

**UNIVERZITA PARDUBICE
DOPRAVNÍ FAKULTA JANA PERNERA
KATEDRA INFORMATIKY V DOPRAVĚ**

**FYZIKÁLNÍ ANALOGIE POPISU MODELŮ
SPOJITÉ SIMULACE**

DIPLOMOVÁ PRÁCE

Vedoucí diplomové práce:
Ing. Oktavián Strádal, Ph.D.

Diplomant:
Bc. Radim Šejbl

2006

UNIVERSITY OF PARDUBICE
THE JAN PERNER TRANSPORT FACULTY
DEPARTMENT OF INFORMATICS IN TRANSPORT

**PHYSICAL ANALOGY IN DESCRIPTION OF
CONTINUOUS SIMULATION MODELS**

THESIS

Supervisor:

Ing. Oktavián Strádal, Ph.D.

Author:

Bc. Radim Šejbl

2006

Prohlašuji:

Prohlašuji, že jsem tuto práci vypracoval samostatně. Veškeré literární prameny a informace, které jsem v práci využil, jsou uvedeny v seznamu použité literatury a pramenech.

Byl jsem seznámen s tím, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorský zákon, zejména se skutečností, že Univerzita Pardubice má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle § 60 odst. 1 autorského zákona a s tím, že pokud dojde k užití této práce mnou nebo bude poskytnuta licence o užití jinému subjektu, je Univerzita Pardubice oprávněna ode mne požadovat přiměřený příspěvek na úhradu nákladů, které na vytvoření díla vynaložila, a to podle okolností až do jejich skutečné výše.

Souhlasím s prezenčním zpřístupněním své práce v Univerzitní knihovně Univerzity Pardubice.

V Pardubicích dne 1.8.2006

.....
Bc. Radim Šejbl

Poděkování

Děkuji Ing. Oktaviánu Strádalovi, Ph.D. za ochotu a pomoc při zpracování mé diplomové práce.

Abstrakt

Tato práce je zaměřena na problematiku fyzikálních analogií modelů spojité simulace. Jsou zde také popsány různé způsoby a možnosti popisu a řešení spojitých systémů. Dále se zabývám otázkou doporučení vhodného softwaru pro simulaci dynamických systémů. Důležitou částí práce je přehled analogických fyzikálních modelů a ukázka analogií na modelech vytvořených v programu Simulink.

Abstract

This thesis is focused on problems of physical analogy of continuous simulation models. There are also described different ways and possibilities of description and solution of dynamic systems. Next part is engaged in recommendation of suitable software for dynamic systems simulation. Important part is a view of analogical physical models and illustration of analogies on models created in Simulink.

Obsah

SEZNAM POUŽITÝCH ZKRATEK A SYMBOLŮ.....	1
1 ÚVOD	2
2 MATEMATICKÉ PROSTŘEDKY A METODY PRO SIMULACI	3
2.1 TEORIE MNOŽIN A TRANSFORMACÍ.....	3
2.2 TEORIE FUZZY MNOŽIN	3
2.3 Maticová algebra.....	4
2.4 Diferenční a diferenciální rovnice.....	5
2.4.1 Diferenční rovnice	5
2.4.2 Diferenciální rovnice	5
3 ALGORITMIZACE SIMULAČNÍHO MODELU	7
3.1 Zobrazení stavů simulovaného modelu	7
3.2 Zobrazení stavových změn a času	8
4 SPOJITÝ SIMULAČNÍ MODEL	9
4.1 Definice spojitého dynamického systému a jeho řešení.....	9
4.2 Metody numerické integrace	11
4.2.1 Runge-Kutty.....	13
4.2.2 Víceuzlové metody	13
4.2.3 Tuhé (stiff) soustavy	14
5 PROGRAMOVACÍ PROSTŘEDKY PRO SIMULACI	17
5.1 Základní klasifikace simulačních jazyků	18
5.2 Jazyky pro objektové orientované programování.....	19
5.3 Použití software pro simulaci	20
5.3.1 MATLAB a SIMULINK.....	20
5.3.1.1 MATLAB.....	20
5.3.1.2 SIMULINK.....	23
5.3.2 Jiný software pro spojitou simulaci	26
5.3.2.1 VisSim.....	27
5.3.2.2 20-Sim	28
6 FYZIKÁLNÍ ANALOGIE.....	30
6.1 Modelování spojitých systému pomocí fyzikální analogie	35
6.2 Elektrické analogy fyzikálních obvodů.....	38
6.2.1 Elektrické analogy dynamických obvodů	38
7 TVORBA MODELŮ A UKÁZKY ANALOGIÍ	42

