



**VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ**

BRNO UNIVERSITY OF TECHNOLOGY

**FAKULTA STROJNÍHO INŽENÝRSTVÍ**

FACULTY OF MECHANICAL ENGINEERING

**ÚSTAV MECHANIKY TĚLES, MECHATRONIKY A BIOMECHANIKY**

INSTITUTE OF SOLID MECHANICS, MECHATRONICS AND BIOMECHANICS

**NOVÉ HYBRIDNÍ METODY PRO ROBUSTNÍ A  
AUTOMATIZOVANÝ ODHAD PARAMETRŮ  
MECHATRONICKÝCH SYSTÉMŮ**

NEW HYBRID METHODS FOR ROBUST AND AUTOMATED PARAMETER ESTIMATION OF MECHATRONIC SYSTEMS

**ZKRÁCENÁ VERZE PHD THESIS**

**OBOR**

**Inženýrská mechanika**

**AUTOR PRÁCE**

**Ing. Jan Najman**

**ŠKOLITEL**

**doc. Ing. Robert Grepl, Ph.D.**

**OPONENTI**

**DATUM OBHAJOBY**

**BRNO 2022**

## **KLÍČOVÁ SLOVA**

optimalizace, odhad parametrů, hybridní algoritmus, mechatronický model, MATLAB, Simulink, Simscape

## **KEY WORDS**

optimization, parameter estimation, hybrid algorithm, mechatronic model, MATLAB, Simulink, Simscape

Místo uložení rukopisu:

Knihovna VUT-FSI v Brně, Technická 2, 616 69 Brno.

Jan Najman, 2022

ISBN 80-214-

ISSN 1213-4198

# Obsah

---

1	Úvod.....	5
2	Současný stav řešené problematiky .....	6
2.1	Úloha estimace parametrů .....	6
2.2	Optimalizační algoritmy .....	7
3	Formulace problému a cílů dizertační práce .....	10
4	Modely mechatronických systémů.....	12
4.1	Volba vstupů do modelů .....	17
4.2	Vizualizace prostoru parametrů pro vybrané případy .....	18
5	Optimalizační algoritmy a nová metoda .....	19
5.1	Návrh srovnávacího testu.....	19
5.2	Výsledky srovnávacího testu .....	20
5.3	Požadavky pro tvorbu nového algoritmu.....	21
5.4	Design nového algoritmu.....	21
5.5	Testování nového hybridního algoritmu a srovnání s jinými metodami... 25	
5.6	Výsledky testů nového hybridního algoritmu a jejich zhodnocení .....	26
6	Pomocné algoritmy pro odhad parametrů.....	30
6.1	Detekce závislých parametrů metodou Monte Carlo .....	30
6.2	Rychlá detekce závislých parametrů .....	30
6.3	Detekce opakování signálu .....	31
6.4	Převzorkování dat .....	33
7	Nový nástroj pro odhad parametrů v inženýrské praxi.....	34
8	Závěr.....	35
9	Seznam zdrojů a použité literatury.....	37
	Životopis autora.....	38
	Abstrakt .....	39
	Abstract.....	39



# 1 ÚVOD

Jedním z problémů, kterým je třeba se zabývat v oblasti simulace, optimalizace a řízení jakýchkoliv reálných systémů, je odhad neznámých parametrů modelu na základě naměřených vstupně-výstupních dat.

Bez spolehlivého modelu zkoumaného systému totiž nelze využít pokročilých technik návrhů a testování řídicích systémů jako například HIL (Hardware-in-the-loop<sup>1</sup>) či dopředné řízení.

V technické praxi je zcela běžné, že jeden či více parametrů nelze na reálném systému změřit či vyčíst z tabulkových hodnot. V takovém případě je nutné parametry odhadovat a na základě srovnání odezvy simulovaného modelu s daty naměřenými na reálném systému iterativně dojít k parametrům, u kterých bude odchylka minimální.

Do této oblasti také spadá problematika vhodné struktury modelu, množství použitých parametrů a citlivosti na jejich změnu.

Navzdory tomu, že v oblasti teorie optimalizačních metod existuje nespočet publikací, studií a jiných vědeckých prací, narážíme v technické praxi na problémy, které jsou spojené se specifickými vlastnostmi konkrétního řešeného problému. Je totiž jasné, že žádná z metod nebude efektivně použitelná univerzálně ve všech případech.

Dalším častým problémem je nevhodná podoba vstupních (naměřených) dat, či již zmíněný počet parametrů.

Mezi hlavní směry v této oblasti, kterými má smysl se zabývat, potom patří snaha o výběr vhodných optimalizačních metod pro specifickou sadu modelů, se kterými se často setkáváme v technické praxi mechatronického oboru. Důležitý je také důraz na automatický běh optimalizačního algoritmu s minimálními požadavky na zásah na uživatele a maximalizace pravděpodobnosti úspěšného odhadu i bez hlubšího porozumění principu funkce použitých optimalizačních metod. V neposlední řadě je pak vhodné zkoumat i možnosti identifikace nedostatků v samotném zadání řešené úlohy, týkající se například formátu vstupních dat, vzájemné provázanosti hledaných parametrů modelu.

---

<sup>1</sup> HIL je technika pro vývoj real-time vestavěných řídicích jednotek, která využívá simulovaného modelu řízeného systému namísto reálného (např. matematický model chování motoru, namísto reálného motoru).

## 2 SOUČASNÝ STAV ŘEŠENÉ PROBLEMATIKY

### 2.1 ÚLOHA ESTIMACE PARAMETRŮ

Přestože z hlediska teorie jsou problémy nastíněné v úvodní kapitole nezávislé na použitém softwarovém nástroji, pro potřeby praktického vývoje a testování jednotlivých tezí a možných řešení je vhodné se zaměřit na konkrétní vývojové prostředí, jeho současné možnosti a případně již implementované funkce z kategorie optimalizace.

V současné době je v akademické i průmyslové sféře jedním z nejčastěji používaných softwarových nástrojů, v oblasti simulace a návrhu řízení, program MATLAB. Z tohoto důvodu má smysl se zabývat i jeho možnostmi v oblasti odhadu parametrů modelů. Veškerý vývoj v rámci této práce tedy bude probíhat právě v MATLABu. Stejně tak při zkoumání dostupných řešení se zaměříme pouze na algoritmy implementované v tomto programu.

Jak již bylo zmíněno v úvodu, úloha odhadu neznámých parametrů modelu reálného systému se skládá z několika hlavních částí, které lze do určité míry oddělit. Těmto částem se budou věnovat následující kapitoly. Stručně je lze rozdělit do níže uvedených bodů:

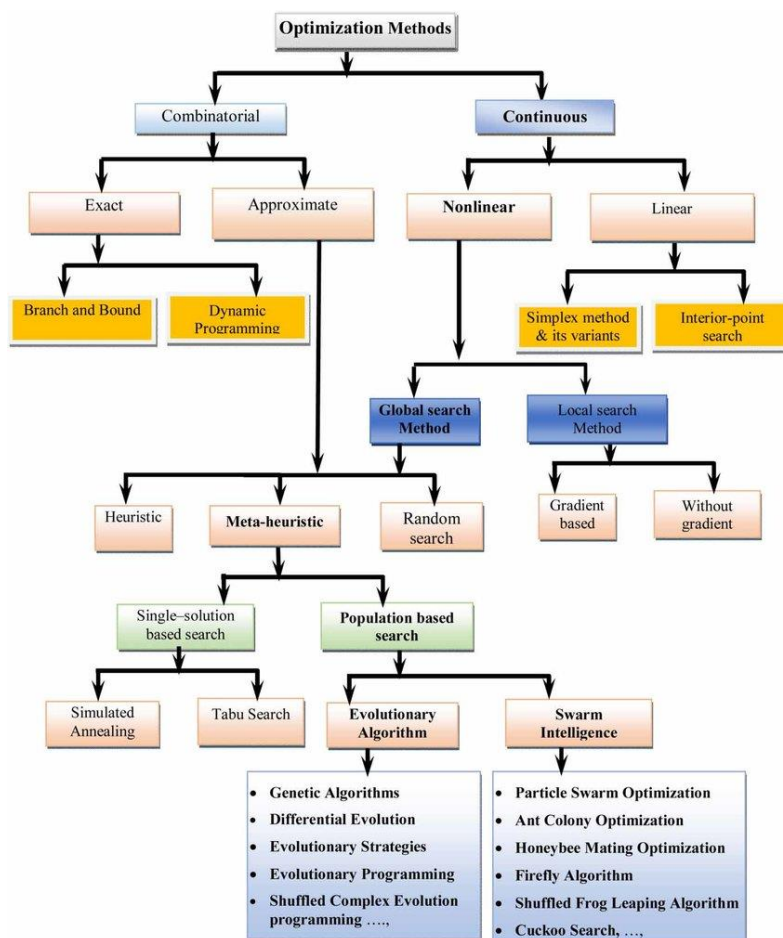
- Vytvoření matematického modelu systému, který co nejvěrněji napodobuje chování reálného systému a obsahuje neznámé parametry.
- Naměření dat na reálném systému.
  - Vstupní data (buzení), případně vychýlení systému z rovnovážného stavu.
  - Výstupní data (reakce systému).
- Nastavení úlohy pro nalezení neznámých parametrů modelu tak, aby simulovaný systém odpovídal reálnému.
  - Volba hodnoticí funkce, která měří „kvalitu“ nalezených parametrů. Typicky kvantifikace rozdílů mezi výstupními signály reálného a simulovaného systému.
  - Volba počátečního odhadu hledaných parametrů, případně omezení prostoru, ve kterém se mohou jejich hodnoty pohybovat.
  - Volba optimalizačního algoritmu, který prohledává prostor možných řešení (hodnot neznámých parametrů). Algoritmus se snaží iterativně přiblížit k optimálnímu řešení, kdy bude hodnoticí funkce (rozdíl reálné a simulované odezvy) minimální, v ideálním případě nulová. K tomu je třeba v každém kroku optimalizace provést simulaci s aktuálně nastavenými hodnotami parametrů, aby byla získána odezva a mohla být kvantifikována hodnoticí funkce.

## 2.2 OPTIMALIZAČNÍ ALGORITMY

Optimalizačních algoritmů a jejich variant existuje velké množství, pro přehlednost je tedy vhodné je sdružovat do obecnějších kategorií. V literatuře můžeme nalézt mnoho publikací zabývajících se kategorizací optimalizačních algoritmů a analýzou vhodnosti jejich použití na různé druhy problémů [1], [2], [3].

Několik variant klasifikace můžeme vidět na obrázku níže (Obr. 1).

Vzhledem k tomu, že mnoho názvů algoritmů nemá zavedený český překlad, budeme v této práci převážně používat jejich anglický název. Díky tomu budou také označení v této práci odpovídat pojmům v anglicky psané odborné literatuře.



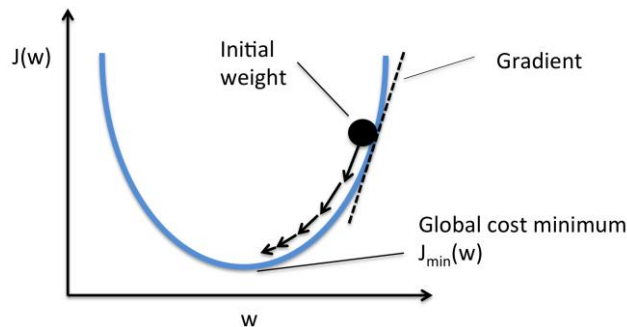
Obr. 1. Kategorizace optimalizačních metod dle [3].

Vzhledem k tomu, že tato práce je zaměřená na metody pro odhad parametrů v programu MATLAB, budeme v následujících kapitolách zkoumat pouze algoritmy, které jsou v tomto programu k dispozici již jako předpřipravené funkce.

### Gradientní metody

- fminunc (Unconstrained gradient solvers)
- fmincon (Constrained gradient solvers)
- lsqnonlin (Nonlinear least squares)

Základním principem funkce této skupiny optimalizačních metod, je znalost hodnoty optimalizované funkce v daném bodě a současně jejího gradientu, resp. její derivace. Na základě těchto hodnot je pak určena velikost a směr kroku pro další iteraci, dokud se algoritmus nedostane do lokálního minima. Ilustrační příklad principu funkce gradientního algoritmu je na Obr. 2. Výhodou je většinou velmi rychlá konvergence do lokálního minima.

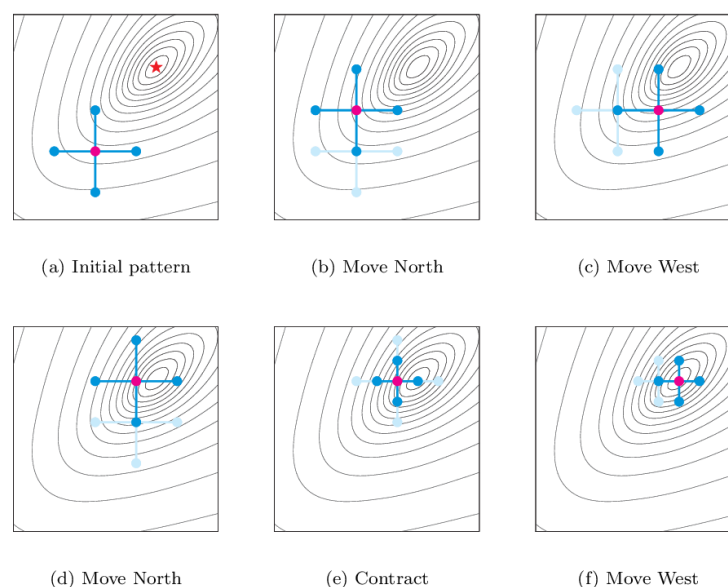


Obr. 2. Základní princip gradientních metod [4].

