocí inkrementálních kroků daných topologií požadovaného uspořádání ke Gröbnerově bázi s tímto uspořádáním. V některých případech musí využít perturbací a aritmetiku velkých čísel. Tento problém byl vyřešen v roce 2007 a došlo k vylepšení původního *Gröbner Walk algoritmu* do nové podoby, tzv. *Generic Gröbner Walk algoritmus* [38]. Podívejme se na základní princip této metody, která je zajímavá nejen z hlediska výpočtu GB , ale i z geometrického pohledu na strukturu GB .

Základní myšlenka vyplývá z významného faktu, že ačkoli existuje nekonečně mnoho možností uspořádání členů, počet výsledných redukovaných Gröbnerových bází příslušných ideálu I je konečný [97]. Druhým významným faktem je, že dvě „sousední“ báze se setkávají na společné stěně takzvaného Gröbnerova kuželu. Dostáváme tedy konečný počet kuželů příslušejících konečnému počtu redukovaných Gröbnerových bází. Tyto kužely tvoří tzv. *restricted Gröbner fan* v \mathbb{R}_+^n . Třetím významným faktem, který přispívá k rychlosti výpočtu je, že přechod do sousedního kuželu, neboli získání GB v sousedním kuželu, lze vypočítat pomocí tzv. překroku, s použitím pouze dělicího algoritmu a tzv. *toric degeneration* $\langle I_\omega \rangle$ původního ideálu I . Abychom mohli dané kužely nalézt a přecházet mezi nimi, je nutné nejprve zavést speciální typ uspořádání.

Vážené uspořádání (wdeg)

Definice 1. Necht' $\omega = (\omega_1, \dots, \omega_n) \in \mathbb{R}^n$, $\mathbf{a} = (a_1, \dots, a_n)$ a $\mathbf{b} = (b_1, \dots, b_n) \in \mathbb{Z}_{\geq 0}^n$. Řekneme, že $\mathbf{a}_\omega > \mathbf{b}$ pokud $\omega \cdot \mathbf{a} > \omega \cdot \mathbf{b}$, nebo $\omega \cdot \mathbf{a} = \omega \cdot \mathbf{b}$ a $\mathbf{a}_\sigma > \mathbf{b}$, kde σ reprezentuje nějaké lexikografické uspořádání ($\langle_{\text{lex}}, \langle_{\text{grevlex}}, \dots$). Potom zapíšeme vztah mezi monomy takto $\mathbf{x}^{\mathbf{a}}_{\sigma_\omega} > \mathbf{x}^{\mathbf{b}}$.

Věta 1. Každé lexikografické uspořádání $\langle_{\text{term order}}$ lze nahradit příslušným váženým uspořádáním \langle_ω pro daný ideál I , neboli $\text{in}_\omega(I) \subseteq \text{in}_\omega(I)$, kde $\text{in}_\omega(I)$ označuje **monomický ideál** vybraný z původního ideálu I vzhledem k uspořádání \langle_ω a $\text{in}_\omega(I)$ označuje ideál vybraný vzhledem k uspořádání \langle_ω .

Poznamenejme, že $\text{in}_\omega(I)$ nemusí být obecně pouze *monomický ideál*, nicméně pro libovolné uspořádání \langle a ideál I existuje $\omega \in \mathbb{R}^n$ takové, že $\text{in}_\omega(I) = \text{in}_\omega(I)$.

Definujme ještě zmíněný Gröbnerův kužel, objekt důležitý pro popis *Gröbner Walk Algoritmu*.

Definice 2. Necht' $G = \{g_1, \dots, g_r\}$ je redukovaná Gröbnerova báze ideálu $I \subseteq k[\mathbf{x}]$ při daném způsobu uspořádání $<$. Gröbnerův kužel $C_<(I) \subseteq \mathbb{R}_+^n$ je tvořen všemi $\omega \in \mathbb{R}_+^n$ takovými, že

$$\omega \cdot \mathbf{u} \geq \omega \cdot \mathbf{v}, \text{ neboli } \omega \cdot (\mathbf{u} - \mathbf{v}) \geq 0, \quad (2)$$

kde $\mathbf{x}^u = LM(g_i)$, $i = 1, \dots, r$, a $\mathbf{x}^v \neq \mathbf{x}^u$ jsou všechny ostatní monomy vyskytující se v g_i .

Lemma. ω je uvnitř Gröbnerova kuželu $C_<(I) \Leftrightarrow in_\omega(g_i) = in_<(g_i), \forall g_i \in G$.

Definice 3 (překrok). Necht' $I \subseteq k$ je ideál a $<_1, <_2$ jsou lexikografická uspořádání na $k[\mathbf{x}]$. Předpokládejme, že G je redukovaná Gröbnerova báze s respektováním $<_1$. Jestliže $\omega \in C_<_1(I) \cap C_<_2(I)$, potom

I) redukovaná Gröbnerova báze pro $in_\omega(I)$ s respektováním $<_1$ je $G_\omega = in_\omega(g_i), \forall g_i \in G$.

II) Jestliže H je redukovaná Gröbnerova báze pro $in_\omega(I)$ s respektováním $<_2$, potom $\{f - \bar{f}^G \mid f \in H\}$ je minimální Gröbnerova báze pro I vzhledem k $<_{2\omega}$, kde \bar{f}^G je jednoznačný zbytek po dělení.

III) Redukovaná Gröbnerova báze pro I vzhledem k $<_{2\omega}$ je totožná s redukovanou Gröbnerovou bází pro I s respektováním $<_2$.

Definice vychází z faktu, že

$$in_\omega(f) = in_\omega(f - \bar{f}^G),$$

neboť všechny monomy \bar{f}^G budou mít striktně nižší váhu než f .

Potom můžeme spočíst vzorec překroku

$$\begin{aligned} in_{<_{2\omega}}(I) &= in_{<_2}(in_\omega(I)) \\ &= \langle in_{<_2}(f_1), \dots, in_{<_2}(f_s) \rangle \\ &= \langle in_{<_2}(in_\omega(f_1)), \dots, in_{<_2}(in_\omega(f_s)) \rangle \\ &= \langle in_{<_2}(in_\omega(f_1 - \bar{f}_1^G)), \dots, in_{<_2}(in_\omega(f_s - \bar{f}_s^G)) \rangle \\ &= \langle in_{<_{2\omega}}(f_1 - \bar{f}_1^G), \dots, in_{<_{2\omega}}(f_s - \bar{f}_s^G) \rangle \end{aligned}$$

kde $\{f_1 - \bar{f}_1^G, \dots, f_s - \bar{f}_s^G\} \subseteq I$ je minimální GB ideálu I vzhledem k uspořádání $<_{2\omega}$.

Příklad (Gröbner Walk)

Nyní již můžeme přistoupit k samotnému postupu výpočtu, který algoritmus Gröbner walk sleduje, i když jsou nutná některá další opatření. Na jednoduchém příkladu nastíníme konkrétní postup výpočtu, který obrázek 1 ilustruje. Příklad je převzat z článku [38].

Uvažujme ideál $I = \langle x^2 - y^3, x^3 - y^2 - x \rangle \subset \mathbb{Q}[x, y]$. Chceme získat jeho GB vzhledem k uspořádání $y <_{lex} x$. Prvním krokem je spočtení GB vzhledem k uspořádání $y <_{grevlex} x$

$$G_{grevlex}^1 = \{\underline{y}^3 - x^2, \underline{x}^3 - y^2 - x\}, \quad (3)$$

kde vedoucí monomy jsou podtrženy.