7.1	TVORBA MATEMATICKÝCH MODELŮ MECHANICKÝCH SYSTÉMŮ.....	42
7.1.1	POPIS PRVKŮ MODELU MECHANICKÉHO SYSTÉMU.....	44
7.2	TVORBA MATEMATICKÝCH MODELŮ ELEKTRICKÝCH SYSTÉMŮ.....	45
7.2.1	POPIS PRVKŮ MODELU ELEKTRICKÉHO SYSTÉMU	46
7.3	UKÁZKA ANALOGIÍ NA MODELECH – PRVNÍ PŘÍKLAD	48
7.3.1	POROVNÁNÍ MODELŮ VYTVOŘENÝCH V SIMULINKU	50
7.3.2	ANALÝZA VÝSLEDKŮ	53
7.3	DRUHÁ UKÁZKA ELEKTROMECHANICKÉ ANALOGIE.....	57
8	ZPRACOVÁNÍ SOUBORŮ Z CD PŘÍLOHY.....	61
9	ZÁVĚR	62
	SEZNAM TABULEK.....	63
	SEZNAM OBRÁZKŮ	63
	PŘÍLOHA – OBSAH PŘILOŽENÉHO DATOVÉHO NOSIČE (CD)	64
	SEZNAM POUŽITÝCH INFORMAČNÍCH ZDROJŮ	65

Seznam použitých zkratek a symbolů

a	zrychlení
C	kapacita
F	síla
$f(x)$	hustota pravděpodobnosti
$F(x)$	distribuční funkce
G	elektrická vodivost
H	hydraulická indukčnost
i	elektrický proud
L	indukčnost
m	hmotnost
p	tlak
Q	teplo
R	odpor, hydraulický odpor
R_m	mechanický odpor
t	čas
T	absolutní teplota
u	elektrické napětí
v	rychlost
φ	fázový posuv
Φ	tepelný tok
τ	časové posunutí
ρ	hustota

1 Úvod

Fyzikální analogie je dnes často používanou technikou, která se díky svému neustálému zdokonalování dokáže vyrovnat i ostatním metodám řešení a popisu modelů spojitě simulace. V některých případech může mít dokonce výjimečné postavení a to především díky mnoha dobrým vlastnostem, které tato metoda má. Jsou jimi jednoduchost, nenáročnost na použité prostředky a bezprostřednost.

Cílem této práce je bližší seznámení s touto technikou popisu modelů spojitě simulace. Dále jsem sestavil přehled fyzikálních analogických modelů a jejich popisů pro různé fyzikální soustavy, především pro elektrickou a mechanickou.

Pokud chceme řešit nějaký dynamický systém, tak ho nejdříve popíšeme jednou nebo více rovnicemi. Ty představují matematický model. Poté se nabízí několik možností, jak dále postupovat a řešit daný systém. Jednou z nich je použití fyzikální analogie, kterou se tato práce zabývá. Dočteme se ale také o jiných způsobech, abychom se v hlavním tématu dozvěděli, proč je způsob popisu modelů spojitě simulace pomocí fyzikální analogie tak zajímavý a především výhodný. V poslední části práce jsem se pomocí několika modelů v programu Simulink snažil dokázat, že tato metoda opravdu funguje a jistě si zaslouží pozornost.

Tato práce předpokládá znalost základních pojmů z oblasti spojitě simulace jako je systém, model, modelování, simulace atd. Tyto základní pojmy je možné najít například v [3].