### Direct search metody (metody přímého prohledávání)

- fminsearch (Nelder-Mead simplex search)
- patternsearch (Pattern search)

Tyto metody na rozdíl od předchozí kategorie nevyžadují informaci o gradientu a vystačí si pouze s hodnotou funkce v daném bodě. Pro volbu kroku a směru prohledávání prostoru parametrů využívají testování sady bodů v okolí aktuálního bodu, dokud nenaleznou řešení s menší hodnotou kritériální funkce. Ilustrační příklad fungování takového algoritmu (konkrétně patternsearch) je na Obr. 3. Podrobnější přehled vlastností těchto metod je možné získat například z [5]. Výhodou je, že jsou odolnější vůči uvíznutí v lokálním minimu a fungují i v případě, že je průběh hodnotící funkce nespojitý.



Obr. 3. Příklad fungování direct search algoritmu (konkrétně patternsearch) [6].



## Heuristické a metaheuristické metody

- simulannealbnnd (Simulated annealing)
- ga (Genetic algorithm)
- particleswarm (Particle swarm optimization)
- GlobalSearch
- MultiStart

Obecně řečeno, heuristika znamená najít nebo objevit metodou pokus-omyl. *Meta-* zde znamená mimo nebo na vyšší úrovni. Metaheuristika má obecně lepší výsledky než jednoduchá heuristika. Metaheuristiku lze považovat za hlavní strategii, která řídí a modifikuje ostatní heuristiky. Všechny metaheuristické algoritmy využívají určitý kompromis mezi náhodným a lokálním hledáním.

Dvěma hlavními složkami metaheuristických algoritmů jsou: intenzifikace a diverzifikace. Diverzifikace znamená generování různorodých řešení tak, aby byl prozkoumán prohledávaný prostor v globálním měřítku, zatímco intenzifikace znamená zaměření prohledávání v lokální oblasti s vědomím, že se v této oblasti nachází aktuální dobré řešení.

Při výběru nejlepších řešení je třeba najít vhodnou rovnováhu mezi intenzifikací a diverzifikací, aby se zlepšila rychlost konvergence algoritmu. Výběr nejlepších zajišťuje, že řešení budou konvergovat k optimu, zatímco diverzifikace prostřednictvím randomizace umožňuje hledání v prostoru od lokálních optim a zároveň zvyšuje rozmanitost řešení. Vhodná kombinace těchto dvou hlavních složek obvykle zajistí, že je možné dosáhnout globální optimality.

## Náhradní model

- surrogateopt (Surrogate optimization)

Náhradním modelováním se označuje technika, která využívá vzorkování dat na původním modelu k vytvoření náhradního modelu, který jsou dostatečný k tomu, aby s relativně vysokou přesností předpovídal výstupy původního (typicky výpočetně náročného) modelu i v bodech mimo vzorkovaná data. [7]

V rámci problematiky optimalizace se pak těchto náhradních modelů využívá k aproximaci reálného tvaru hodnotící funkce. Díky tomu, že náhradní model je výpočetně nenáročný, střídá se během optimalizační úlohy fáze hledání vhodných bodů k prozkoumání na náhradním modelu s fází vyhodnocení skutečné hodnotící funkce v těchto bodech. Během střídání těchto dvou fází se náhradní model postupně zpřesňuje.

### 3 FORMULACE PROBLÉMU A CÍLŮ DIZERTAČNÍ PRÁCE

V předchozích kapitolách jsme uvedli hlavní aspekty úlohy odhadu parametrů matematických modelů, kterou se v této práci chceme zabývat. Dále byly představeny nejdůležitější nástroje a optimalizační metody v programu MATLAB, které jsou pro řešení této úlohy k dispozici. MATLAB byl zvolen z důvodu jeho rozšířenosti v inženýrské praxi i v oblasti výzkumu.

Na základě vlastních praktických zkušeností v této oblasti v rámci výzkumu a vývoje, a dále na základě zkušeností z výuky této látky v magisterském studijním programu<sup>2</sup> bylo vyzorováno několik hlavních problémů:

- I relativně jednoduché modely vytvořené pomocí nástrojů fyzikálního modelování mají nezanedbatelnou dobu výpočtu.
- Úloha hledání parametrů u mechatrických modelů není triviální, zejména s rostoucím počtem parametrů a nepřesností počátečního odhadu.
- V prostoru hledaných parametrů optimalizační úlohy se často vyskytují lokální minima.
- V případě omezených vstupních informací o matematickém modelu není snadné provést výběr vhodné optimalizační metody a nastavení úlohy, včetně počátečních odhadů parametrů a jejich limitů.
- V případě výběru lokálních optimalizačních metod, které konvergují relativně rychle, je nutné, aby uživatel úlohu neustále dozoroval, upravoval její nastavení či odhady parametrů, a znovu spouštěl.
- V případě výběru globálních optimalizačních metod je nutnost neustálých zásahů o něco menší, i tak však není snadné předem odhadnout rozsah úlohy. Pravděpodobnost nalezení globálního minima je vyšší než u lokálních metod, avšak výpočetní čas je často násobně vyšší, přičemž není snadné algoritmus přerušit během výpočtu.
- Nevhodnou volbou počtu odhadovaných parametrů či formátu dat naměřených na reálném systému si uživatel komplikuje úlohu a zvyšuje dobu výpočtu.

Na základě výše uvedených poznatků pak byly formulovány následující cíle dizertační práce:

#### 1. Vytvoření souboru technických modelů a jejich analýza

Pro vývoj efektivních postupů odhadu parametrů je výhodné zaměřit se pouze na vybrané typy modelů, na nichž lze nové algoritmy otestovat a zhodnotit výsledky. Na základě těchto poznatků pak lze z těchto modelů vycházet při práci se složitějšími systémy.

---

<sup>2</sup> Předmět RPO – Simulace a řízení v reálném čase (Studijní program Mechatronika)

Výstupem bude sada mechatronických modelů s konkrétními parametry. Tato sada by měla zahrnovat zejména mechanické a elektrické systémy, se kterými se v inženýrské praxi často setkáváme.

S pomocí těchto modelů bude poté probíhat další výzkum v rámci níže uvedených cílů, kdy proběhne jejich analýza a návrh vhodných postupů při odhadu jejich parametrů v praxi.

## **2. Kombinace globálních a lokálních optimalizačních metod pro mechatronické modely**

V praxi je na uživateli, aby správně nastavil optimalizační úlohu pro danou aplikaci což může vést ke komplikacím, způsobených výběrem nevhodné optimalizační metody, okrajových podmínek či počátečních odhadů. Tento praktický problém je ještě markantnější v případech, kdy má uživatel na počátku jen velmi málo informací o vlastnostech daného systému.

Jedním z cílů této práce je tedy výzkum a vývoj nových postupů pro identifikaci parametrů systému s využitím kombinace globálních a lokálních optimalizačních metod a jejich praktická implementace pro rychlejší a spolehlivější nalezení globálního optima.

Výstupem bude algoritmus hledání parametrů daného modelu kombinující globálního a lokální optimalizačních metody. Cílem je vytvořit algoritmus pro konkrétní sadu modelů (viz první cíl) s malou citlivostí na případné špatné počáteční odhady parametrů, vysokou pravděpodobností nalezení globálního optima, a co nerychlejší konvergencí k němu.

## **3. Vytvoření algoritmů pro detekci a analýzu nevhodně navržených úloh odhadu parametrů**

Jedním z důvodů neúspěšného či pomalého odhadu parametrů je kromě použité optimalizační metody i nevhodná struktura modelu, počet parametrů či podoba vstupních naměřených dat. Dalším cílem této práce je tedy výzkum a vývoj metod pro odhalení některých těchto nedostatků.

Výstupem bude sada algoritmů, které budou schopny alespoň částečně vyřešit otázky týkající se vzájemné závislosti parametrů modelu a jejich separovatelnosti, vhodného vzorkování vstupních naměřených dat a jejich redundance.

## **4. Nový nástroj pro odhad parametrů v inženýrské praxi**

Nástroje pro odhad parametrů v programu MATLAB mají v současné době omezenou funkcionalitu a pro netriviální případy musí uživatel nutně znát jejich úskalí a být schopen správně vyhodnotit jejich výsledky, aby se vyhnul jejich nevhodné interpretaci.

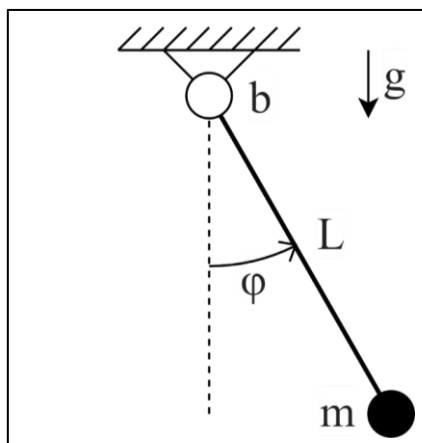
Výsledkem bude sada nástrojů (funkcí) v programu MATLAB, které prakticky implementují algoritmy vyvinuté v rámci předchozích cílů (2 a 3).

## 4 MODEL Y MECHATRONICKÝCH SYSTÉMŮ

### Jednoduché kyvadlo

Tento model představuje jednoduché kyvadlo (Obr. 4) a jako jediný z celé sady nevyužívá nástroje Simscape, pouze základních bloků sestavených na základě dynamické rovnice systému. Obsahuje hmotný bod na nehmotné tyči, která je uchycena do rotační vazby s viskózním třením.

Výhodou tohoto modelu je výrazně nižší výpočetní náročnost než u ostatních vytvořených systémů. Vybuzení modelu je docíleno změnou počátečního stavu.

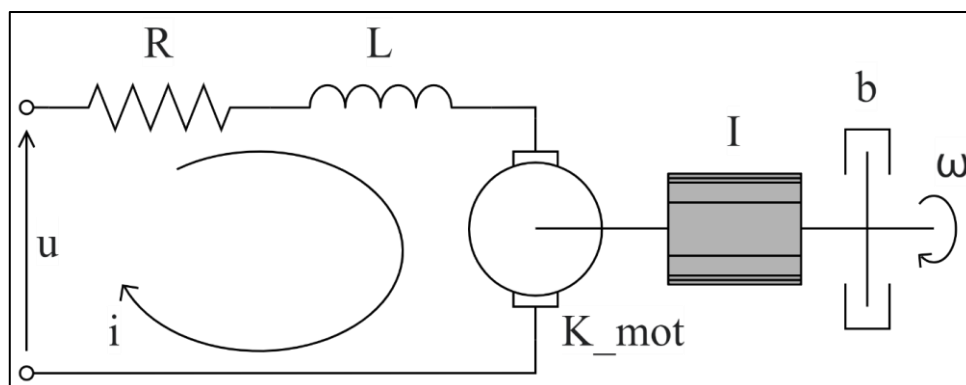


Obr. 4. Schematický diagram modelu Jednoduché kyvadlo.

### DC motor

Tento model představuje základní verzi stejnosměrného motoru (Obr. 5) a obsahuje hlavní prvky elektrické části (odpor, indukčnost vinutí), mechanické části (moment setrvačnosti, viskózní tření) a elektromechanickou přeměnu energie. Vstupem do systému je napětí na svorkách motoru, výstupem pak jeho úhlová rychlost.

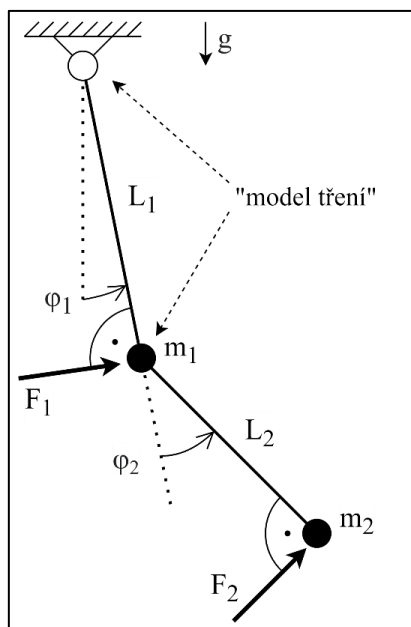
Model je vytvořen s využitím Simscape knihoven pro elektrické a mechanické systémy. Rozhraní mezi elektrickou a mechanickou částí je tvořeno blokem obecného rotačního elektromechanického převodníku.



Obr. 5. Schematický diagram modelu DC motor.

## Dvojité kyvadlo

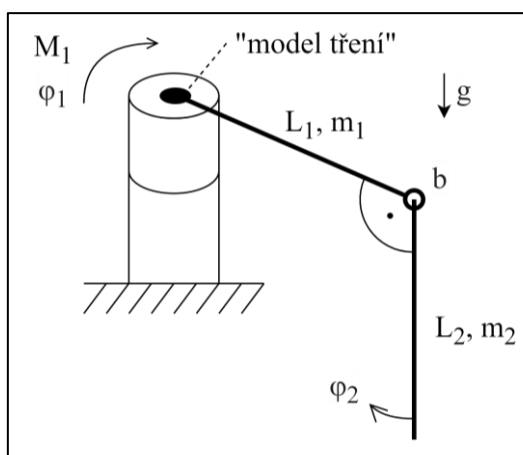
Tento model je rozšířením jednoduché varianty z předchozí kapitoly, avšak na rozdíl od něj, obsahuje o jedno rameno víc a také na něj mohou působit dvě vnější síly, které jsou vždy kolmé na dané rameno (viz Obr. 6). V obou rotačních vazbách je implementován model tření, který vychází z článku [8] a zahrnuje tření statické, Coulombovské, viskózní a tzv. Striebeckův efekt.