Gröbnerův kužel je dán průnikem oblastí ohraničených hraničními přímkami, které získáme z jednotlivých monomů GB. Pro lepší srozumitelnost přepíšeme bázi $G_{grevlex}^1$ do tvaru

$$G_{<_{\text{revlex}}}^1 = \{x^0 y^3 - x^2 y^0, x^3 y^0 - x^0 y^2 - x^1 y^0\}.$$

Určíme hraniční přímky Gröbnerova kuželu

$$\begin{aligned} g_1: \quad & 0 \omega_1 + 3 \omega_2 \geq 2 \omega_1 + 0 \omega_2 \\ & -2 \omega_1 + 3 \omega_2 \geq 0 \quad \Rightarrow \quad \omega \cdot (-2, 3) \geq 0 \\ g_2: \quad & 3 \omega_1 - 2 \omega_2 \geq 0 \quad \Rightarrow \quad \omega \cdot (3, -2) \geq 0 \\ g_3: \quad & 2 \omega_1 - 0 \omega_2 \geq 0 \quad \Rightarrow \quad \omega \cdot (2, 0) \geq 0. \end{aligned}$$

Dostáváme

$$C_{<_{\omega}}(I) = C(\{(-2, 3)\} \cap \{(3, -2)\}) \subseteq \mathbb{R}_+^n.$$

Uvnitř kuželu platí, že všechny přípustné váhy ω vedou na $G_{<_{\text{revlex}}}^1$. Na hranicích kuželu dostáváme ideály příslušející hraničním přímkám ve tvaru

$$\begin{aligned} \langle in_{\omega(-2,3)=0}(G^1) \rangle &= \langle y^3, x^3 - y^2 \rangle \\ \langle in_{\omega(3,-2)=0}(G^1) \rangle &= \langle y^3 - x^2, x^3 \rangle, \end{aligned} \quad (4)$$

kde vedoucí monomy jsou stále stejné jako v (3). Nacházíme se na hranici kuželu a z (4) vidíme, které monomy se při přechodu hranice stanou vedoucími v sousedních kuželech. Směr, kterým budeme postupovat, je dán právě možnostmi změny vedoucích monomů a naším konečným požadavkem na uspořádání ve tvaru $y <_{\text{lex}} x$.

Redukovaná Gröbnerova báze respektující žádané uspořádání $<_{\text{lex}}$ pro druhý torický ideál $\langle in_{\omega=(3,-2)}(G^1) \rangle$ je $H_{<_{\text{lex}}} = \{x^2 - y^3, x y^3, y^6\}$ a pomocí překroku $L_G(f) = f - \bar{f}^G$ dostáváme Gröbnerovu bázi v sousedním kuželu

$$\{L_G(x^2 - y^3), L_G(x y^3), L_G(y^6)\} = \{x^2 - y^3, x y^3 - y^2 - x, y^6 - x y^2 - x^2\},$$

kterou převedeme do redukovaného tvaru (pro korektnost algoritmu není nutností)

$$G^2 = \{x^2 - y^3, x y^3 - y^2 - x, y^6 - x y^2 - y^3\}.$$

Dostali jsme redukovanou GB v sousedním kuželu vzhledem k uspořádání ω , které již není jedním z typických lexikografických uspořádání. Znovu nalezneme hraniční přímky

$$\begin{aligned} g_1: \quad & \omega \cdot (2, -3) \geq 0 \\ g_2: \quad & \omega \cdot (1, 1) \geq 0 \\ & \omega \cdot (0, 3) \geq 0 \\ g_3: \quad & \omega \cdot (-1, 4) \geq 0 \\ & \omega \cdot (0, 3) \geq 0 \end{aligned}$$

a určíme Gröbnerův kužel pro G^2

$$C_{<_{\omega}}(I) = C(\{(2, -3)\} \cap \{(-1, 4)\}) \subseteq \mathbb{R}_+^n.$$

V tomto jednoduchém příkladu, kdy máme pouze dvě výsledné hraniční přímky, je zřejmé kterým směrem budeme postupovat. Opakujeme předchozí postup a dostáváme $\langle in_{\omega(-1,4)=0}(G^2) \rangle = \langle x^2, x y^3, y^6 - x y^2 \rangle$ a z něj $H_{<_{\text{lex}}}^2 = \{x^2, x y^2 - y^6, y^7\}$. Pomocí překroku dostáváme bázi

$$\{L_G(x^2), L_G(x y^2 - y^6), L_G(y^7)\} = \{x^2 - y^3, x y^2 - y^6 + y^3, y^7 - y^4 - y^2 - x\},$$

kterou převedeme do redukovaného tvaru

$$G^3 = \{x^2 - y^3, x y^2 - y^6 + y^3, y^7 - y^4 - y^2 - x\}. \quad (5)$$

Z vedoucích monomů vidíme, že i báze (5) ještě není hledanou GB s požadovaným uspořádáním.

Proto znovu nalezneme hraniční přímky

$$\begin{aligned} g_1: & \quad \omega \cdot (2, -3) \geq 0 \\ g_2: & \quad \omega \cdot (1, -4) \geq 0 \\ & \quad \omega \cdot (1, -1) \geq 0 \\ g_3: & \quad \omega \cdot (0, 3) \geq 0 \\ & \quad \omega \cdot (0, 5) \geq 0 \\ & \quad \omega \cdot (-1, 7) \geq 0 \end{aligned}$$

Gröbnerův kužel pro G^3 je dán

$$C_{<\omega}(I) = C(\{(1, -4)\} \cap \{(-1, 7)\}) \subseteq \mathbb{R}^n,$$

a pomocí $\langle in_{\omega(-1,7)=0}(G^3) \rangle = \langle x^2, x y^2, y^7 - x \rangle$, $H_{<\text{lex}}^3 = \{y^9, x - y^7\}$ a překroku

$$\{L_G(y^9), L_G(x - y^7)\} = \{y^9 - 2y^6 - y^4 + y^3, x - y^7 + y^4 + y^2\}$$

dostáváme další bázi

$$G^4 = \{y^9 - 2y^6 - y^4 + y^3, x - y^7 + y^4 + y^2\}. \quad (6)$$

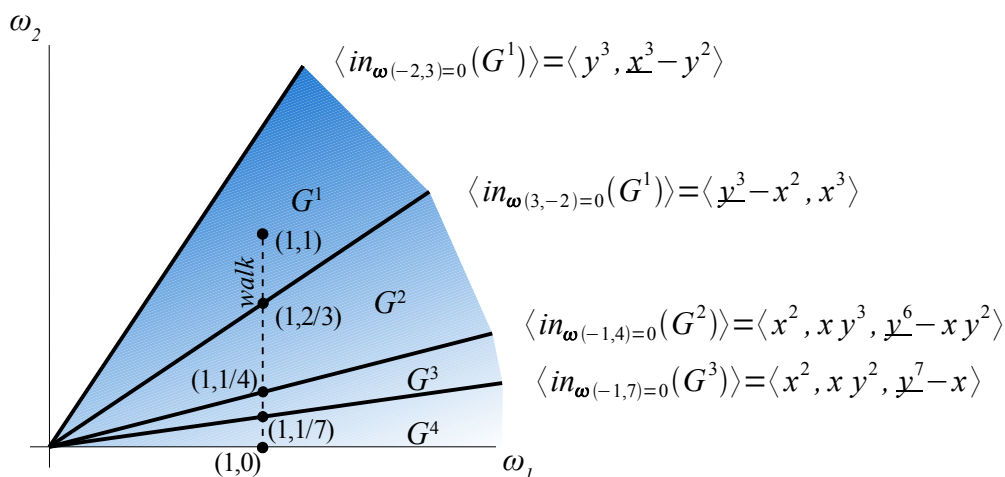
Pomocí předcházejícího Lemmatu zjistíme, že $in_{\omega}(G^4) = in_{<\text{lex}}(G^4)$. Algoritmus se zastaví a G^4 je hledanou bází $G_{<\text{lex}}^4$ ideálu I .