2 Matematické prostředky a metody pro simulaci

V této kapitole jsou ukázány možnosti popisu modelů.

2.1 Teorie množin a transformací

První z matematických prostředků, které zmíním, je Teorie množin a transformací. Užívá se především k popisu změn stavu systému. Předpokládejme, že jsme definovali množinu, jejíž prvky reprezentují všechny možné stavy systému, které se mohou realizovat v průběhu jeho vývoje. Tato množina nemusí být přirozeně konečná, jejími prvky mohou být např. písmena nějaké abecedy. Dále zvolíme vhodný časový interval a ke každému ze stavů přiřadíme stav, do něhož by uvažovaný systém v průběhu zvoleného časového intervalu mohl přejít. Směr přechodu znázorníme šipkou. Takto vznikne model stavových změn (stavových přechodů), jenž kvalitativně popisuje vývoj sledovaného systému. [3]

Takto vzniklé schéma můžeme též považovat za orientovaný graf, jehož uzly představují stavy systému a hrany možné stavové přechody. Setkáváme se stále se složitějšími objekty. Množství informace potřebné k jejich popisu se přirozeně rovněž zvětšuje. Přesné modely složitých systému se stávají nezpracovatelnými pomocí konvenčních matematických prostředků, proto je třeba hledat prostředky nové. Příčina tohoto stavu spočívá v tzv. principu inkompatibility. Roste-li složitost systému, klesá naše schopnost formulovat přesné a významné soudy o jeho chování, až je dosaženo hranice, za níž se přesnost a relevantnost prakticky vylučují. [3]

2.2 Teorie fuzzy množin

Dalším vhodným prostředkem pro popis nepřesných (vágních) pojmů je teorie fuzzy množin. Nejsme-li schopni přesně vymezit hranice nějaké třídy určené vágním pojmem, pak přiřadíme každému prvku míru jeho příslušnosti k dané třídě.

Bude-li škála pro tuto míru uspořádaná, pak zřejmě platí následující tvrzení. Čím menší je míra příslušnosti daného prvku k uvažované třídě, tím blíže je tento prvek hranici třídy. Tato míra se nazývá stupeň příslušnosti prvku do uvažované třídy a třída, jejíž každý prvek je charakterizován stupněm příslušnosti do ní, se nazývá fuzzy množina. Teorie fuzzy množin lze využít i při zkoumání reálných systémů. Rozlišují se dva druhy fuzzy systémů - skutečné fuzzy systémy, jejichž přesný teoretický popis neexistuje a systémy, jež jsou natolik složité, že je nejsme schopni klasickými metodami přesně popsat.

Teorie fuzzy množin vytváří jakýsi most mezi verbálním a matematickým modelem reálného systému. [3]

2.3 Maticová algebra

Neméně důležitou roli při vytváření modelu hraje lineární algebra, zejména maticová algebra. Matic se využívá především k matematickému popisu struktury systému a interakcí mezi jednotlivými prvky systému. Například velikosti populací v n -složkovém systému můžeme popsat pomocí sloupcového vektoru

$$(x_1, x_2, \dots, x_n)^T$$

nebo řádkového vektoru

$$(x_1, x_2, \dots, x_n),$$

kde x_i značí velikost i -té populace. [3]

Interakce mezi jednotlivými populacemi se přehledně vyjadřují pomocí interakční matice:

$$\begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{pmatrix}$$

v níž prvek a_{ij} ($i, j = 1, 2, \dots, n$) reprezentuje interakci mezi i -tou a j -tou populací. Maticová reprezentace je např. typická pro Leslieho diskretní model rozvoje populace, jejíž natalita i mortalita jsou funkcí věku organismu. [3]

2.4 Diferenční a diferenciální rovnice

Diferenční nebo diferenciální rovnice se při tvorbě modelů uplatňují nejčastěji. Pomocí těchto rovnic se simulují časové změny stavových proměnných systému. Jelikož se v práci zabývám simulací spojitého systému, které jsou zpravidla vždy popsány jednou nebo více diferenciálními rovnicemi, je dobré, aby této části byla věnována pozornost.