Obr. 6. Schematický diagram modelu Dvojité kyvadlo.

## Rotační kyvadlo

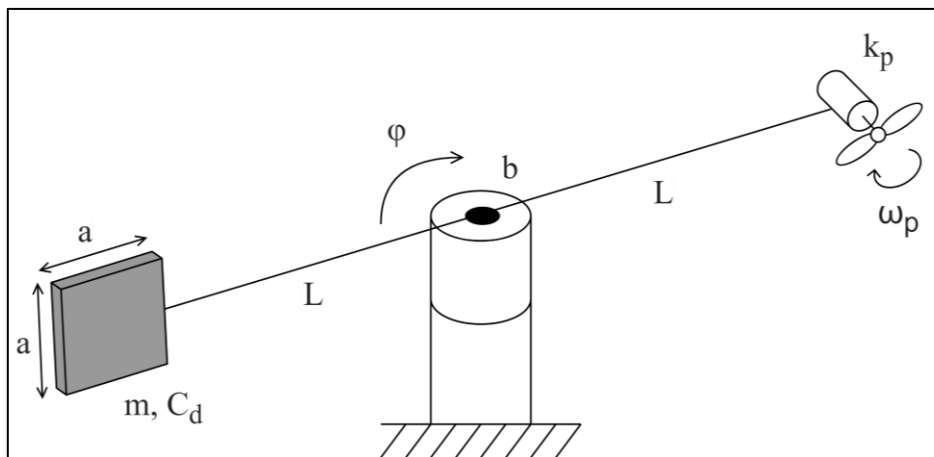
Rotační kyvadlo (Obr. 7) je v mnoha ohledech podobné předchozímu modelu, ovšem s několika odchylkami. Model tření, popsáný v předchozí kapitole, působí pouze v první rotační vazbě, ve druhé působí již jen viskózní tření. Také konfigurace ramen je modifikovaná, přičemž první rameno, hnané vstupním momentem, se pohybuje v horizontální rovině a druhé rameno se pohybuje v rovině kolmé na první.



Obr. 7. Schematický diagram modelu Rotační kyvadlo.

### Soustava s vrtulí (horizontální)

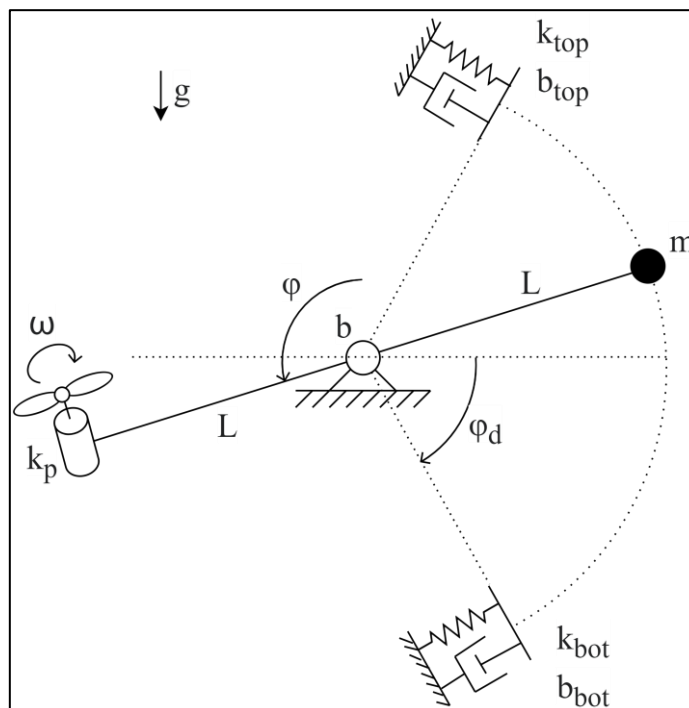
Soustava s vrtulí (Obr. 8) patří, co se týče složitosti mechanismu a počtu parametrů, mezi jednodušší modely. Je zde použit další druh buzení soustavy, kdy otáčky vrtule generují tahovou sílu, která roztáčí tyč, na jejímž druhém konci je umístěna hmotná deska. Proti tahu motoru působí viskózní tření v rotační vazbě a odpor vzduchu pohybem desky.



Obr. 8. Schematický diagram modelu Soustava s vrtulí (horizontální).

### Soustava s vrtulí (vertikální)

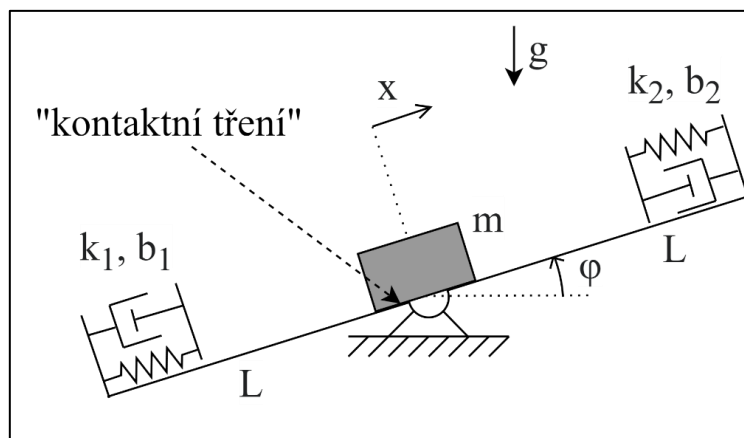
Model soustavy s vrtulí, pohybující se ve vertikální rovině, kombinuje několik prvků z předchozích modelů (kyvadlo, vrtulový pohon) a přidává navíc dva pružné dorazy s tlumením, které omezují pohyb soustavy na obou stranách (Obr. 9).



Obr. 9. Schematický diagram modelu Soustava s vrtulí (vertikální).

## Těleso na naklápěcí plošině

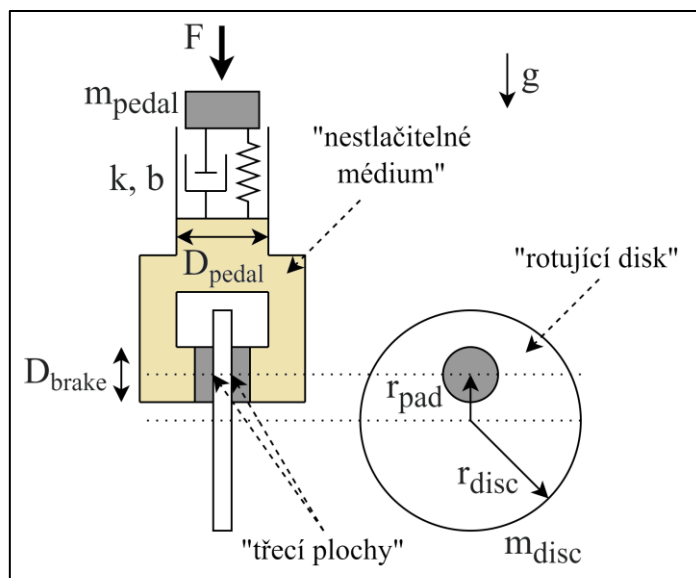
Tento model tělesa na naklápěcí plošině (Obr. 10) je částečně podobný předchozím modelům, kromě upravené topologie však obsahuje několik důležitých modifikací. Na rozdíl od většiny ostatních má tento model kinematické buzení. Kontakt tělesa s plošinou je realizován pomocí bloku „Spatial Contact Force“ z knihovny Simscape Multibody. V rámci tohoto bloku je modelován malý vzájemný průnik tělesa a plošiny, na základě něhož jsou počítány normálové a třecí síly.



Obr. 10. Schematický diagram modelu Těleso na naklápěcí plošině.

## Kotoučová brzda

Kotoučová brzda je model, se vstupní silou, která je přenášena přes hmotné těleso s pružinou a tlumičem na nestačitelné médium. To pak dále působí na brzdné destičky, které brzdí rotující disk (Obr. 11). Disk má na počátku nenulovou úhlovou rychlost. Model v Simulinku využívá bloků Simscape pro translační a rotační pohyb a také speciálního bloku „Disc Brake“, který modeluje chování kotoučové brzdy.

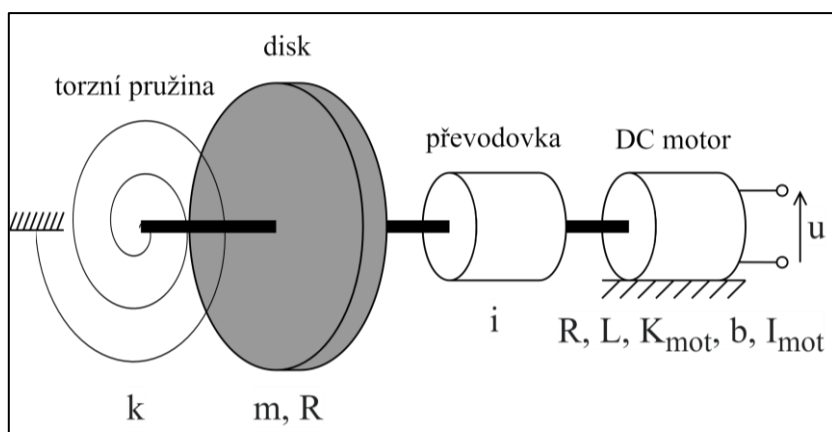


Obr. 11. Schematický diagram modelu Kotoučová brzda.

## DC motor + setrvačník na pružině

Poslední dvojice modelů je podobná v tom, že oba obsahují model DC motoru s převodovkou, u kterého je možné ovládat připojení a odpojení zdroje napětí. U této první varianty modelu pak zátěž tvoří hmotný disk s torzní pružinou (viz Obr. 12).

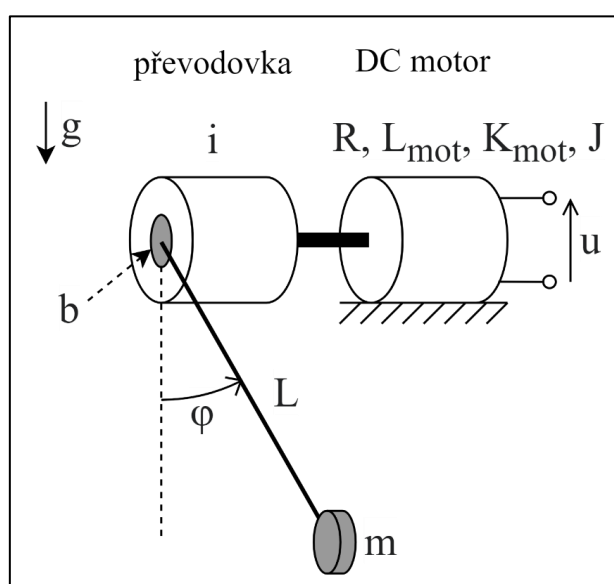
Na rozdíl od modelu jednoduchého DC motoru je v tomto případě využito pro propojení elektrické a mechanické části odlišného bloku, konkrétně „DC Motor“, který má větší možnost nastavení a parametrizace, než dříve použitý blok obecného rotačního elektromechanického převodníku.



Obr. 12. Schematický diagram modelu DC motor + setrvačník na pružině.

## DC motor + závaží na rameni

Model DC motoru se závažím na rameni je alternativou k předchozímu modelu, kdy je zátěž tvořena závažím na rameni (Obr. 13). Elektrická část je stejná jako u přechodícího modelu. Velká část mechanických komponent je vytvořena s pomocí knihovny Simscape Multibody.



Obr. 13. Schematický diagram modelu DC motor + závaží na rameni.



## 4.1 VOLBA VSTUPŮ DO MODELŮ

Vzhledem k tomu, že v rámci optimalizační úlohy probíhá veškeré nastavení a ovládání modelu z MATLABu, je vhodné mít funkce, pomocí kterých dokážeme vygenerovat vstupní signály. Bohužel v současnosti neexistují hotové funkce, které by generovaly signály přesně dle našich potřeb, byla proto vytvořena vlastní sada těchto funkcí.

Níže je uveden seznam vytvořených funkcí. Výstup všech funkcí je proměnná, kde první sloupec matice jsou údaje o čase a druhý sloupec jsou hodnoty signálu v daném čase. Společnými parametry všech funkcí je celkové trvání signálu (v sekundách) a vzorkovací perioda signálu (v sekundách), přičemž pokud je u diskretních signálů nastavena na hodnotu 0, jsou generovány pouze klíčové body signálu (začátek, konec, změna hodnoty). Dále má každá funkce své další specifické vstupní parametry, ovlivňující výsledný tvar signálu.

- Nulový signál
- Konstantní nenulový signál
- Skokový signál (step)
- Pulz
- Náhodné schody
- Sinusovka
- Chirp signál (čerp)<sup>3</sup>

V případě modelu s více vstupy, je pak vhodné sjednotit tyto signály tak, aby měly společnou vzorkovací periodu. Pro tento účel byla vytvořena samostatná funkce.

### **Vygenerování signálů pro konkrétní modely**

Pro každý z modelů, byla za pomoci funkcí uvedených výše vytvořena sada vstupů skládající se vždy z těchto signálů:

1. Skokový signál
2. Sinusovka
3. Pulz
4. Náhodné schody
5. Chirp

Vstupy byly vždy vytvořeny tak, aby adekvátně vybudily model – dostatečně silné, ale ne tak silné, aby se model hned dostal do limitního či extrémního stavu.