Hledáním hraničních přímek pro G^4

$$\begin{aligned} g_1: & \quad \omega \cdot (0, 3) \geq 0 \\ & \quad \omega \cdot (0, 5) \geq 0 \\ & \quad \omega \cdot (0, 6) \geq 0 \\ g_2: & \quad \omega \cdot (1, -7) \geq 0 \\ & \quad \omega \cdot (1, -4) \geq 0 \\ & \quad \omega \cdot (1, -2) \geq 0 \end{aligned}$$

dospějeme k faktu, že „spodní stěna“ splývá s hranicí *restricted Gröbner fan*, a tedy není kam pokračovat.

Gröbner walk se používá všude tam, kde již tradiční metody hledání *GB* vzhledem k lexikografickému uspořádání $<_{\text{lex}}$ selhávají. Další nástroje, jako je tento algoritmus, jsou neustále rozvíjeny. Jsou převáděny do dalších oblastí, např. *konvexních polytopů*, kde jsou hledány další zajímavé vlastnosti a podobnosti. Zavádí se i nové pojmy jako například *phylogenetické stromy* a další objekty *torické algebry*. Poznamenejme, že *FGLM algoritmus* byl s využitím *Hilbertových funkcí* taktéž zobecněn pro obecné použití. Pro hlubší prostudování těchto metod odkazují čtenáře na [97]. Knihovny pro nalezení a práci s Gröbnerovou bází lze nalézt v komerčních softwarech jako např.: Maple, Macsyma, Mathematica, Macaulay 2, CoCoA, GAP, Magma, Singular, atd.



Obrázek 1: Gröbner fan.

VOLBA TYPU USPOŘÁDÁNÍ

Otázkou zůstává, jak vybrat nejvhodnější uspořádání členů (monomů) pro danou soustavu rovnic vzhledem k nejsnazšímu nalezení Gröbnerovy báze. Tímto a dalšími problémy spojenými s typy uspořádání se podrobně zabývá Bernd Sturmfels ve své publikaci [97]. Podívejme se pouze na jeden z jeho motivačních příkladů a nastiňme možnosti váženého uspořádání v této oblasti. Heuristický algoritmus pro nalezení optimálního uspořádání je popsán v [14] a je implementován v softwaru REDUCE Gröbner basis package.

Příklad. Předpokládejme, že hledáme počet komplexních řešení polynomiální soustavy

$$F = \{x^5 + y^3 + z^2 - 1, x^2 + y^2 + z - 1, x^6 + y^5 + z^3 - 1\} \subset \mathbb{Q}[x, y, z]$$

Můžeme zkusit využít čistě lexikografické uspořádání. Pro tento případ existuje 6 GB pro 6 různých lexikografických uspořádání. Bohužel každá z těchto bází obsahuje polynomy vysokého stupně. Například pro uspořádání $x < y < z$ dostáváme nejvyšší polynom 21. stupně s největším koeficientem 1553067597584776499. Výpočet je samozřejmě pomalý. Celkový počet řešení bychom museli určit pomocí ostatních bázových polynomů.

Pokud ovšem zvolíme vážené uspořádání $\omega = (3, 4, 7)$, hlavní členy soustavy F budou

$$in_{\omega}(F) = \{x^5, y^2, z^3\}.$$

Tyto monomy jsou nesoudělné a F je tedy pro tento typ uspořádání přímo Gröbnerovou bází. Vidíme, že ideál $\langle F \rangle$ je ideálem nulové dimenze a soustava má tedy podle věty 2.7 nanejvýš 30 řešení v \mathbb{C}^3 .

B ALGORITMY *GAVS* A *GBAVS*

VÝPOČETNÍ PROCEDURY ALGORITMU *GAVS*

```
function y = U1(AM,DM,VM,s0,v0,a0,sf,vf,af,t11,t32)
sol = zeros(1,5);
solution = zeros(1,5);

%set1
t31 = -(vf-VM+AM*t32)/DM;
t12 = -(-VM+v0+DM*t11)/AM;
t2 = 1/2*(2*v0*DM^2*t11+2*AM^2*vf*t32-AM^2*t32^2*DM-AM*VM^2+2*AM*DM*sf-2*AM*DM*s0-
AM*DM^2*t11^2+AM*vf^2+DM*v0^2+t32^2*AM^3+DM^3*t11^2-DM*VM^2-2*AM*v0*DM*t11-2*AM*DM*vf*t32)/AM/DM/VM;
sol(1,:) = [t11,t12,t2,t31,t32];

%set2
t2 = 0;

p0 = 2*AM^2*t32^2+2*AM*s0-2*AM*sf+AM*DM*t11^2+2*AM*v0*t11-DM^2*t11^2-v0^2+4*AM*t32*vf-
2*v0*DM*t11+vf^2;
p1 = 2*AM*DM*t32+2*AM*vf+2*DM*vf+2*AM^2*t32;
p2 = DM^2+AM*DM;
p = [p2 p1 p0];
p = facelift(p); %ošetřuje podmíněnost koeficientů polynomu

%t31
t = roots(p);
for i = 1:length(t)
t12 = -(v0-vf-AM*t32+DM*t11-DM*t(i))/AM;
sol(i+1,:)=[t11,t12,t2,t(i),t32];
end;
%end;
for i = 1:length(sol)
if sol(i,*)>=0 & isreal(sol(i,*))
y.solution = sol(i,);
return
end;
end;
y.sol = sol;

function y = s_minmax_values_avs(sol,u,s0,v0,sf,vf)
%urceni minimalnich a maximalnich hodnot t-opt reseni;
s_1 = s0 + v0*sol(1) + u(1)*(sol(1)^2)/2;
s_4 = sf - vf*sol(5) + u(5)*(sol(5)^2)/2;
ss = sort([s_1,s_4])
y.smax = ss(2);
y.smin = ss(1);
%!!!!!!!!!!!!!!!!!!!! end of v_values
```



```

%6)kontrola zda reseni opravdu vede do cile (do xf)
tol = 10^-7; %tolerance pro numer chyby
kontrola = s_minmax.coeff(8,:);
if (abs(kontrola(2) - a_fin) < tol) && (abs(kontrola(3) - v_fin) < tol) && (abs(kontrola(4) - s_fin)
< tol)

else
    'error 7 - chyba malych cisel'
    return
end;
%6)!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

%7)!!!!!!znovu kontrola omezeni VM,AM,DM pro vysledne reseni
%melo by to vzdy platit, ale...
if (a_minmax.amax <= c_acc+tol) && (a_minmax.amin >= -c_dec-tol) && (v_minmax.vmax <= c_vel+tol) &&
(v_minmax.vmin >= -c_vel-tol)

else %kontrola omezeni AM,VM,DM neprosla
    'error 6 - unexpected' %CHYBA ALGORITMU
    return
end;

%8)!!!!!!!!!!!!!!!!!!!! test c_pos
%kontrola zda se reseni "vejde" do pracovniho prostoru S
if (s_minmax.smin >= c_pos_m) && (s_minmax.smax <= c_pos_p)
    solution %vypis reseni
    for i = 1:7 %vypocet vektoru tau (tau1...tau7)
        tau(i) = sum(solution(1:i));
    end;
    tau %vypis tau
    u %vypis prislusneho rizeni
    finaltime = tau(7)

    a_minmax %vypis vysledneho casu
    v_minmax %vypis min-max hodnot
    s_minmax

else %nalezene reseni vede do cile ale nevejde se do S
    'error 5 - poruseni c_pos'
    s_minmax
    return
end;
%!!!!!!!!!!!!!!!!!!!!!!!!!!!! end test c_pos

--- RESENI NALEZO A ZKONTROLOVANO ---
%!!!!!!!!!!!!!!!!!!!!!!!!!!!!hledani koeficientu polynomu

pom = [0,tau];
for i = 1:7 %prepocetni koeficientu s_minmax.coeff do pocatku casu viz zprava
    a_z = polyval([u(i) s_minmax.coeff(i,2)],-pom(i));
    v_z = polyval([u(i)/2 s_minmax.coeff(i,2) s_minmax.coeff(i,3)],-pom(i));
    s_z = polyval([u(i)/6 s_minmax.coeff(i,2)/2 s_minmax.coeff(i,3) s_minmax.coeff(i,4)],-pom(i));
    coeff(i,:) = [u(i) a_z v_z s_z];
end;

function y = a_minmax_values(sol,u1,a0,af)
%urceni minimalnich a maximalnich hodnot t-opt reseni;
a1_p = a0;
a2_p = af;
a3_p = a0 + u1*sol(1);
a4_p = af - u1*sol(7);
y.amin = min([a1_p,a2_p,a3_p,a4_p]);
y.amax = max([a1_p,a2_p,a3_p,a4_p]);
%!!!!!!!!!!!!!!!!!!!!!!!!!!!! end of a-values