2.4.1 Diferenční rovnice

Diferenční rovnice reprezentují změny, které se uskutečňují v průběhu diskretních časových úseků. Je-li V_t hodnota určité stavové proměnné V v čase t , pak diferenční rovnice

$$V_{t+1} = f(V_t, t)$$

určuje hodnotu této proměnné (jako funkci původní hodnoty V_t a čase t) po uplynutí časové jednotky. Pokud se dynamika systému popisuje soustavou diferenčních rovnic, V_t představuje vektor stavových proměnných v čase t a $f(V_t, t)$ soustavu rovnic. Diferenční rovnice jsou mimořádně vhodné pro vyjádření časových zpoždění, např.

$$V_{t+1} = f(V_t, V_{t-1}, \dots, t). [3]$$

2.4.2 Diferenciální rovnice

Diferenciální rovnice popisují změny, které probíhají v **čase spojitě**. Mají obvykle tvar

$$\frac{dV}{dt} = f(V, t),$$

přičemž V , $\frac{dV}{dt}$ i $f(V, t)$ mohou být chápány jako vektory. Diferenciální rovnice uvedeného typu vyjadřuje rychlost změny stavové proměnné V jako funkci okamžitých hodnot této stavové proměnné a času. Vedle obyčejných diferenciálních rovnic a jejich soustav se při modelování systému poměrně často setkáváme i s parciálními diferenciálními rovnicemi a jejich soustavami (např. při studiu

dynamiky populací v nehomogenním prostředí). V případě stochastických modelů se využívá stochastických diferenciálních rovnic, jež mají tvar

$$dV_t = f(V_t, t) + g(V_t, t)dW_t$$

v němž $\{V_t, t \geq 0\}$ značí náhodný proces a dW_t diferenciál Wienerova procesu (zjednodušené náhodné veličiny s normálním rozdělením, jejíž střední hodnota je nulová a rozptyl roven dt). Stochastické diferenciální rovnice se uplatňují např. v genetice a při modelování aktivity neuronu.

Kromě již zmíněných matematických prostředků se při modelování systému často uplatňuje i matematická logika (klasická logika, vícehodnotová logika, fuzzy logika, temporální logika), integrodiferenciální a integrální rovnice (např. při studiu dynamiky populace se započtením zpoždění ve vzájemných interakcích), orientované grafy (při modelování transportních jevů), strukturní termodynamika (zobecněné termodynamické síly a toky při modelování transportu). V podkapitole 2.4 jsem čerpal z [3].

3 Algoritmizace simulačního modelu

Jelikož je součástí práce vytvoření modelů spojité simulace a následná práce s nimi, budu se v této kapitole zabývat věcmi jako je zobrazení stavového času a změn a stavů simulovaného modelu. Pro formální matematický popis činnosti výchozího systému není prakticky možné navrhnout obecně platná pravidla. Určitě však má smysl pokusit se navrhnout jakousi univerzální metodiku pro algoritmizaci simulačního modelu (simulátoru) a jeho počítačové implementace. Taková univerzální metodika se přirozeně nemůže týkat těch částí algoritmizace, které jsou specifické pro konkrétní simulační úlohy, ale jen tzv. simulačního jádra, tj. té části, jež je prakticky společná všem simulačním programům. V této části jsem se zabýval pouze takovými datovými strukturami simulačního jádra, které jsou společné pro simulační modely diskrétního i spojitého typu. Hlavní úkoly při konstrukci simulačního jádra jsou [3]:

1. navrhnout strukturu dat pro reprezentaci stavu simulovaného systému;
2. navrhnout operátory nad touto datovou strukturou, které realizují změny stavu systému;
3. zobrazit čas modelu a jeho průběh;
4. zajistit synchronizaci stavových změn systému tak, aby tyto změny probíhaly v určitém pořadí a při určitých hodnotách času nebo v okamžicích, kdy je splněna určitá podmínka týkající se stavu či konfigurace modelu.