---

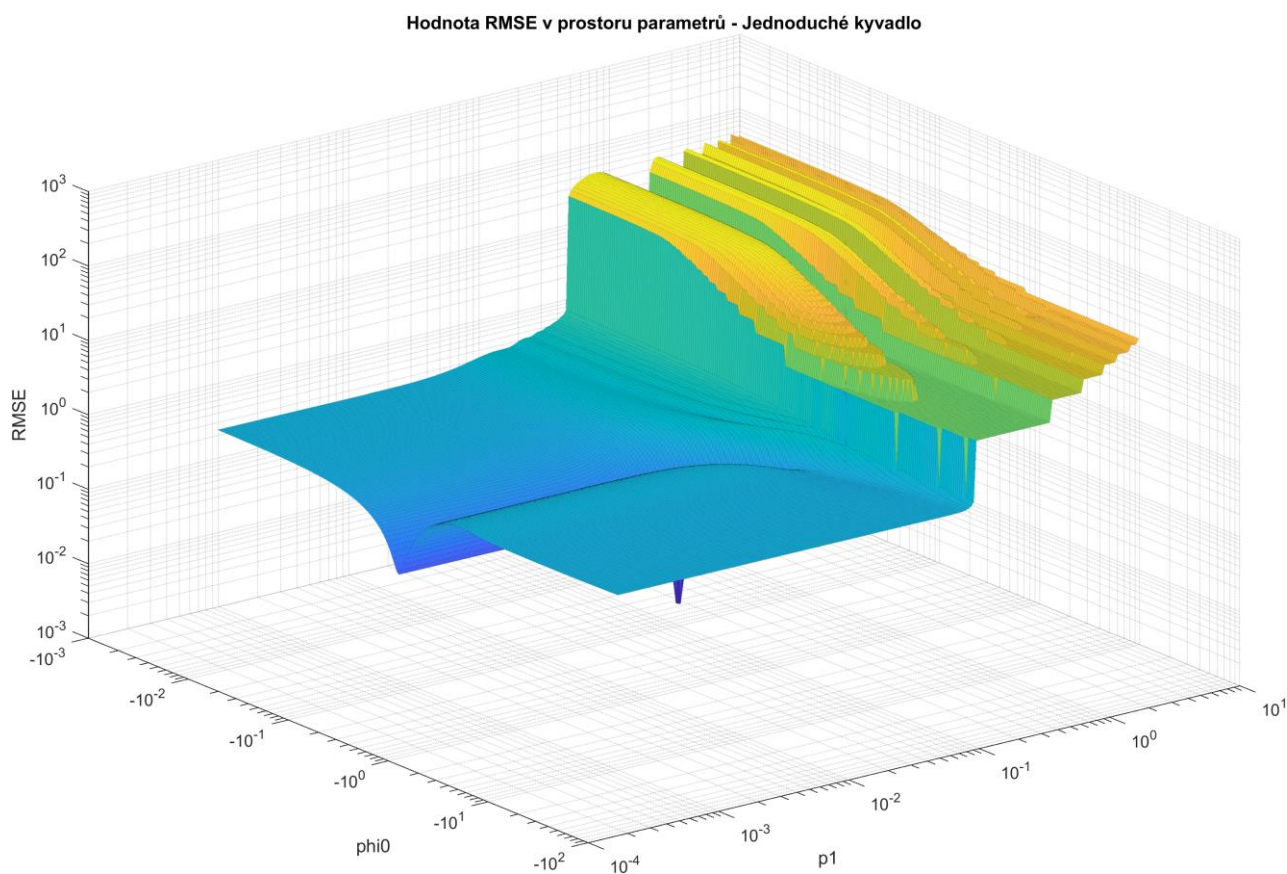
<sup>3</sup> Čerp (z anglického chirp) je sinusový signál jehož frekvence roste lineárně s časem.

## 4.2 VIZUALIZACE PROSTORU PARAMETRŮ PRO VYBRANÉ PŘÍPADY

Díky tomu, že v tuto chvíli máme k dispozici modely, jejich parametry i vstupní signály, můžeme provést vizualizaci hodnotící funkce optimalizační úlohy v prostoru parametrů. Připomeňme, že hodnotící funkcí je v našem případě kvantifikace rozdílu mezi odezvou reálného (původního) systému a odezvou modelu s hledanými parametry, konkrétně pomocí výpočtu RMSE (Root Mean Square Error).

Aby byla pro člověka snadno čitelná, můžeme tuto vizualizaci efektivně zobrazit v maximálně 3D grafu, což znamená závislost RMSE na dvou parametrech.

Jak je možné vidět na příkladu jednoho z těchto grafů (Obr. 14), i pro relativně nízké dimenze prohledávaného prostoru se v systémech často vyskytuje mnoho lokálních minim, která někdy bývají i relativně „hluboká“, což může představovat problém zejména pro lokální řešiče. V jiných oblastech se pak hodnota RMSE mění jen minimálně, což může také v některých případech působit optimalizačnímu algoritmu problémy (např. volba příliš velkých kroků gradientních metod).



Obr. 14. Závislost RMSE na vychýlených parametrech (Jednoduché kyvadlo).

## 5 OPTIMALIZAČNÍ ALGORITMY A NOVÁ METODA

### 5.1 NÁVRH SROVNÁVACÍHO TESTU

Před návrhem nového optimalizačního algoritmu je třeba prozkoumat chování a úspěšnost již dostupných algoritmů. Vzhledem k tomu, že modely i pomocné funkce byly dosud vytvářeny v programu MATLAB, zaměříme se pouze metody, které jsou v tomto nástroji dostupné (viz 2.2).

Výsledkem byl návrh srovnávacího testu, který je bodově popsán níže:

- 10 modelů systémů (viz 4).
- 11 optimalizačních algoritmů (viz 2.2).
- 2 varianty nastavení optimalizačního algoritmu.
- 7 variant kombinací hledaných parametrů.
- 2 druhy vstupních signálů do modelu.
- 4 varianty vychýlení počátečního odhadu hledaných parametrů od skutečné hodnoty.
- 2 varianty nastavení hranic prohledávaného prostoru parametrů.

Kombinací všech těchto variant dostáváme celkový počet 24640 optimalizačních úloh, jejichž výsledky budeme následně vyhodnocovat.

Pro účely praktické realizace srovnávacího testu bylo potřeba vytvořit několik doplňkových funkcí a jednu novou třídu objektů:

#### **Hlavní třída pro simulaci modelů**

Tato univerzální třída slouží pro nastavení modelu v Simulinku, opakované spouštění simulací se změněnými parametry během optimalizace, výpočet hodnotící funkce RMSE, ukládání výstupních dat a monitorování celkového průběhu optimalizace (uplynulý čas, počet simulací, počet chyb).

#### **Výstupní funkce optimalizačních metod**

Tyto funkce slouží k průběžnému ukládání mezivýsledků optimalizace. Testované algoritmy totiž ukládání mezivýsledků samy o sobě nepodporují.

#### **Nastavení limitů parametrů pro experimenty**

Tato funkce slouží zejména pro snadnější a automatické nastavení srovnávacího testu. Na základě dodaných počátečních odhadů parametrů jsou vygenerovány horní a dolní limity těchto parametrů s využitím zvolené metody pro náhodné generování a řádu odchylky.

## 5.2 VÝSLEDKY SROVNÁVACÍHO TESTU

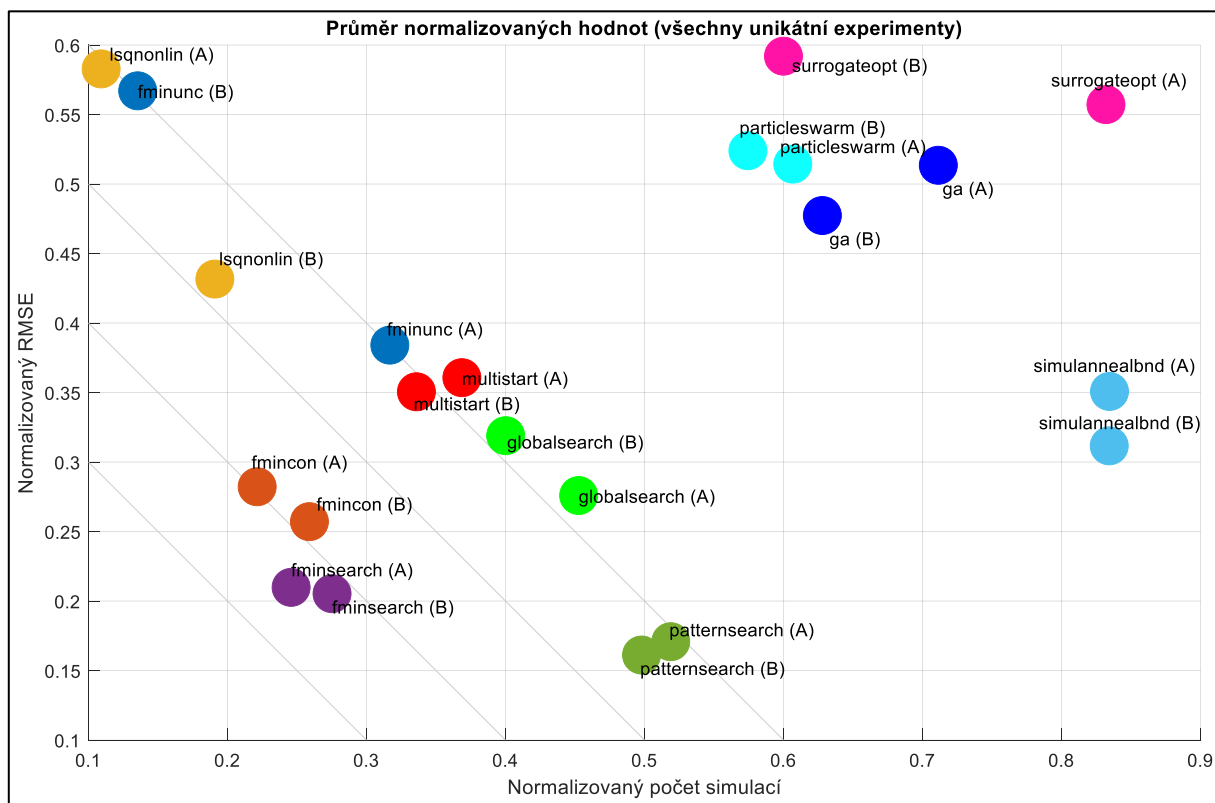
Hlavní sledované metriky pro vyhodnocení srovnávacího testu byly:

- Finální hodnota hodnotící funkce (RMSE).
- Počet simulací modelu (výpočet hodnotící funkce).

Aby bylo možné vzájemně srovnat jednotlivé algoritmy, není možné přímo numericky srovnávat počet simulací a finální RMSE napříč všemi experimenty. Z tohoto důvodu byla provedena normalizace obou sledovaných metrik (škálování hodnot do intervalu od 0 do 1) pro každý unikátní experiment.

Ve finále získáme 1120 unikátních experimentů, pro každý optimalizační algoritmus a jeho varianty. V rámci každého jednoho unikátního experimentu, jsou normalizovány dvě hlavní sledované metriky – RMSE a počet simulací.

Z těchto výsledků byl vytvořen výsledný graf, který je na Obr. 15. Tento graf byl vytvořen zprůměrováním normalizovaných hodnot obou sledovaných metrik napříč všemi experimenty.



Obr. 15. Srovnání normalizovaných výsledků optimalizačních algoritmů (průměr).

Z výsledků plyne, že nejúspěšnější optimalizační algoritmy pro námi definované nastavení experimentů jsou:

- fmincon – velmi rychlý, méně přesný.
- fminsearch – relativně rychlý i přesný.
- patternsearch – pomalejší, ale velmi přesný.

Lokální optimalizační algoritmy si vedly lépe než globální. To může být zapříčiněno tím, že globální metody potřebují větší počet simulací, než se projeví jejich výhoda, oproti lokálním algoritmům. Nejlepší globální algoritmy jsou:

- GlobalSearch – spíše přesnější.
- MultiStart – spíše rychlejší.

Další globální optimalizační algoritmy (particleswarm, surrogateopt, simulanneal, ga) dosáhly viditelně horších výsledků.

### **5.3 POŽADAVKY PRO TVORBU NOVÉHO ALGORITMU**

Důležitými požadavky na nový optimalizační algoritmus je mimo jiné jeho robustnost a automatické prohledávání prostoru parametrů. Typický uživatel tohoto algoritmu nemusí mít velké zkušenosti s optimalizačními úlohami, ale má k dispozici výpočetní výkon. Například mu nevádí, že optimalizace bude běžet několik hodin (třeba přes noc), hlavně když najde globální optimum, a to ideálně bez nutnosti zásahu uživatele během celého procesu.

Zkoumané prostory parametrů obsahují mnoho lokálních minim a nemusí být spojitě což znevýhodňuje algoritmy využívající gradient hodnotící funkce. Naopak výhodnými se jeví „direct search“ metody, což potvrzují i dosavadní výsledky testů.