```

```

function y = v_minmax_values(sol,u1,v0,a0,vf,af)
%urceni minimalnich a maximalnich hodnot t-opt reseni;
a_2 = a0 + u1*sol(1);   %=a2_p      urceni a(t) v case t2
a_3 = a_2 - u1*sol(3);  %=a3_p      urceni a(t) v case t3
v_2 = v0 + a0*(sol(1)+sol(2)) + u1*(sol(1)^2/2 + sol(1)*sol(2));%urceni v(t) v case t2
%!!!!!!!!!!!!!!!!!!!!!! v-values
%!!!!!!!!!!!!!!v2_p
if sign(a0)~=sign(a_2)
    v2_p = v0 - a0^2/(2*u1);
else
    v2_p = v0;
end;
%!!!!!!!!!!!!!!!!!!!!!!

%!!!!!!!!!!!!!! v3_p
if sign(a_2) ~= sign(a_3)
    v3_p = v_2 + a_2^2/(2*u1);
else
    v3_p = v0;
end;
%!!!!!!!!!!!!!!!!!!!!!!

%!!!!!!!!!!!!!! v4_p:  obdobne jako u v0
if sign(af)~=sign(a_3)
    v4_p = vf - af^2/(2*u1);
else
    v4_p = vf;
end;
%!!!!!!!!!!!!!!!!!!!!!!

y.vmax = max([v0,v2_p,v3_p,v4_p,vf]);
y.vmin = min([v0,v2_p,v3_p,v4_p,vf]);
%!!!!!!!!!!!!!!!!!!!!!! end of v_values

function y = s_minmax_values(sol,u1,s0,v0,a0,sf,vf,af)
s1_p = s0;
s2_p = sf;
s3_p = s0;           %zatim dame neutralni hodnotu
m = 0;              %rika kolik mame novych s3_p
b = [u1,0,-u1,0,-u1,0,u1,0];
a_0 = a0;
v_0 = v0;
s_0 = s0;
coeff = zeros(8,4); %matice koeficientu viz. zprava
% hledame extremy funkce s(t) => musime najit takove casy, kde v(t) = 0
%projizdime vsechny intervaly t1..t7 + jeden krok abych pozdeji mohl zkontrolovat jestli reseni vede
do xf.
for k = 1:8
    ti = roots([b(k)/2 a_0 v_0]); %urcim koreny polynomu v(t) v danem casovem intervalu ti
    for i = 1:length(ti) %pro kazdy REALNY KLADNY koren-
        if isreal(ti(i)) && (ti(i)>=0) && (ti(i) <= sol(k))%a mensi nez dany casovy interval z
reseni 'sol'-
            m = m + 1;
            s3_p(m) = s_0 + v_0*ti(i) + a_0/2*ti(i)^2 + b(k)/6*ti(i)^3; %urcim novy mozny extrem
        end;
    end;
    coeff(k,:) = [b(k) a_0 v_0 s_0]; %ulozim pocatecni hodnoty stavu jako koeff polynomu
    s_0 = s_0 + v_0*sol(k) + a_0/2*sol(k)^2 + b(k)/6*sol(k)^3;%spoctu nove poc hodnoty pro dalsi
krok
    v_0 = v_0 + a_0*sol(k) + b(k)/2*sol(k)^2;
    a_0 = a_0 + b(k)*sol(k);
end;
y.smax = max([s1_p,s2_p,s3_p]);
y.smin = min([s1_p,s2_p,s3_p]);
y.coeff = coeff;
%!!!!!!!!!!!!!!!!!!!!!! end of s-values

%test AM
function y = testAM(bm,am,a0,t1,e)
am_prouzek = a0 + bm*t1;
if am_prouzek <= am + e;
    y = 1;
else
    y = 0;
end;

```



```

%test DM
function y = testDM(bm, dm, af, t7, e)
dm_prouzek = af - bm*t7;
if dm_prouzek >= -dm - e;
    y = 1;
else
    y = 0;
end;

function y = admissible_codomain(BM, AM, DM, VM, Smin, Smax, s0, v0, a0, sf, vf, af)
%test pripustnosti x0 a xf

%1) otestujem AM a DM
y = 0;
if (a0 <= AM) && (a0 >= -DM) && (af <= AM) && (af >= -DM)

else
    'spatne vstupni hodnoty a0 nebo af'
    y = 1;
end;

%2) otestujem VM
if y == 0
    pom1 = a0^2/(2*BM);
    pom2 = af^2/(2*BM);
    vrch1 = v0-pom1;    %VB+
    vrch2 = v0+pom1;    %VB-
    vrch3 = vf-pom2;    %VE+
    vrch4 = vf+pom2;    %VE-
    if (abs(vrch1)<=VM) && (abs(vrch2)<=VM) && (abs(vrch3)<=VM) && (abs(vrch4)<=VM)

    else
        'spatne vstupni hodnoty v0 nebo vf'
        y = 1;
    end;
end;

%otestujem Smin, Smax (krome codomain v-a otestujem zda jsou vstupni polohy
% v rozsahu smin, Smax)
if y == 0;
    if (s0 <= Smax) && (s0 >= Smin) && (sf <= Smax) && (sf >= Smin)

    else
        'spatne vstupni hodnoty s0 nebo sf'
        y = 1;
    end;
end;
end;