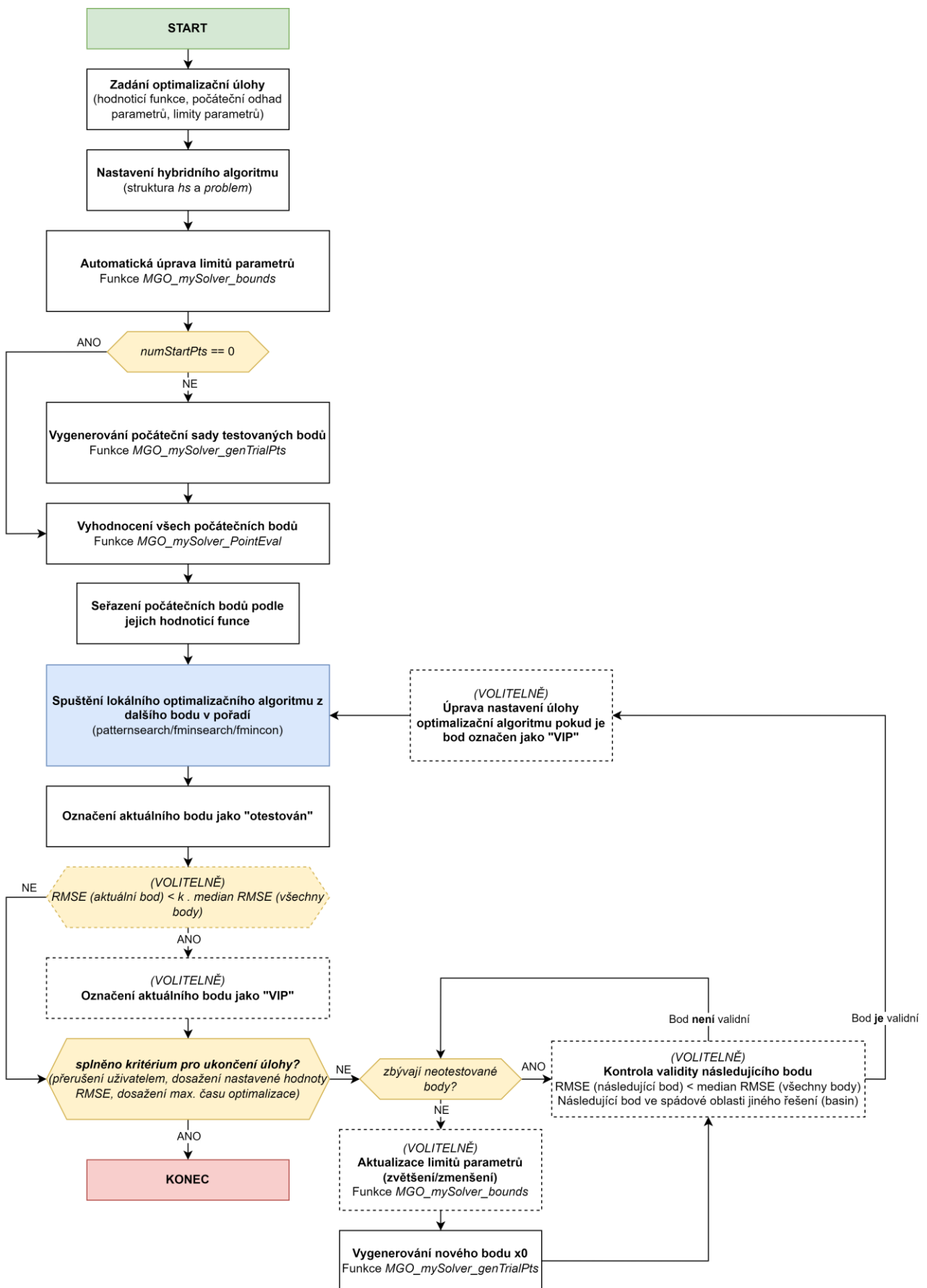
Globální optimalizační algoritmy jako GlobalSearch a MultiStart jsou výhodné z hlediska důkladného prohledávání prostoru parametrů (mnoho náhodných počátečních bodů), nicméně trpí různými nedostatky jako je omezená možnost volby lokálních optimalizačních metod či závislost na nastavení počtu náhodně generovaných počátečních bodů, což může vést k předčasnému ukončení optimalizace nebo naopak příliš mnoha iteracím. Předčasné (korektní) ukončení těchto algoritmů zásahem uživatele je také poměrně komplikované.

Nový hybridní algoritmus by se tedy měl inspirovat metodami globálního prohledávání a zaměřit se na „direct search“ metody. Dále by měl implementovat mechanismy pro automatické prohledávání bez zásahu uživatele a případně implementovat dodatečné heuristiky pro zrychlení konvergence optimalizačního algoritmu.

### **5.4 DESIGN NOVÉHO ALGORITMU**

V této kapitole představíme námi navrženou strukturu nového hybridního optimalizačního algoritmu. Označení „hybridní“ volíme z důvodu, že kombinuje metody lokálních optimalizačních metod s globálním prohledáváním, a také pro jednoznačné odlišení od názvu jiných globálních algoritmů, zejména GlobalSearch.

Přehledový vývojový diagram hybridního algoritmu je vidět na Obr. 16. Jeho jednotlivé kroky a funkce pak postupně popíšeme níže.



Obr. 16. Vývojový diagram navrženého hybridního algoritmu

## **Zadání optimalizační úlohy**

Jako první je nutné, podobně jako u ostatních optimalizačních algoritmů, zadat tři hlavní vstupy – reference na hodnoticí funkci, počáteční odhad hledaných parametrů a limity hledaných parametrů (pokud jsou zadány)

## **Nastavení hybridního algoritmu**

Stejně jako u algoritmů GlobalSearch a MultiStart je nastavení všech parametrů optimalizační úlohy provedeno pomocí dvou proměnných. Struktura `hs` obsahuje nastavení a data nadřazené (globální) části algoritmu, `problem` pak obsahuje nastavení lokálního optimalizačního algoritmu.

## **Automatická úprava limitů parametrů**

Tato funkce má dva hlavní účely:

- Kontrola nastavení limitů parametrů `lb`, `ub`
- Rozšíření/zmenšení limitů parametrů `lb`, `ub` (volitelně)

## **Vygenerování počáteční sady testovaných bodů**

Zde dojde k vygenerování počáteční testovací sady náhodných bodů, přičemž je možné zvolit jejich počet, typ rozložení náhodných bodů (rovnoměrné, normální, mřížka) a případně jeho směrodatnou odchylku.

## **Vyhodnocení všech počátečních bodů**

Tato funkce vypočítá hodnotu hodnoticí funkce pro všechny body počátečních odhadů parametrů vygenerované v předchozím kroku.

## **Seřazení počátečních bodů podle jejich hodnoticí funkce**

Vygenerované body jsou následně seřazeny vzestupně dle výsledků hodnoticí funkce v předchozím kroku. Optimalizační algoritmus pak bude díky tomu testovat nejdříve body s nejmenší hodnoticí funkcí v místě počátečního odhadu.

## **Spuštění lokálního optimalizačního algoritmu z dalšího bodu v pořadí**

Hybridní algoritmus v tomto kroku vybere ze seznamu první (další) bod v pořadí a spustí lokální optimalizaci. Jak již bylo zmíněno, v rámci této práce používáme pouze metody `fmincon`, `fminsearch` nebo `patternsearch`.

## **Označení aktuálního bodu jako "otestován"**

Po ukončení běhu lokálního optimalizačního algoritmu je pro daný bod zapsána hodnota nalezených parametrů a výsledné hodnoticí funkce.

## **Označení aktuálního bodu jako "VIP" (volitelné)**

Pokud se uživatel rozhodne tento krok provést, je po ukončení lokální optimalizační úlohy proveden test, zda je dosažená hodnoticí funkce v daném bodě menší než medián výsledků hodnoticí funkce řešení pro všechny předchozí otestované body, vynásobený konstantou  $k$ . Na základě experimentů jsme zvolili hodnotu konstanty  $k = 0,7$ . Pokud aktuální bod tento výsledek splní, je jeho status změněn na „VIP“.

### **Test splnění kritéria pro ukončení úlohy**

Dále je provedena kontrola, zda nebylo splněno některé z kritérií pro ukončení hybridního optimalizačního algoritmu:

- Manuální přerušení výpočtu uživatelem.
- Dosažení požadované hodnoty hodnoticí funkce
- Dosažení maximálního povoleného času výpočtu

Pokud je některé z těchto kritérií splněno, algoritmus se ukončí a vrátí jako řešení nejlepší nalezený bod ze seznamu. V opačném případě algoritmus pokračuje v prohledávání prostoru parametrů.

### **Kontrola zbývajících neotestovaných bodů**

Pokud v seznamu zbývají body, ze kterých ještě nebyl spuštěn lokální optimalizační algoritmus, je vybrán další takovýto bod v pořadí a následuje volitelný krok *Kontrola validity následujícího bodu*. Pokud se již žádný takový bod v tabulce nenachází, pokračuje algoritmus krokem *Aktualizace limitů parametrů* (volitelný) nebo rovnou *Vygenerování nového bodu*.

### **Aktualizace limitů parametrů (volitelné)**

V případě, že si uživatel není jistý hodnotami zvolených limitů parametrů a chce tyto limity během běhu optimalizace postupně zvětšovat nebo naopak zmenšovat, je možné provést v tomto kroku jejich aktualizaci (vynásobení zvoleným koeficientem).

### **Vygenerování nového bodu**

V případě že již nejsou k dispozici další body počátečních odhadů parametrů pro lokální optimalizační algoritmus je vygenerován další bod. Pro nově vygenerovaný bod je pak vypočítána hodnoticí funkce a je přidán na konec seznamu všech bodů.

### **Kontrola validity následujícího bodu (volitelné)**

V případě potřeby je možné v tomto kroku provést kontrolu validity bodu, který je další v pořadí pro testování. Je tak možné vyfiltrovat body, které například nemají výrazně lepší počáteční hodnotu hodnoticí funkce než ostatní body, případně ignorovat body, které leží ve spádové oblasti jiného řešení. Pokud je následující bod validní, pokračuje algoritmus dalším krokem v pořadí. V opačném případě se vrátí ke kroku *Kontrola zbývajících neotestovaných bodů*.

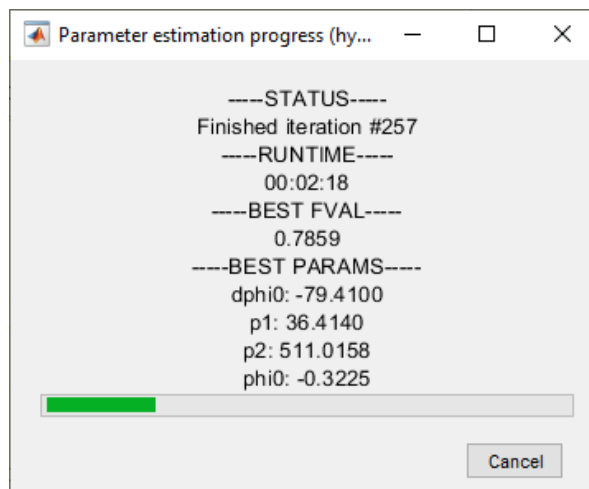
### **Úprava nastavení úlohy optimalizačního algoritmu, pokud je bod označen jako "VIP" (volitelné)**

Pokud je další testovaný bod v pořadí označen jako „VIP“, je před spuštěním lokálního optimalizačního algoritmu provedena dočasná úprava jeho nastavení tak, aby bylo možné tento bod důkladněji prozkoumat.

Poté již následuje krok *Spuštění lokálního optimalizačního algoritmu z dalšího bodu v pořadí* a celý proces se opakuje, dokud není splněna některá z podmínek pro ukončení hybridního optimalizačního algoritmu.



V průběhu optimalizace se také uživateli zobrazí jednoduché okno aktuálního stavu (Obr. 17). Pomocí tohoto dialogového okna je navíc možné jednoduše manuálně ukončit optimalizaci bez toho, aby došlo ke ztrátě dat.



Obr. 17. Dialogové okno hybridního algoritmu s aktuálním stavem optimalizace.

## 5.5 TESTOVÁNÍ NOVÉHO HYBRIDNÍHO ALGORITMU A SROVNÁNÍ S JINÝMI METODAMI

Poté co byl dokončen návrh nového hybridního algoritmu, byl proveden další velký srovnávací test algoritmů, podobně jako v kapitole 5.1.

Níže uvedené položky srovnávacího testu byly ponechány beze změny:

- 10 modelů systémů
- 7 variant kombinací hledaných parametrů
- 4 varianty vychýlení počátečního odhadu hledaných parametrů od skutečné hodnoty
- 2 varianty nastavení hranic prohledávaného prostoru parametrů

Následující položky byly upraveny:

- 6 optimalizačních algoritmů
  - Hybridní algoritmus (lokální algoritmus patternsearch)
  - Hybridní algoritmus (lokální algoritmus fminsearch)
  - Hybridní algoritmus (lokální algoritmus fmincon)
  - GlobalSearch
  - MultiStart
  - Genetický algoritmus
- 1 varianta nastavení optimalizačního algoritmu
- 1 druh vstupního signálu do modelu (skokový signál)

Kombinací všech těchto variant dostáváme celkový počet 3360 optimalizačních úloh, jejichž výsledky budeme následně vyhodnocovat.

Hlavní rozdíl oproti předchozímu srovnávacímu testu (mimo výše uvedených bodů), spočíval v tom, že pro každý optimalizační algoritmus byl výrazně zvýšen časový limit pro výpočet. Vyšší limit času pro výpočet dává algoritmům větší prostor pro důkladné prozkoumání prostoru parametrů a mělo by omezit případy, kdy se díky tomuto omezení nemohly projevit všechny silné stránky daného optimalizačního algoritmu. Časový limit pro jednu optimalizační úlohu se v závislosti na počtu parametrů pohyboval v rozmezí 0,5 – 2 hodiny.

Než přejdeme k vyhodnocení výsledků tohoto testu, je nutné zmínit, že u hybridního algoritmu nebyly v tomto případě aplikovány všechny z volitelné kroky, které jsou popsány v přechozí kapitole 5.4 a na přehledovém diagramu algoritmu (Obr. 16). Použit byl pouze volitelný krok výběru VIP bodů pro důkladnější otestování.