```



```

if A(index+i,:) >= 0 & isreal(A(index+i,:))
    pocet = pocet+1;
    Apositiv(pocet,:) = A(index+i,:);
    ok = v_minmax_values(A(index+i,:),BM,v0,a0,vf,af);
    if ok.vmax <= VM+e && ok.vmin >= -VM-e
        num_of_sol = num_of_sol+1;
        solution(num_of_sol,:) = A(index+i,:);
    else
        % 'set3_i je kladny ale necti VM'
    end;
end;
end;
index = index+length(r3y);
%!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

%set4
s4t1 = s1t1;
s4t4 = 0;
s4t5 = 0;
s4t6 = 0;

p0 = ((v0+vf)*2*BM+af^2-a0^2)*6*AM^2+((s0-sf)*3*BM^2+(-af*vf-a0*v0)*3*BM-af^3+a0^3)*8*AM+(-
    v0^2+vf^2)*12*BM^2+(af^2*vf+a0^2*v0)*12*BM+3*af^4-3*a0^4;
p1 = -48*BM*(1/2*af^2-1/2*AM*af+BM*vf)*(af-AM);
p2 = 12*BM^2*(-6*AM*af+AM^2+5*af^2+2*BM*vf);
p3 = -24*BM^3*(-AM+2*af);
p4 = 12*BM^4;
p = [p4 p3 p2 p1 p0];
p = facelift(p);
r4y = roots(p);

for i = 1:length(r4y)
    s4t7 = r4y(i);
    s4t3 = s1t3 + s4t7 - af/BM;
    s4t2 = (2*BM^2*s4t7^2+2*BM*(vf-v0-2*s4t7*af)+a0^2+af^2-2*AM^2)/2/BM/AM;
    A(index+i,:) = [s4t1 s4t2 s4t3 s4t4 s4t5 s4t6 s4t7];

    if A(index+i,:) >= 0 & isreal(A(index+i,:))
        pocet = pocet+1;
        Apositiv(pocet,:) = A(index+i,:);
        ok1 = v_minmax_values(A(index+i,:),BM,v0,a0,vf,af);
        ok2 = testDM(BM,DM,af,s4t7,e);
        if ok1.vmax <= VM+e && ok1.vmin >= -VM-e && ok2 == 1
            num_of_sol = num_of_sol+1;
            solution(num_of_sol,:) = A(index+i,:);
        else
            % 'set4_i je kladny ale necti VM nebo DM'
        end;
    end;
end;
index = index+length(r4y);
%!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

%set5
s5t3 = ((2*a0^2+4*BM*(VM-v0))^(1/2))/2/BM;
s5t1 = -a0/BM + s5t3;
s5t2 = 0;
s5t4 = (6*(2*(-v0-VM)*BM+a0^2)*DM*(2*a0^2+4*BM*(VM-v0))^(1/2)+6*(2*(-vf-VM)*BM+af^2)*DM^2+8*(3*(-
    s0+sf)*BM^2+3*(v0*a0-vf*af)*BM+af^3-a0^3)*DM+12*(vf^2-VM^2)*BM^2-
    12*BM*vf*af^2+3*af^4)/DM/BM^2/VM/24;
s5t5 = s1t5;
s5t6 = s1t6;
s5t7 = s1t7;
index = index+1;
A(index,:) = [s5t1 s5t2 s5t3 s5t4 s5t5 s5t6 s5t7];

if A(index,:) >= 0
    pocet = pocet+1;
    Apositiv(pocet,:) = A(index,:);
    ok = testAM(BM,AM,a0,s5t1,e);
    if ok == 1
        num_of_sol = num_of_sol+1;
        solution(num_of_sol,:) = A(index,:);
    else
        % 'set5 je sice kladny ale necti AM'
    end;
end;

```

```

end;
%!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

%set6
t1 = s5t1;
t2 = 0;
t3 = s5t3;
t5 = s2t5;
t6 = 0;
t7 = s2t7;
t4 = (2*(3*af^2-6*BM*(vf+VM))*t5*BM+2*(3*a0^2-6*BM*(v0+VM))*t3*BM+4*af^3-12*BM*vf*af-
      4*a0^3+12*a0*BM*v0-12*BM^2*(s0-sf))/BM^2/VM/12;
A(index+i,:) = [t1 t2 t3 t4 t5 t6 t7];

if A(index+i,:) >= 0
    pocet = pocet+1;
    Apositiv(pocet,:) = A(index+i,:);
    ok1 = v_minmax_values(A(index+i,:), BM, v0, a0, vf, af);
    ok2 = testDM(BM, DM, af, t7, e);
    ok3 = testAM(BM, AM, a0, t1, e);
    if ok1.vmax <= VM+e && ok1.vmin >= -VM-e && ok2 == 1 && ok3 == 1
        num_of_sol = num_of_sol+1;
        solution(num_of_sol,:) = A(index+i,:);
    else
%       'set6_i je kladny ale necti VM nebo DM nebo AM'
    end;
end;
index = index+4;
%!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

%set7 zkontrolovat AM,VM
s7t2 = 0;
s7t4 = 0;
s7t5 = 0;
s7t7 = s1t7;

p0 = ((v0+vf)*2*BM+a0^2-af^2)*6*DM^2+((s0-sf)*3*BM^2+(a0*v0+vf*af)*3*BM+a0^3-af^3)*8*DM+(v0^2-
      vf^2)*12*BM^2+(vf*af^2+v0*a0^2)*12*BM-3*af^4+3*a0^4;
p1 = 48*(DM+a0)*(1/2*DM*a0+BM*v0+1/2*a0^2)*BM;
p2 = 12*BM^2*(DM^2+6*DM*a0+5*a0^2+2*BM*v0);
p3 = 24*BM^3*(2*a0+DM);
p4 = 12*BM^4;
p = [p4 p3 p2 p1 p0];
p = facelift(p);
r7y = roots(p);

for i = 1:length(r7y)
    s7t1 = r7y(i);
    s7t3 = DM/BM + a0/BM + s7t1;
    s7t6 = (2*BM^2*s7t1^2+2*BM*(v0-vf+2*a0*s7t1)+a0^2+af^2-2*DM^2)/2/BM/DM;
    A(index+i,:) = [s7t1 s7t2 s7t3 s7t4 s7t5 s7t6 s7t7];

    if A(index+i,:) >= 0 & isreal(A(index+i,:))
        pocet = pocet+1;
        Apositiv(pocet,:) = A(index+i,:);
        ok1 = v_minmax_values(A(index+i,:), BM, v0, a0, vf, af);
        ok2 = testAM(BM, AM, a0, s7t1, e);
        if ok1.vmax <= VM+e && ok1.vmin >= -VM-e && ok2 == 1
            num_of_sol = num_of_sol+1;
            solution(num_of_sol,:) = A(index+i,:);
        else
%           'set7_i je kladny ale necti VM nebo AM'
        end;
    end;
end;
index = index+length(r7y);

%!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
%set8
s8t2 = 0;
s8t4 = 0;
s8t5 = 0;
s8t6 = 0;

p0 = 36*af^2*a0^2*BM*v0+48*af^3*BM*v0*a0-144*BM^3*v0*a0*s0-72*BM^2*vf*a0^2*v0-36*a0^2*af^2*BM*vf-
      144*BM^3*vf*af*sf+144*BM^3*vf*af*s0+48*BM*vf*af*a0^3-
      a0^6+17*af^6+9*a0^2*af^4+48*af^3*BM^2*sf-

```


*Nakonec je všechno dobré a pokud ne,
tak ještě není konec.*

LITERATURA

- [1] R. Abraham, J.E. Marsden, *Foundations of Mechanics*. 2nd Edition, Addison-Wesley, 1978.
- [2] A. C. Antoulas, New Results on the Algebraic Theory of Linear Systems: The solution of Cover Problems, *Linear Alg. and Appl.*, vol. 50, pp. 1-43, 1983.
- [3] K. J. Astrom, T. Hagglund, PID controllers: theory, design and tuning, *Instrument Society of America*, 1995.
- [4] M. Athans, P. Falb, *Optimal Control: An Introduction to the Theory and its applications*. McGraw-Hill, New York, 1966.
- [5] J. Baillieul et al., *Robotics*, Proceedings of Symposia in Applied Mathematics, Vol 41, American mathematical Society, Providence, Rhode Island, 1990.
- [6] Bayer, *Computational Algebraic Geometry Part One: Basic Tools*, Columbia University, Lecture Notes, Conference on Computers and Mathematics, Stanford University, 1986.
- [7] L. Bláha, Gröbnerova báze a teorie řízení, *Rigorózní práce*, 2010.
- [8] L. Bláha, *Stabilizace a automatické řízení pohybu dvoukolky*, Diplomová práce, Západočeská Univerzita v Plzni, Plzeň, 2007.
- [9] L. Bláha, Stabilization and motion control of a inverted pendulum on a two wheel cart. In *The 1st Young Researchers Conference on Applied Sciences*. pp. 218-223, University of West Bohemia in Pilsen, Pilsen, 2007. ISBN: 978-80-7043-574-8.
- [10] L. Bláha, M. Schlegel, Mechatronic System Design Applied to Double Inverted Pendulum with Flywheel Actuator. *Diffusion and Defect Data Pt.B:Solid State Phenomena*, Vol 147-149, No. 1, pp.272-277, 2009. ISSN 1012-0394
- [11] L. Bláha, M. Schlegel, J. Mošna, Optimal control of chain of integrators with constraints. *17th International Conference on Process Control 2009*, pp. 134-139. 2009.
- [12] A. M. Bloch, *Nonholonomic Mechanics and Control*, volume 24 of *Interdisciplinary Applied Mathematics*, Springer-Verlag, New York-Heidelberg-Berlin, ISBN 0-387-095535-6, 2003.
- [13] J. E. Bobrow, S. Dubowsky, J. S. Gibson, Time-Optimal Control of Robotic Manipulators Along Specified Paths, *The International Journal of Robotics Research* 4 (3), 3-17, 1985
- [14] W. Boege, R. Gebauer and H. Kredel, Some examples for solving systems of algebraic equations by calculating Gröbner bases. *Journal of Symbolic Computation*, Vol. 2, pp. 83-98, 1986.
- [15] V. F. Borisov, Fuller's phenomenon: Review, *In Journal of Mathematical Sciences*, Vol. 100, No. 4, 2000.
- [16] N. K. Bose, Two Decades of Gröbner Bases in Multidimensional Systems, *Special Semestr on Grobner Bases and Related Methods*, Austria, 2006.

- [17]A. E. Bryson, Y. C. Ho, *Applied Optimal Control*. New York: Blaisdell, 1969.
- [18]B. Buchberger, F. Winkler. *Gröbner Bases and Applications*. Volume 251 of *London Mathematical Society Series. Proc. of the International conference "33 years of Grobner Bases"*, Cambridge University Press, 1998.
- [19]B. Buchberger, *An algorithm for Finding the Bases Elements of the Residue Class Ring Modulo Zero Dimensional Polynomial Ideal* (german) PhD thesis, Univ. of Innsbruck, Austria, 1965.
- [20]B. Buchberger, A Criterion for Detecting Unnecessary Reductions in the Contruction of Gröbner Bases, *Proceedings of the International Symposium on Symbolic and Algebraic Manipulation*, Marseille, France, Vol. 72 of *Lecture Notes in Computer Science*, pp. 3-21, Springer, 1979.
- [21]B. Buchberger, *Gröbner-Bases: An Algorithmic Method in Polynomial Ideal Theory, Multidimensional Systems Theory*, edited by N. K. Bose, D. Reidel Publishing Company, Dordrecht, pp. 184-232, 1985.
- [22]B. Buchberger, Gröbner-Bases and System Theory, To appear as *Special Issue on Applications of Gröbner bases in Multidimensional Systems and Signal processing*, Kluwer Academic publishers, Vol. 12, pp. 223-251, 2001.
- [23]B. Buchberger, *Gröbner Bases: An Algorithmic Method in Polynomial Ideal Theory. In Recent Trends in Multidimensional Systems Theory*, edited by N. K. Bose. Chapter 6, pp. 184-232. D. Reidel Publishing Company, 1986.
- [24]B. Buchberger, Gröbner Bases: A Short Introduction for System Theorists.
- [25]S.S.L. Chang, *Synthesis of Optimum Control Systems*, McGraw-Hill Book Company, New York, 1961.
- [26]S.S.L. Chang, Minimal Time Control with Multiple Saturation Limits. *IEEE Transaction on automatic control*, pp. 35-42, 1963.
- [27]Ch. Charoenlarnnoppa, N. K. Bose, Multidimensional FIR Filter Bank Design Using Gröbner Bases, *IEEE Transactions on Circuits and Systems-II: Analog and digital Processing*, Vol. 46, No. 12, 1999.
- [28]S. Collart, M. Kalkbrener, D. Mall, Converting Bases with Gröbner Walk. *Journal of Symbolic Computation*, Vol. 24, pp. 465-469, 1997.
- [29]D. Cox, J. Little, and D. O'Shea, *Ideals, Varieties, and Algorithm: An Introduction to Computational Algebraic Geometry and Commutative Algebra*, 2nd. ed. New York: Springer Verlag, 1996.
- [30]M. Davis. Hilbert's tenth problem is unsolvable. *Amer. Math. Monthly*, Vol. 80, pp. 233-269. 1973.
- [31]L. E. Dickson. Finiteness of the Odd Perfect and Primitive Abundant Numbers with n Distinct Prime Factors. *American Journal of Mathematics*, Vol. 35, pp. 413-426, 1913.
- [32]B. Ezair, T. Tassa, Z. Shiller, Multi-axis High-order Trajectory Planning, International Conference on Intelligent Robots and Systems, IROS 2012.
- [33]F. Fahimi, *Autonomous Robots; Modeling, Path Planning, and Control*, Springer Science+Business Media, LLC 2009.
- [34]J. C. Faugere, A new efficient algorithm for computing Gröbner bases (F4). *Journal of pure and Applied Algebra*, Vol. 139, No. 1, pp. 61- 88, 1999.
- [35]J. C. Faugere, A new efficient algorithm for computing Gröbner Bases without reduction to zero (F5). *ISSAC '02*, pp. 75-83, New York, 2002.
- [36]J. C. Faugere, Gianni, Lazard, and Mora, Efficient computation of zero-dimensional Grobner bases by change of ordering, *Journal of Symbolic Computation*, Vol. 16, pp. 377-399, 1993.
- [37]J. Freeke, *Linking Groebner Bases and Toric Varieties*, Glasgow, 2009.
- [38]K. Fukuda, A. N. Jensen, N. Lauritzen, R. Thomas, The generic Gröbner Walk, Institute of Operations Research, 2008.

- [39]A. T. Fuller, Dimensional Properties of Optimal and Sub-optimal Nonlinear Control Systems, *Journal of the Franklin Institute*, Vol. 289, No. 5, pp. 379-393, 1970.
- [40]A. T. Fuller, Sub-optimal Nonlinear Controllers for Relay and Saturating Control Systems, *International Journal of Control*, Vol. 13, No. 3, 401-428, 1971.
- [41]A. T. Fuller, Simplification of Some Time-Optimal Switching Functions, *Technical Notes and Correspondence*, 1973.
- [42]R.F. Hartl, P. S. Suresh, and G. V. Raymond, A survey of the maximum principles for optimal control problems with state constraints, *SIAM review* 37.2, 1995.
- [43]R. Haschke, E. Weitnauer, and H. Ritter, On-Line Planning of Time-Optimal, Jerk-Limited Trajectories, *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pp.3248–3253, 2008.
- [44]M. Heng, Smooth and time optimal trajectory generation for high-speed machine tools, University of Waterloo, 2008.
- [45]D. Hilbert, *Über die Theorie der algebraischen formen*, Math. Annalen Vol. 36, pp. 473-534, 1890. Reprinted in *Gesammelte Abhandlungen*, vol II, Chelsea, New York, 1965.
- [46]Han-Pang Huang; N.H. McClamroch, Time-optimal control for a robotic contour following problem, *Robotics and Automation, IEEE Journal of*, vol.4, no.2, pp.140,149, Apr 1988.
- [47]A. Iglesias. B-Splines and NURBS curves and surfaces, *Lectures in computer-aided geometric design and surfaces*, University of Cantabria, 2001.
- [48]A. N. Jensen, *Gfan version 0.4: A User's Manual*, 2009.
- [49]N. Karcnias and D. Vafiadis, On the cover problems of geometric theory, *Kybernetika*, vol. 29. no.6, pp. 547-562, 1993.
- [50]N. Karcnias, Matrix pencil approach to geometric system theory, *Proc. IEE*, Vol. 126, No. 6, pp. 585-590, 1979.
- [51]N. Karcnias, J. Leventides, Grassmann Invariants, Matrix Pencils, and Linear System Properties, *Linear Algebra and its Applications*, pp. 705-731, 1996.
- [52]W. Khalil, D. Etienne. *Modeling, identification and control of robots*. Butterworth-Heinemann, 2004.
- [53]H.K. Knudsen, An Iterative Procedure for Computing Time-Optimal controls, *IEEE Transaction on automatic control*, pp. 23-30, 1964.
- [54]J. Königsmarková, *Přiřazení pólů lineárnímu dynamickému systému výstupní zpětnou vazbou*, Bakalářská práce, Západočeská Univerzita v Plzni, Plzeň, 2010.
- [55]T. Kröger, On-line trajectory generation: Nonconstant motion constraints, *Robotics and Automation (ICRA), 2012 IEEE International Conference on*, vol., no., pp.2048,2054, 14-18 May 2012.
- [56]T. Kröger, F. M. Wahl, On-Line Trajectory Generation: Basic Concepts for Instantaneous Reactions to Unforeseen Events, *IEEE Trans. on Robotics*. Vol. 26, No. 1, pp. 94-111, 2010.
- [57]T. Kröger, A. Tomiczek, and F. M. Wahl, Towards On-Line Trajectory Computation. In *Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 736-741, Beijing, China, October 2006.
- [58]V. Kučera, Diophantine Equation in Control-A Survey. *Automatica*, Vol. 29, No. 6, pp. 1361-1375, 1993.
- [59]T. Kunz, M. Stilman, Time-Optimal Trajectory Generation for Path Following with Bounded Acceleration and Velocity, In: *Robotics: Science and Systems*, 2012.
- [60]K.J.Kyriakopoulos, G.N.Saridis, Minimum Jerk Path Generation', *IEEE 1988 International Conference on Robotics and Automation*, Philadelphia, PA, USA, April 1988.
- [61]N. Lauritzen, Grobner Bases (and the Frobenius problem) – a gentle introduction. Denmark, 2004.
- [62]G. LEE, Y, J. Byung, D. KIM; Z. CHOI,: New robotic motion generation using digital convolution with physical system limitation.. In: *CDC-ECC: IEEE*, 2011.