Důvod pro toto rozhodnutí je, že v testu chceme primárně zkoumat výhodnost využití metod přímého prohledávání pro lokální optimalizaci. Pokud se ukáže, že hybridní algoritmus dosahuje dobrých výsledků i bez použití doplňkových funkcí filtrace bodů či úpravy limitů parametrů, bude s těmito funkcemi pravděpodobně fungovat ještě lépe. Druhým důvodem je, že v době testování ještě nebyla funkcionálna těchto volitelných kroků zcela odladěna a mohla by paradoxně způsobit zhoršení výsledků hybridního algoritmu.

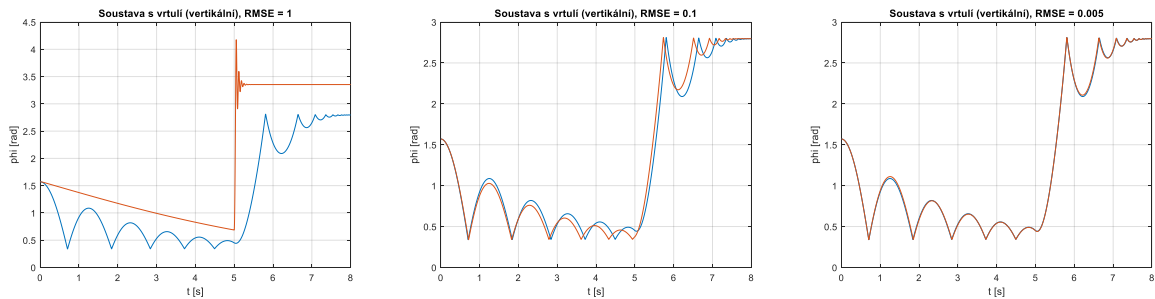
## **5.6 VÝSLEDKY TESTŮ NOVÉHO HYBRIDNÍHO ALGORITMU A JEJICH ZHODNOCENÍ**

Na rozdíl od přechozího srovnávacího testu, jsme se v tomto případě zaměřili na dvě částečně odlišné metriky pro jeho vyhodnocení:

- Kolikrát (pro daný model) našel algoritmus globální minimum.
- V případě že algoritmus globální minimum našel, kolik vyhodnocení hodnotící funkce (spuštění simulace) k tomu bylo zapotřebí.

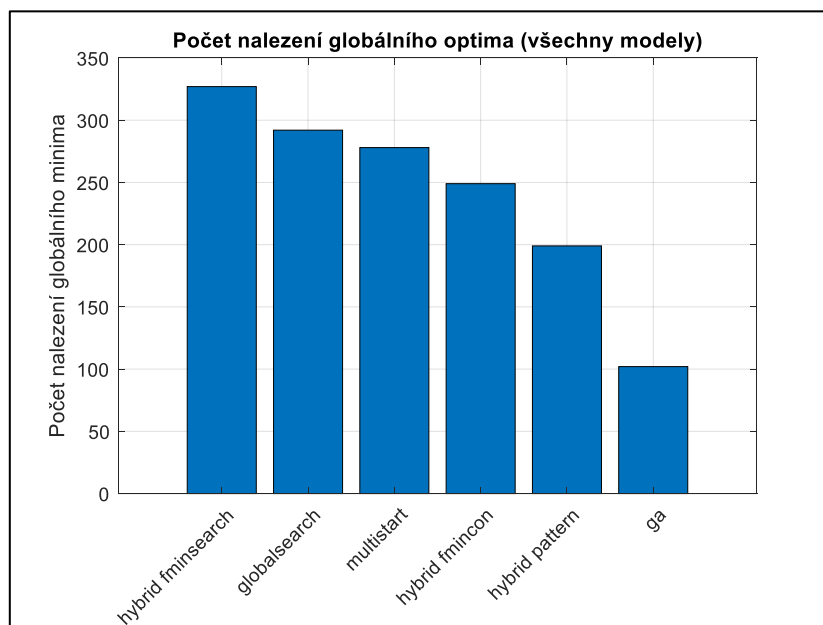
Ve chvíli, kdy máme takto zvolené sledované metriky, musíme samozřejmě určit limit pro to, co považujeme za globální optimum, na základě finální hodnotící funkce (RMSE). Volba tohoto limitu závisí nakonec vždy na uživateli, který zpravidla určuje kvalitu nalezených parametrů na základě vizuální inspekce grafů odezvy původního systému a modelu s odhadnutými parametry. Tento postup jsme zvolili i v případě této práce.

Pro každý model jsme tedy provedli vizuální inspekci grafů jejich odezvy. V grafech byla vždy zobrazena odezva modelu s původními (reálnými) parametry a dále odezva systému s odhadnutými parametry. Toto srovnání jsme u každého modelu provedli pro několik hodnot výsledného RMSE a rozhodli jsme, jaká je limitní hodnota RMSE, kdy se již oba průběhy signálu dostatečně shodují. Příklad tohoto postupu je na Obr. 18.



Obr. 18. Srovnání původního signálu (modře) s odezvou modelu s nalezenými parametry (červeně) pro různé hodnoty RMSE (model Soustava s vrtulí (vertikální)). Vlevo – špatná shoda, Střed – dobrá shoda, Vpravo – téměř dokonalá shoda

Výsledný Počet nalezení globálního optima napříč všemi modely a experimenty je níže na Obr. 19. V grafu můžeme pozorovat, že nejlepších výsledků dosáhl náš nově navržený hybridní optimalizační algoritmus ve variantě s lokálním algoritmem fminsearch.



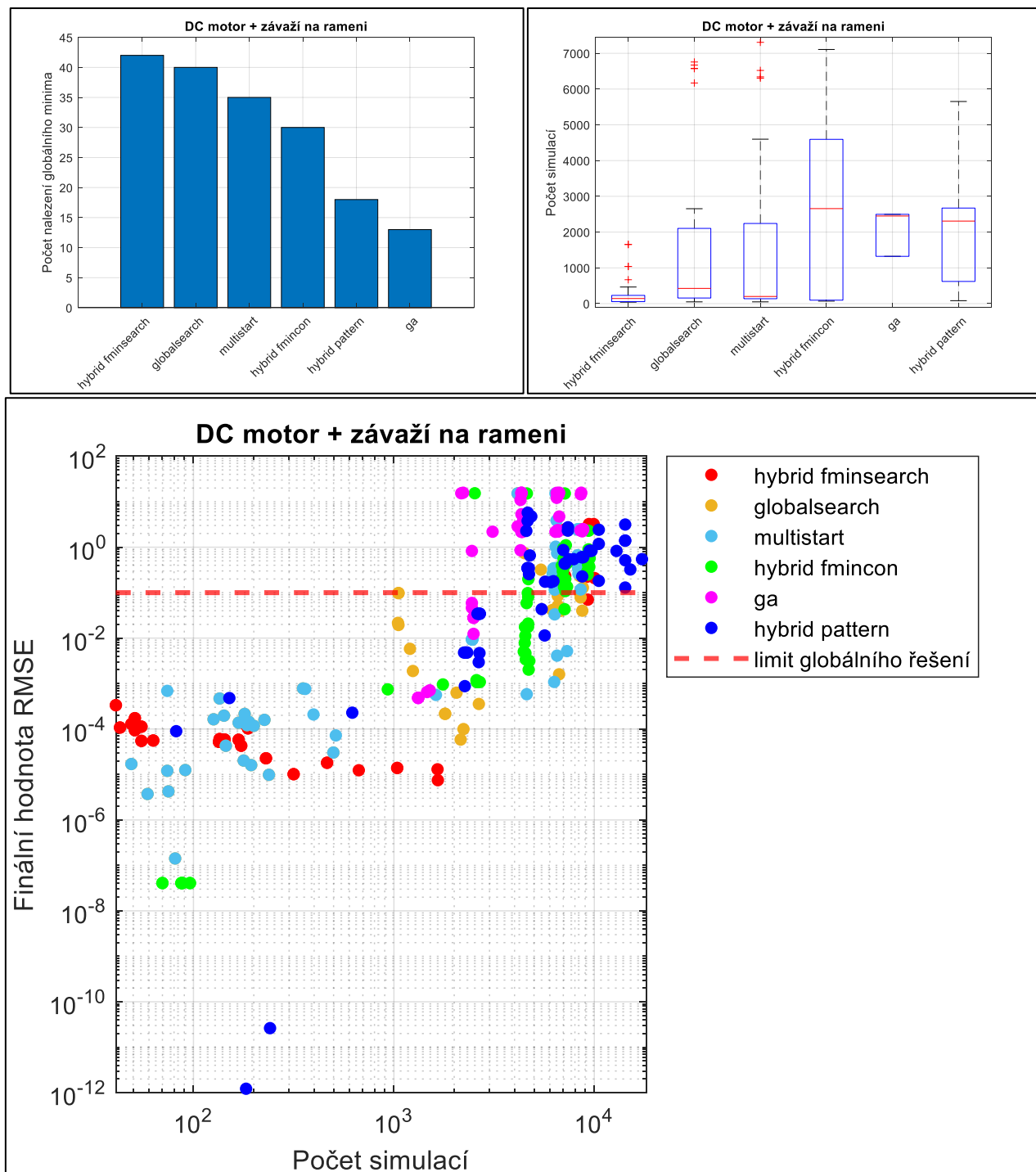
Obr. 19. Počet nalezení globálního optima napříč všemi experimenty.

Jako příklad na tomto místě nyní uvedeme grafy výsledků pro jeden konkrétní model. Tyto výsledky jsou níže na Obr. 20 a skládají se ze tří grafů:

- Vlevo nahoře: graf počtu nalezení globálního optima jednotlivých algoritmů.
- Vpravo nahoře: statistické vyhodnocení<sup>4</sup> počtu potřebných simulací k dosažení globálního optima. Toto vyhodnocení je provedeno pomocí funkce `boxplot`.
- Dole: vykreslení všech výsledků optimalizačních úloh pro daný model, bez ohledu na nalezení globálního optima.

<sup>4</sup> Středová značka každého rámečku označuje medián a spodní a horní okraj rámečku 25., resp. 75. percentil. Metličky sahají k nejextrémnějším datovým bodům, které nejsou považovány za odlehlé hodnoty, a odlehlé hodnoty jsou vykresleny jednotlivě pomocí symbolu "+".

Z uvedených grafů pro tento konkrétní model můžeme vyčíst, že nový hybridní algoritmus ve variantě fminsearch má nejen nejlepší úspěšnost v počtu dosažených globálně optimálních řešení, ale je i v průměru výrazně rychlejší.



Obr. 20. Grafy výsledků srovnávacího testu globálních optimalizačních algoritmů (DC motor + závaží na rameni)

Pokud bychom měli pro jednotlivé modely stručně shrnout dosažené výsledky hybridního algoritmu v porovnání mezi sebou a s ostatními testovanými globálními metodami, můžeme tyto výsledky heslovitě vyjádřit pomocí tabulky níže (Tabulka 1).

<b>Model</b>	<b>Hybridní algoritmus (fminsearch)</b>	<b>Hybridní algoritmus (fmincon)</b>	<b>Hybridní algoritmus (patternsearch)</b>
Jednoduché kyvadlo	Nejúspěšnější a výrazně nejrychlejší	Velmi úspěšný a rychlý	Málo úspěšný a pomalý
DC motor	Středně úspěšný, středně rychlý	Středně úspěšný ale nejrychlejší	Málo úspěšný a pomalý
Dvojitě kyvadlo	Nejúspěšnější a nejrychlejší	Průměrně úspěšný a rychlý	Velmi úspěšný, středně rychlý
Rotační kyvadlo	Nejúspěšnější ale pomalejší	Málo úspěšný ale rychlý	Velmi málo úspěšný a pomalý
Soustava s vrtulí (horizontální)	Nejúspěšnější, středně rychlý	Málo úspěšný, ale rychlý	Velmi málo úspěšný a pomalý
Soustava s vrtulí (vertikální)	Středně úspěšný, středně rychlý	Velmi málo úspěšný, středně rychlý	Nejúspěšnější a středně rychlý
Těleso na naklápěcí plošině	Nejúspěšnější a výrazně nejrychlejší	Neúspěšný, a pomalý	Středně úspěšný a středně rychlý
Kotoučová brzda	Nejúspěšnější a nejrychlejší	Nejúspěšnější a rychlý	Nejúspěšnější ale pomalý
DC motor + setrvačnick na pružině	Nejúspěšnější a nejrychlejší	Středně úspěšný a rychlý	Středně úspěšný a velmi pomalý
DC motor + závaží na rameni	Nejúspěšnější a výrazně nejrychlejší	Středně úspěšný a pomalý	Málo úspěšný a pomalý

*Tabulka 1. Slovní shrnutí úspěšnosti nalezení globálního optima a rychlosti konvergence hybridního algoritmu pro jednotlivé modely.*

Po uvážení všech výše uvedených informací, lze konstatovat, že nově navržený algoritmus Hybrid-fminsearch, pro námi zkoumanou úlohu odhadu parametrů mechatronických systémů, přináší nezanedbatelné zlepšení oproti jiným optimalizačním algoritmům, které jsou dostupné v programu MATLAB. Toto zlepšení se projevuje v průměru větší pravděpodobností nalezení globálního optima za kratší čas.

## 6 POMOCNÉ ALGORITMY PRO ODHAD PARAMETRŮ

### 6.1 DETEKCE ZÁVISLÝCH PARAMETRŮ METODOU MONTE CARLO

V případě, že jsou hledané parametry vzájemně závislé, můžeme dojít k řešení, které je z pohledu optimalizačního algoritmu kvalitní, ovšem parametry neodpovídají reálnému systému. Existuje totiž nekonečně mnoho optimálních řešení této úlohy.