- [63]V. Levandovsky, E. Alvarez, *Obstructions to Genericity in Study of Parametric Problems in Control Theory*, 2007.
- [64]Z. Lin, Output Feedback Stabilizability and Stabilization of Linear n-D Systems, In *Multidimensional Signals, Circuits and Systems*, Taylor and Francis, Vol. 4, pp. 59-76, 2001.
- [65]A. Locatelli, *Optimal Control: An Introduction*. Basel, Boston, Berlin; Birkhäuser Verlag, 2001.
- [66]E. Mansfield, *Differential Gröbner Bases: Ph. D Thesis*, University of Sydney, 1991.
- [67]Maplesoft, a division of Waterloo Maple Inc. 2011.
<http://www.maplesoft.com/support/help/Maple/view.aspx?path=Groebner/MonomialOrders>
- [68]N. Marchand, A. Hably, Global stabilization of multiple integrators with bounded controls, *Automatica*, Vol. 41, pp. 2147-2152, 2005.
- [69]J. E. Marsden, A. Weinstein, *Calculus III*. Springer-Verlag, New York, 1985.
- [70]C. McKay, *An analysis of improvements to buchberger's algorithm for groebner basis computation. Master's thesis*, University of Maryland, College Park, 2004.
- [71]J. Melichar, *Lineární systémy 1*, Západočeská Univerzita v Plzni, 2004. Pomocný učební text.
- [72]J. Melichar, *Lineární systémy 2*, Západočeská Univerzita v Plzni, 2004. Pomocný učební text.
- [73]F. C. Moon, *Applied Dynamics: With Applications to Multibody and Mechatronic Systems*, WILEY-WCH Verlag GmbH & Co. KgaA, Weinheim, ISBN 0-471-13828-2, 2004.
- [74]T. Pajdla, *Inteligentní robotika*, Praha, 2008. Pomocný učební text.
- [75]L.Y. Pao, G.F. Franklin, Proximate Time-Optimal Control of Third-Order Servomechanisms. *IEEE Transaction on automatic control*, vol. 38, No. 4, pp.560-580, 1993.
- [76]J. G. Papastavridis, *Analytical Mechanics: A Comprehensive Treatise on the Dynamics of Constrained Systems; for Engineers, Physicists, and Mathematicians*, Oxford University Press, Walton Street, Oxford, 2002. ISBN 0-19-512697-1
- [77]P. A. Parrilo, ACM 270 Algebraic techniques and semidefinite optimization, *Lecture notes*, CALTECH, 2007.
- [78]R. Paul, *Robot Manipulators: Mathematics, Programming and Control*, MIT Press, Cambridge, Massachusetts, 1981.
- [79]E. V. Persson, Synthesis of Control Systems operating linearly for Small Signals and approximately 'Bang-Bang' for Large Signals, *Proc. IFAC Congress, Basle, Theory*. pp. 210, 1963.
- [80]L. Piegl, W. Tiller, *The NURBS Book*, Springer, 1997.
- [81]PLCopen, Technical committee 2, Function blocks for motion control, version 2, 2011.
- [82]P. Pták. *Introduction to Linear Algebra*. Vydavatelství ČVUT, Praha, 1997.
- [83]E. P. Ryan, Synthesis of a third-order time fuel optimal control system. Control theory centre. University of Warwick. Report No. 67. 1977.
- [84]E. P. Ryan, Synthesis of a Third-Order Time Fuel Optimal Control System. *IEEE Transaction on automatic control*, Vol. ac-23, No. 5, pp.952-954, 1978.
- [85]E. P. Ryan, On the Sensitivity of a Time-Optimal Switching Function. *IEEE Transaction on automatic control*, Vol. ac-25, No. 2, pp.275-277, 1980.
- [86]T. K. Sakar, O. Pereira, Using the Matrix Pencil Method to Estimate the Parameters of a Sum of complex Exponentials, *IEEE Antennas and Propagation Magazine*, Vol. 37, No. 1, 1995.
- [87]L. Sciavicco, B. Siciliano, *Modeling and Control of Robot Manipulators*, 2-nd ed. Springer-Verlag, 2005.
- [88]M. Schlegel, *Modální řízení konečněrozměrných lineárních systémů*, Západočeská Univerzita v Plzni, Plzeň, 2005.
- [89]M. Schlegel, L. Bláha, Parametrization of all three impulse zero vibration input shapers. *Mechatronic Systems and Materials: Mechatronic Systems and Robotics*, Vol 2010, No. 164, pp.271-277, 2010. ISSN 1012-0394.
- [90]M. Schlegel, J. Königsmarková, Parametric Jordan Form Assignment Revisited, *IFAC World Congress*, Milano, Italy, 2011.