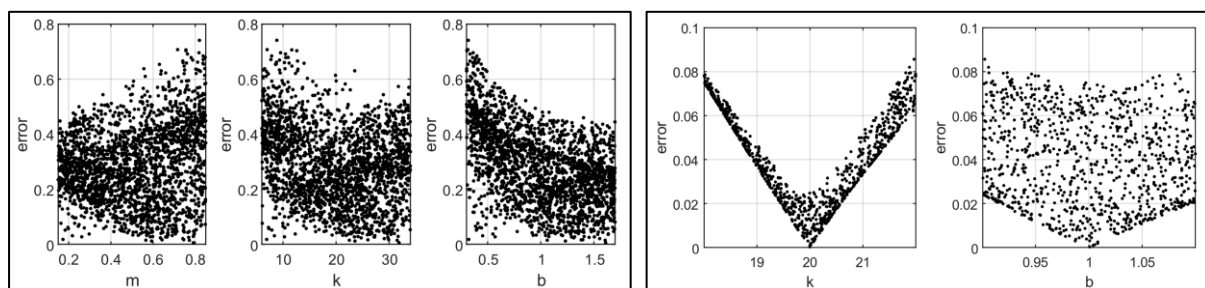
První algoritmus slouží k otestování vzájemné závislosti parametrů metodou Monte Carlo a byl popsán ve vlastním samostatném článku [9].

Vypočítané (resp. odsimulované) závislosti hodnotící funkce na jednotlivých náhodně vychýlených parametrech vykreslíme do bodového grafu. Pokud jsou parametry vzájemně závislé, bude existovat více řešení se stejnou hodnotou hodnotící funkce jako u původně nalezeného řešení. Viz rovnice (6.1) pro jednoduchý mechanický oscilátor se třemi závislými parametry  $k, m, b$ , a graf na Obr. 21 vlevo.

$$\ddot{x} = -\frac{k}{m}x - \frac{b}{m}\dot{x} \quad (6.1)$$

Naopak pokud jsou parametry nezávislé, měl by být v grafech jasně patrný tvar „V“, protože při vychýlení nezávislého parametru z optimální hodnoty se hodnotící funkce vždy zvětší. To můžeme vidět po úpravě rovnice tak, aby obsahovala pouze nezávislé parametry  $p_1, p_2$  (6.2), na grafu s charakteristickým „V“ tvarem (Obr. 21 vpravo).

$$\ddot{x} = -p_1x - p_2\dot{x} \quad (6.2)$$



Obr. 21. Grafy pro mechanický oscilátor se třemi závislými parametry (vlevo) a dvěma nezávislými parametry (vpravo). [9]

### 6.2 RYCHLÁ DETEKCE ZÁVISLÝCH PARAMETRŮ

Nevýhodami předchozího algoritmu je, že je nutné jej provést až po ukončení optimalizační úlohy a jeho relativně vysoká výpočetní náročnost. Proto byla navržena další, v zásadě velmi triviální, zato však rychlá a efektivní metoda, která dokáže detekovat závislé parametry ještě před začátkem optimalizační úlohy.

Pokud provedeme se dvěma zkoumanými parametry  $a$  a  $b$  některou z ekvivalentních operací (Tabulka 2) a zjistíme, že se výstupní signál systému  $y$  vůbec nezměnil, je důvodné podezření že tyto dva parametry jsou závislé a neseparovatelné.

Základní rovnice	Ekvivalentní operace 1		Ekvivalentní operace 2	
$y = (ab)x$	$y = (ba)x$	$a = b$ $b = a$	$y = \left(ak \ b \frac{1}{k}\right)x$	$a = ak$ $b = b \frac{1}{k}$
$y = \left(\frac{a}{b}\right)x$	$y = \left(\frac{1}{\frac{b}{1}}\right)x$	$a = \frac{1}{b}$ $b = \frac{1}{a}$	$y = \left(\frac{ak}{bk}\right)x$	$a = ak$ $b = bk$
$y = (a + b)x$	$y = (b + a)x$	$a = b$ $b = a$	$y = ((a + k) + (b - k))x$	$a = a + k$ $b = b - k$
$y = (a - b)x$	$y = (-b - (-a))x$	$a = -b$ $b = -a$	$y = ((a + k) - (b + k))x$	$a = a + k$ $b = b + k$
$y = (a^b)x$	$y = \left((a^2)^{\frac{b}{2}}\right)x$	$a = a^2$ $b = \frac{b}{2}$	$y = \left((a^k)^{\frac{b}{k}}\right)x$	$a = a^k$ $b = \frac{b}{k}$

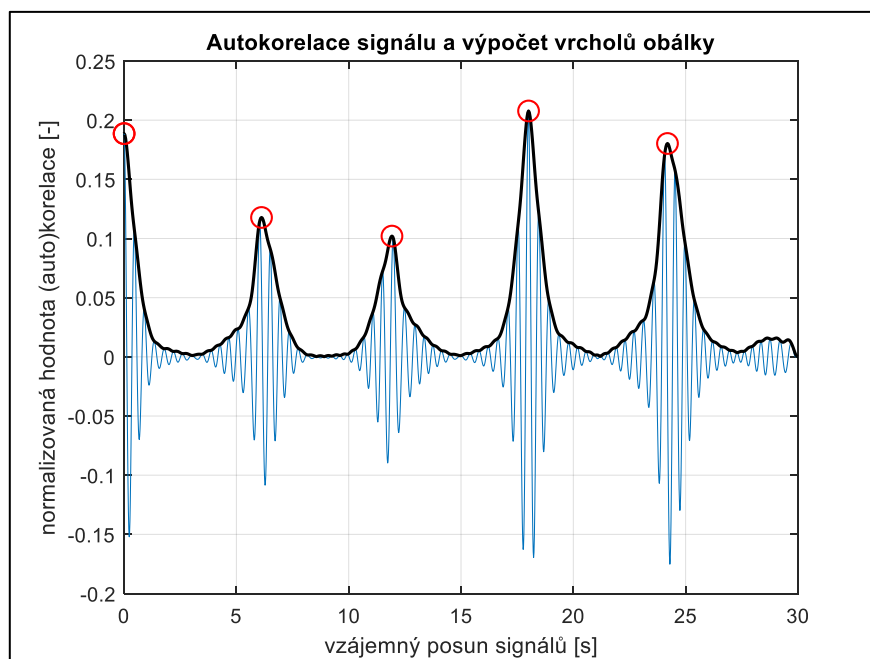
Tabulka 2. Příklady ekvivalentních operací a souvisejících substitucí.

Tento postup je nezávislý na přesnosti počátečních odhadů zkoumaných parametrů. Nevýhodou je, že tento test funguje pouze pokud se parametry vyskytují jen na jednom místě rovnice.

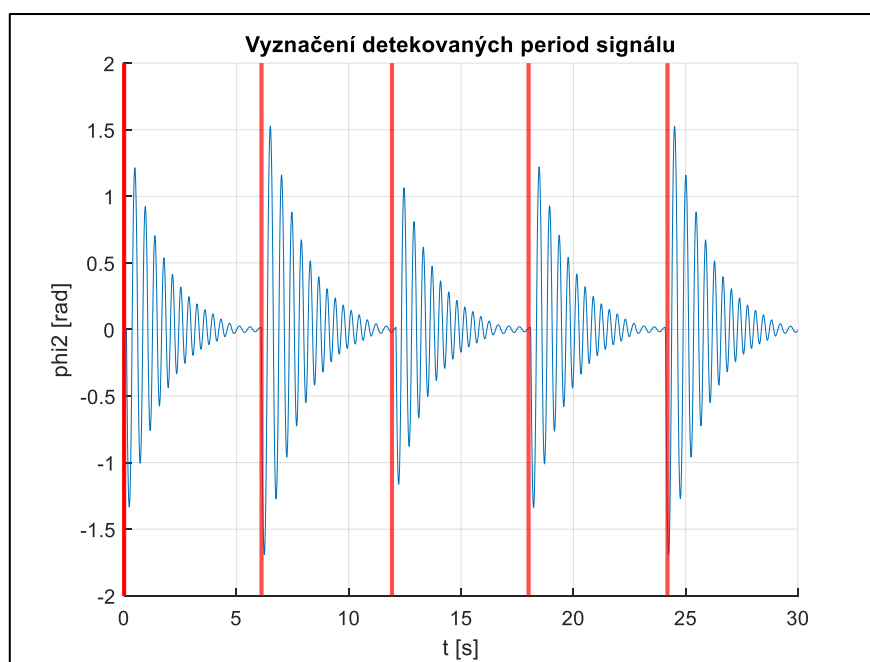
### 6.3 DETEKCE OPAKOVÁNÍ SIGNÁLU

Je zřejmé, že pokud se tvar vstupního, ale zejména výstupního signálu periodicky opakuje bez výrazné odchylky, je možné říci, že z dat následujících po první periodě již pravděpodobně nezískáme žádné nové informace o chování systému. Námí navržený algoritmus pro automatickou detekci opakujícího se signálu se skládá z několika níže uvedených kroků a jeho základní myšlenka je postavena na výpočtu hodnoty autokorelace signálu.

1. Automatické odstranění stejnosměrné složky signálu (volitelně)
2. Výpočet autokorelace signálu
3. Výpočet obálky autokorelace a detekce špiček (Obr. 22)
4. Vykreslení detekovaných opakujících se sekcí signálu (Obr. 23)



Obr. 22. Autokorelace signálu (modrá), obálka pomocí Hilbertovy transformace (černá), detekce vrcholů obálky (červená).



Obr. 23. Vyznačení vzájemně podobných sekcí v původním signálu.

Algoritmus detekce není nikdy zcela dokonalý a je v konečném důsledku na samotném uživateli, aby předložené výsledky zhodnotil. Uvedené kroky algoritmu byly také implementovány do podoby samostatné funkce.

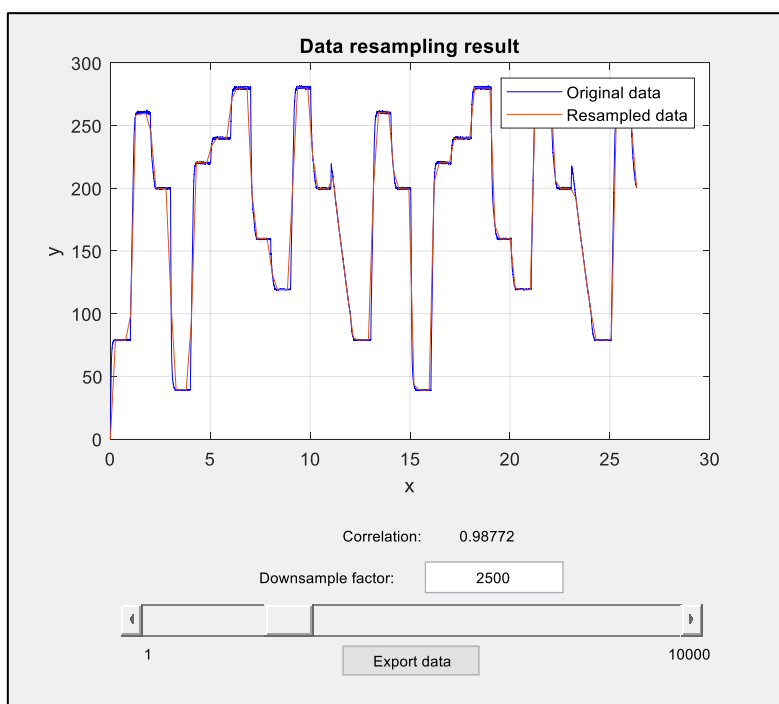


## 6.4 PŘEVZORKOVÁNÍ DAT

Dalším potenciálním problémem v oblasti podoby naměřených dat na reálném modelu, je jejich vzorkovací frekvence. Stává se, že naměřená data jsou vzorkována zbytečně vysokou frekvencí, což může způsobit zpomalení výpočtu vzhledem k objemu dat nebo nevhodnou volbě kroku simulace na základě měřených dat.

Z tohoto důvodu byl navržen interaktivní nástroj, který uživateli umožňuje měnit koeficient převzorkování, vždy na menší počet vzorků než původní signál (Obr. 24).

Nástroj je spuštěn zavoláním funkce, která nemá sama o sobě žádné výstupy, a jejími vstupy jsou parametry  $x$  a  $y$ , které představují vektory hodnot signálu  $y$  v závislosti  $x$ .



Obr. 24. Vytvořené GUI pro převzorkování dat.

Uživatel může koeficient převzorkování měnit manuálně pomocí posuvníku nebo přímým zadáním hodnoty do pole „Downsample factor“. Vizualně je pak možné porovnat tvar původních a převzorkovaných dat pro aktuálně zvolený koeficient.

Pro kvantifikaci míry podobnosti obou signálů je pak navíc vypočítána i hodnota koeficientu korelace obou signálů s využitím MATLAB funkce `corrcoef`.

Hodnota koeficientu korelace 0,999 pak byla experimentálně určena jako vhodná výchozí hodnota, kdy je ještě zachována většina hlavních složek tvaru původního signálu. Automatický algoritmus pak během několika iterací nalezne takovou hodnotu koeficientu převzorkování, aby byla výsledná hodnota koeficientu korelace právě 0,999 a tento koeficient převzorkování uživateli nabídne jako výchozí hodnotu.

## 7 NOVÝ NÁSTROJ PRO ODHAD PARAMETRŮ V INŽENÝRSKÉ PRAXI

V rámci řešení jednotlivých cílů dizertační práce, adresovaných v kapitolách 4, 5 a 6, vzniklo mnoho funkcí, které jsou samostatně použitelné a na sobě nezávislé.