- [91]Z. Shiller, On Singular Time Optimal Control Along Specified Paths, *IEEE Transactions of Robotics and Automation*, Vol. 10, No. 4, pp. 561-566, 1994.
- [92]H. S. Shin, S. Lall, Optimal Decentralized Control of Linear Systems via Groebner Bases and Variable Elimination, *American Control Conference*, 2010.
- [93]J.J. E. Slotine, H. S. Yang, Improving the efficiency of timeoptimal path following algorithms, *IEEE Transactions on Robotics and Automation*, 5 (1): 118-124, February 1989.
- [94]E. D. Sontag, H. J. Sussmann, Time-optimal control of manipulators, *Robotics and Automation. Proceedings. 1986 IEEE International Conference on*, vol.3, no., pp.1692,1697, Apr 1986.
- [95]M.W. Spong, S. Hutchinson, M. Vidyasagar, *Robot Modeling and Control*, John Wiley and Sons, 2006.
- [96]B. Sturmfels, Computing final polynomials and final syzygies using Buchberger's Grobner bases method, *Results Math.* Vol.15, pp. 351-360, 1989.
- [97]B. Sturmfels, *Gröbner Bases and Convex Polytopes*, University Lecture Series 8, Amer. Math. Soc., Providence, RI, 1989.
- [98]B. Sturmfels, Two Lectures on Gröbner Bases *New horizon in Undergraduate Mathematics*, VMath Lecture Series, Mathematical Science Research Institute, Berkeley, California, 2005.
- [99]B. Sturmfels, What is a Gröbner Basis?, *Notices of the AMS*, Vol. 52, No. 10, 2005.
- [100]M. Sugita, M. Kawazoe, H. Imai, Relation between XL algorithm and Gröbner Bases Algorithms.
- [101]H. J. Sussmann, *Nonlinear controllability and optimal control*. Vol. 133. CRC Press, 1990.
- [102]H. J. Sussmann, An introduction to coordinate-free maximum principle, *Geometry of Feedback and Optimal Control*, B.Jakubczyk and W. Respondek, editors pp.463-557, New York, 1997.
- [103]H. J. Sussmann, New theories of set-valued differentials and new versions of the maximum principle of optimal control theory, *Nonlinear Control in the Year 2000*, n. 259 in Lecture Notes in Control and Information Sciences, Springer-Verlag, 2000.
- [104]D. Vafiadis, N. Karcanias, The Solution of the Cover Problem using Gröbner Basis Techniques, Greece, 1994.
- [105]J. G. Vargas, I. H. Hermoso, J. M. Morales, J. M. U. Enriquez, *Sudokus and Grobner bases: not only a Divertimento*.
- [106]U. Walther, T.G. Tryphon, A. Tannenbaum, On the Computation of Switching Surfaces in Optimal Control: A Gröbner Basis Approach, *IEEE Transaction on automatic control*, Vol. 46, No. 4, pp.534-540, 2001.
- [107]CH. H. Wang, J. G. Horng, Constrained Minimum-Time Path Planning for Robot Manipulators Via Virtual Knots of the Cubic B-Spline Functions, *IEEE Transaction on automatic control*, Vol. 35, No. 5, pp.573-577, 1990.
- [108]E. W. Weisstein, Sylvester Matrix, From *MathWorld*—A wolfram Web Resource. <http://mathworld.wolfram.com/SylvesterMatrix.html>
- [109]L. Xu, J. Q. Ying, and O. Saito, "Feedback Stabilization for a Class of MIMO nD Systems by Grobner Basis Approach," In *Abstract of First International Workshop on Multidimensional Systems*, pp. 88-90. Poland, 1998.
- [110]L. Xu, Synthesis of Multidimensional Control Systems by Gröbner Basis Approach, *SICE Annual Conference 2008*, pp. 2256-2261, Japan, 2008.
- [111]X. Xu, P. J. Antsaklis, On time optimal control of integrator switched systems with state constraints, *Nonlinear Analysis*, Vol. 62, pp. 1453-1465, 2005.
- [112]M. Zefran, Review of the Literature on Time-Optimal Control of Robotic Manipulators, *Technical Report MS-CIS-94-30*, University of Pennsylvania, Philadelphia, PA, May 1994.
- [113]Q. Zhang, S Li, Minimum time feed-rate optimization along predefined toolpath with acceleration constraints on each axis, *Control Conference (CCC), 2011 30th Chinese*, pp.5462,5467, 22-24 July 2011.

SEZNAM PUBLIKOVANÝCH I NEPUBLIKOVANÝCH PRACÍ AUTORA

4.4.1 PUBLIKACE

BLÁHA, L.; SCHLEGEL, M.; KÖNIGSMARKOVÁ, J. Design and Control of Underwater Vehicle for NDT inspections. In Oceanic Engineering Society - IEEE AUV 2014.

BLÁHA, L. Multi-axis Time Synchronization for Uncoordinated Motion Planning with Hard Constraints. In IFAC 2014.

BLÁHA, L. Stabilization of Variable Rope Length Pendulum using Lyapunov Theory and Differential Geometry. In *Applied Mechanics and Materials Vols. 152-154* s.618-623, 2012. Online available on www.scientific.net/AMM.152-154.618

SCHLEGEL, M.; BLÁHA, L. Parametrization of all three impulse zero vibration input shapers. In *Solid State Phenomena Vol. 164*, s. 271-277, 2010. Online available on www.scientific.net/SSP.164.271

BLÁHA, L.; SCHLEGEL, M. Mechatronic System Design Applied to Double Inverted Pendulum with Flywheel Actuator. In *Mechatronic Systems and Materials III Vols.147-149*. 2009, s.272-277, ISSN 1662-9779.

BLÁHA, L.; SCHLEGEL, M.; MOŠNA, J. Optimal control of chain of integrators with constraints. In The 17th International Conference on Process Control 2009. Štrbské Pleso : Slovak University of Technology in Bratislava, 2009. s. 51-56. ISBN: 978-80-227-3081-5.

BLÁHA, L.; SCHLEGEL, M. Tracking Control Via Dynamic Inversion. In *Process Control 2008*. Pardubice: University of Pardubice, 2008. s. 1-8. ISBN 978-80-7395-077-4.

BLÁHA, L. Stabilization and motion control of a inverted pendulum on a two wheel cart. In The 1st Young Researchers Conference on Applied Sciences. Plzeň: Západočeská univerzita, 2007. s. 218-223. ISBN 978-80-7043-574-8.

4.4.2 OSTATNÍ PRÁCE

Návrh ultralehkého robotického ramene pro měřicí, kalibrační a vysokorychlostní aplikace, *zpráva k projektu: Výukový model pro robotiku (VS-14-019)*, 2014.

Stabilizace a automatické řízení pohybu ponorného robotického zařízení, *příloha k průběžné zprávě – Projekt: Nová robotická dálkově ovládaná technologie pro diagnostiku a opravu ponořených zařízení (TA02020414)*, 2014.

Návrh ultrazvukového modulu pro nedestruktivní testování svarů ponořených technologií, *příloha k průběžné zprávě – Projekt: Nová robotická dálkově ovládaná technologie pro diagnostiku a opravu ponořených zařízení (TA02020414)*, 2014.

Návrh konstrukce ponorného robotického zařízení pro diagnostické účely, *příloha k průběžné zprávě – Projekt: Nová robotická dálkově ovládaná technologie pro diagnostiku a opravu ponořených zařízení (TA02020414)*, 2014.

Systémy statického ponořování, *příloha k průběžné zprávě – Projekt: Nová robotická dálkově ovládaná technologie pro diagnostiku a opravu ponořených zařízení (TA02020414)*, 2013.

Návrh mechanismu polohování přídavné zátěže pro nastavení referenční orientace ponorky v prostoru, *příloha k průběžné zprávě – Projekt: Nová robotická dálkově ovládaná technologie pro diagnostiku a opravu ponořených zařízení (TA02020414)*, 2013.

Rozbor umístění pohonů a jejich počtu vzhledem k počtu stupňů volnosti a možnostem pohybu ponorky, *příloha k průběžné zprávě – Projekt: Nová robotická dálkově ovládaná technologie pro diagnostiku a opravu ponořených zařízení (TA02020414)*, 2013.

Odhad potřebného výkonu pohonů, *příloha k průběžné zprávě – Projekt: Nová robotická dálkově ovládaná technologie pro diagnostiku a opravu ponořených zařízení (TA02020414)*, 2012.

Model ponořeného tělesa v kapalině s pohyblivou bodovou hmotou, *příloha k průběžné zprávě – Projekt: Nová robotická dálkově ovládaná technologie pro diagnostiku a opravu ponořených zařízení (TA02020414)*, 2012.

Hamiltonovský model ponořeného tělesa v kapalině, *příloha k průběžné zprávě – Projekt: Nová robotická dálkově ovládaná technologie pro diagnostiku a opravu ponořených zařízení (TA02020414)*, 2012.

Stabilization of variable rope length pendulum with hard structural constraints, 2012.

Gröbnerova báze a teorie řízení, *Rigorózní práce*, 2010.

Beauty of Mathematics in Robotics and Control, 2010.

REX - Connection of Control Theory and Practical Applications, 2010.

Stabilizace a automatické řízení pohybu dvoukolky, *Diplomová práce*, 2007.