V budoucnu se předpokládá integrace těchto funkcí do ucelené podoby nového MATLAB toolboxu pro úlohu hledání parametrů. V současné chvíli jsou tyto funkce připraveny pro spuštění přímo z příkazové řádky, případně ze skriptu či jiné funkce.

Funkce vytvořené v této práci mají specifické pojmenování a jsou identifikovatelné dle předpon v jejich názvech:

- MGO\_
  - Funkce týkající se nastavení optimalizační úlohy a nového hybridního algoritmu
  - Funkce pomocných algoritmů (detekce závislých parametrů, převzorkování dat)
- SigGen\_
  - Funkce pro generování vstupních signálů systému
- forSim
  - Hlavní třída pro simulaci modelů

Praktický návod pro řešení úlohy a aplikace jednotlivých výsledků této práce s využitím námi vytvořených funkcí je bodově popsán níže:

1. Nastavení úlohy (model systému, neznámé parametry, naměřená data, nastavení úlohy)
2. Analýza vstupních dat
  - a. Detekce neúčinného opakování signálu (viz kap. 6.3)
  - b. Převzorkování naměřených dat (viz kap. 6.4)
3. Preventivní detekce závislých parametrů (viz kap. 6.2)
4. Spuštění optimalizační úlohy – Hybridní algoritmus (viz kap. 5.4)
5. Detekce korelovaných parametrů po ukončení optimalizace (viz kap. 6.1)

Vzhledem k návaznosti tématu této dizertační práce na dizertační práci Ing. Martina Appela – „Výzkum metod pro významné zrychlení odhadu parametrů simulačních modelů“, bude případná integrace námi vytvořených funkcí do podoby toolboxu provedena v návaznosti na jeho praktické výsledky. Práce Ing. Appela nebyla v době tvorby této dizertace ještě dokončena.

Bez ohledu výše uvedenou situaci lze však i nyní použít vyvinuté funkce a nástroje pro efektivnější řešení úlohy hledání parametrů.

## 8 ZÁVĚR

### Vytvoření souboru technických modelů a jejich analýza

- Byla vytvořena sada mechatronických modelů v prostředí MATLAB Simulink s využitím principů fyzikálního modelování (Toolbox Simscape). Tyto modely a jejich parametry jsou odvozeny od reálných mechanismů a zařízení, která se v inženýrské praxi vyskytují.
- Námi vytvořená sada modelů má několik důležitých vlastností, díky kterým je možné ji využít i mimo téma této disertační práce:
  - Jde o konzistentní sadu modelů, které jsou zjednodušenými ale funkčními obrazy reálných strojů. Díky využití fyzikálního modelování jsou dostatečně přehledné a lze je jednoduše upravovat a rozšiřovat.
  - Všechny parametry modelů jsou známy. Vyskytují se navíc i korelované parametry, čehož bylo využito při řešení třetího cíle této práce.
  - Vhodné budicí signály byly připraveny pro všechny modely tak, aby se projevil všechny dynamické vlastnosti.
  - Byla provedena analýza vlastností modelů v souvislosti s úlohou odhadu parametrů. Ta zahrnuje detailní zmapování závislosti hodnotící funkce na zvolených parametrech. Z provedených analýz jsou patrné charakteristické vlastnosti prostoru parametrů pro tyto modely.
- Soubor lze využít pro výuku i další vědecké práce v oblasti identifikace systémů, řídicích algoritmů apod.

### Kombinace globálních a lokálních optimalizačních metod pro mechatronické modely

- Byl proveden praktický srovnávací test optimalizačních algoritmů v MATLABu zahrnující:
  - 10 fyzikálních modelů
  - 7 kombinací hledaných parametrů pro každý model
  - 11 optimalizačních algoritmů, které byly vybrány na základě analýzy z předchozího kroku.
  - 2 varianty nastavení každého algoritmu.
  - 2 druhy vstupních signálů do modelů (buzení).
  - 4 varianty vychýlení počátečního odhadu hledaných parametrů.
  - 2 varianty nastavení hranic prohledávaného prostoru parametrů.
- Celkem bylo tedy testováno 2464 variant optimalizační úlohy pro každý z 10 modelů. Výsledky tohoto srovnávacího testu byly následně zpracovány do souhrnného přehledu, přičemž hlavními sledovanými indikátory byla rychlost konvergence algoritmu a také odchylka nalezeného řešení (hodnoty hledaných parametrů) od skutečnosti.
- V návaznosti na výsledky benchmarku byly identifikovány algoritmy s nejlepšími výsledky a byla navržena a otestována nová varianta hybridního optimalizačního algoritmu s robustním a automatizovaným prohledáváním

prostoru parametrů na globální úrovni v kombinaci s lokální optimalizací využívající metody přímého prohledávání.

- Následně byl nový algoritmus porovnán s jinými globálními optimalizačními metodami. Při výsledném srovnání jsou jasně patrné výhody nového algoritmu:
  - Pravděpodobnost nalezení globálního minima je stejná nebo lepší.
  - Počet potřebných simulací (výpočtů hodnotící funkce) je výrazně menší.
  - Algoritmus umožňuje omezenou i neomezenou dobu výpočtu, stejně jako manuální přerušování uživatelem.
  - Algoritmus je připraven na případné rozšíření o volitelné kroky či použití jiné lokální prohledávací metody.

### **Vytvoření algoritmů pro detekci a analýzu nevhodně navržených úloh odhadu parametrů**

- V rámci analýzy řešených úloh jako celku byly navrženy algoritmy, umožňující odhalit některé skryté problémy, které mohou bránit úspěšnému nalezení optima:
  - Detekce redundantních parametrů upozorní na případy, kdy jsou parametry v rovnicích vzájemně provázané a nelze je od sebe separovat (viz náš článek [9]). V těchto případech je pak vhodné parametry sloučit a řádově tak snížit obtížnost úlohy.
  - Byl vyvinut algoritmus pro analýzu vstupních (měřených) dat od uživatele, který díky detekci signálů s nulovou či periodickou složkou, případně s nevhodným vzorkováním, umožní zkrácení doby simulace modelu a tím i celého optimalizačního algoritmu.

### **Nový nástroj pro odhad parametrů v inženýrské praxi**

- Výše uvedené výsledky byly prakticky implementovány v podobě sady funkcí a skriptů v programu MATLAB. Tyto funkce jsou určeny jak pro použití koncovým uživatelem (např. v rámci výuky), tak i jako výchozí bod pro další výzkum a vývoj v této oblasti.
- Vytvořené skripty tedy nejsou pouze oddělenými, samostatně funkčními bloky, ale umožňují propojení do celku – nového nástroje pro odhad parametrů, jehož vývoj stále probíhá. Bez ohledu na to lze však už nyní použít vyvinuté funkce a nástroje pro efektivnější řešení úlohy hledání parametrů.

Závěrem lze konstatovat, že všechny cíle vytyčené cíle se podařilo splnit, přičemž v rámci této dizertační práce vznikla nová sada mechatronických modelů, nový hybridní optimalizační algoritmus a sada pomocných algoritmů pro úlohu odhadu parametrů.

Další předpokládaný vývoj v této oblasti spočívá v transformaci praktických výsledků této práce do podoby nového toolboxu pro odhad parametrů v programu MATLAB, rozšíření hybridního algoritmu o další funkcionalitu, a rozšíření sady pomocných algoritmů pro detekci nevhodného nastavení úlohy (významnost parametrů, interaktivní asistent pro nastavení optimalizační úlohy apod.).

## 9 SEZNAM ZDROJŮ A POUŽITÉ LITERATURY

- [1] FIGUEIRA, Gonçalo a Bernardo ALMADA-LOBO. Hybrid simulation-optimization methods: A taxonomy and discussion [online]. 2014 [vid. 2018-11-15]. Dostupné z: doi:10.1016/j.simpat.2014.03.007
- [2] DRÉO, Johann, Alain PÉTROWSKI, Patrick SIARRY a Eric TAILLARD. *Metaheuristics for Hard Optimization* [online]. Berlin/Heidelberg: Springer-Verlag, 2006. ISBN 3-540-23022-X. Dostupné z: doi:10.1007/3-540-30966-7
- [3] JANGA REDDY, M. a D. NAGESH KUMAR. Evolutionary algorithms, swarm intelligence methods, and their applications in water resources engineering: A state-of-the-art review. *H2Open Journal* [online]. 2020, **3**(1), 135–188. ISSN 26166518. Dostupné z: doi:10.2166/H2OJ.2020.128/699810/H2OJ2020128.PDF
- [4] *What are gradient descent and stochastic gradient descent?* [online]. [vid. 2022-06-18]. Dostupné z: <https://sebastianraschka.com/faq/docs/gradient-optimization.html>
- [5] LEWIS, Robert Michael, Virginia TORCZON a Michael W. TROSSET. Direct search methods: then and now. *Journal of Computational and Applied Mathematics* [online]. 2000, **124**(1–2), 191–207. ISSN 0377-0427. Dostupné z: doi:10.1016/S0377-0427(00)00423-4
- [6] KOLDA, Tamara G., Robert Michael LEWIS a Virginia TORCZON. Optimization by direct search: New perspectives on some classical and modern methods. *SIAM Review* [online]. 2003, **45**(3), 385–482. ISSN 00361445. Dostupné z: doi:10.1137/S003614450242889
- [7] HAN, Zhong-Hua a Ke-Shi ZHANG. Surrogate-Based Optimization. *Real-World Applications of Genetic Algorithms* [online]. 2012. Dostupné z: doi:10.5772/36125
- [8] MAKKAR, C., W. E. DIXON, W. G. SAWYER a G. HU. A new continuously differentiable friction model for control systems design. *IEEE/ASME International Conference on Advanced Intelligent Mechatronics, AIM* [online]. 2005, **1**, 600–605. Dostupné z: doi:10.1109/AIM.2005.1511048
- [9] NAJMAN, Jan, Martin BRABLC, Matej RAJCHL, Michal BASTL, Tomáš SPÁČIL a Martin APPEL. Monte Carlo Based Detection of Parameter Correlation in Simulation Models. *Advances in Intelligent Systems and Computing* [online]. 2020, **1044**, 54–61 [vid. 2022-05-06]. ISSN 21945365. Dostupné z: doi:10.1007/978-3-030-29993-4\_7

# **ŽIVOTOPIS AUTORA**

## **Osobní údaje**

Jméno: Jan Najman  
Datum narození: 13. 8. 1990  
Občanství: ČR  
Bydliště: Brno  
Email: jan.najman@vutbr.cz

## **Vzdělání**

2015 – souč. Vysoké učení technické v Brně; Fakulta strojního inženýrství  
Studijní program: Aplikované vědy v inženýrství (Doktorský)  
Obor: Inženýrská mechanika

2013 – 2015 Vysoké učení technické v Brně; Fakulta strojního inženýrství  
Studijní program: Aplikované vědy v inženýrství (Magisterský)  
Obor: Mechatronika

2010 -2013 Vysoké učení technické v Brně; Fakulta strojního inženýrství  
Studijní program: Aplikované vědy v inženýrství (Bakalářský)  
Obor: Mechatronika

2002 -2010 Gymnázium T.G. Masaryka, Hustopeče

## **Zaměstnání**

2016 – souč. Vysoké učení technické v Brně, Fakulta strojního inženýrství  
Pozice: Odborný asistent  
Člen řešitelského týmu 3 projektů TAČR, 1 projektu MPO a několika projektů specifického výzkumu.

2018 – souč. MECHSOFT s.r.o.  
Pozice: Vývojář hardware/software

## **Jazykové znalosti**

Angličtina: pokročilá, Němčina: základní

## **Technické znalosti a dovednosti**

Znalost programovacích jazyků a prostředí – MATLAB/Simulink (pokročilý), LabVIEW (středně pokročilý), C (mírně pokročilý).  
Další technické znalosti a dovednosti – Design PCB, Programování PLC, Pneumatické systémy.

## **ABSTRAKT**

Dizertační práce se zabývá vývojem nového hybridního optimalizačního algoritmu pro mechatronické modely. Úvodní kapitoly se věnují obecnému popisu problematiky odhadu neznámých parametrů systému, na základě vytvořeného matematického modelu a naměřených dat. Dále je uveden přehled a stručný popis dostupných optimalizačních algoritmů, které jsou vhodné pro řešení tohoto typu problému. Na základě rešeršní studie jsou pak formulovány konkrétní cíle práce. Ve druhé části práce je popsána nově vytvořená sada mechatronických modelů vytvořených pomocí nástrojů pro fyzikální modelování. Následně je s pomocí těchto modelů proveden srovnávací test rychlosti a úspěšnosti vybraných optimalizačních algoritmů. Na základě výsledků tohoto testu je navržen design nového hybridního algoritmu, který je následně otestován a srovnán s ostatními algoritmy. Na závěr práce je představeno několik nových pomocných funkcí a nástrojů pro detekci a analýzu nevhodně navržených úloh odhadu parametrů.

## **ABSTRACT**

The thesis deals with the development of a new hybrid optimization algorithm for mechatronic models. The introductory chapters are devoted to a general description of the problem of estimating unknown system parameters, based on the developed mathematical model and measured data. Furthermore, an overview and a brief description of available optimization algorithms that are suitable for solving this type of problem is given. Based on the research study, the specific objectives of the paper are then formulated. In the second part of the thesis, a newly developed set of mechatronic models created using physical modelling tools is described. Subsequently, a comparative test of the speed and success rate of the selected optimization algorithms is performed using these models. Based on the results of this test, the design of a new hybrid algorithm is proposed, which is then tested and compared with the other algorithms. Finally, several new auxiliary functions and tools are presented to detect and analyze improperly designed parameter estimation problems.