3.1 Zobrazení stavů simulovaného modelu

Pro reprezentaci stavu výchozího systému se používají nejrůznější datové struktury v závislosti na typu zvoleného programovacího jazyka. V nejjednodušším případě odpovídají skalárním stavovým veličinám jednoduché proměnné a vektorům datová pole. Složitější datové struktury se využívají v případech, kdy se v průběhu času mění struktura systému nebo počet jeho prvků (např. při simulaci systému hromadné obsluhy). V těchto případech je výhodné použít takových programovacích jazyků, jež umožňují přidělovat paměť exemplářům datových struktur dynamicky až

v průběhu výpočtu. Objektově orientované programovací jazyky dovolují uživateli definovat (deklarovat) prakticky libovolně složité datové struktury a tyto dále obohacovat (koncepte tříd a podtříd). Např. při simulaci systému hromadné obsluhy je velmi účelné použití spojových seznamů, které reprezentují fronty požadavku před obslužnými linkami. [3]

3.2 Zobrazení stavových změn a času

Každá změna stavu je obecně realizována nějakým operátorem, který pracuje nad příslušnou datovou strukturou. Takový operátor sestává z posloupnosti příkazu, jejíž forma (makro, procedura, podprogram) je dána použitým programovacím jazykem. V moderních (objektově orientovaných) jazycích se tyto operátory logicky spojují přímo s odpovídajícími datovými strukturami (např. metody v deklaraci tříd jazyku Simula). Místo termínu „operátor“ se někdy používá slova „událost“, ale není to vhodné, protože termín „událost“ se používá v oblasti simulace v jiném významu. [3]

Zobrazení času představuje specifický problém simulace. Čas výchozího (simulovaného) systému se zobrazuje v modelu pomocí aritmetické proměnné (buď typu real nebo typu integer). Pro tuto veličinu se zavádí speciální název - simulární čas. Nedoporučuje se používat v češtině dosti rozšířeného termínu „simulovaný čas“, ani termínu „modelový čas“, resp. „systémový čas“. Přitom musí být splněny určité podmínky [3]:

- hodnota simulárního času nesmí v průběhu výpočtu klesat,
- děje v simulačním modelu závisejí na simulárním čase stejným způsobem, jako jejich vzory ve výchozím systému na toku přirozeného času.

Hodnota simulárního času by neměla být uživateli přímo přístupná. V simulačních programovacích jazycích se zpravidla dodržuje zásada, že uživatel může hodnotu simulárního času zjišťovat (např. prostřednictvím vhodné funkční procedury), nikoliv však měnit.

4 Spojitý simulační model

Spojité simulační modely jsou charakterizovány tím, že všechny stavové proměnné nabývají hodnot z nějakého nedegenerovaného intervalu a v průběhu času se mění pouze spojitě. [3]

Chování (evoluce) spojitého modelu se popisuje pomocí soustavy diferenciálních rovnic, ať už obyčejných nebo parciálních. Simulační čas je diskretizován, a to tak hustě, aby numerické řešení diferenciálních rovnic splňovalo požadavky na přesnost výpočtu. Uchovává se jen okamžitá hodnota simulárního času, tato hodnota se postupně zvyšuje o délku integračního kroku. [3]

Typické úlohy řešené pomocí spojitých simulačních modelů:

- kmitání mechanických systémů,
- řešení elektrických a elektronických obvodů,
- kompartmentové systémy,
- dynamika populací.

4.1 Definice spojitého dynamického systému a jeho řešení

Vycházíme z rigorózní definice spojitého dynamického systému jakožto modelu nějakého reálného systému se spojitě se měnícími hodnotami atributu.

Ve zdroji [3] se spojitý dynamický systém definuje takto:

Spojitý dynamický systém je matematická struktura tvořená:

- otevřenou množinou $\Omega \subset \mathbb{R}^n$ (tzv. stavovým prostorem),
- množinou $\{f_1, f_2, \dots, f_n\}$ reálných funkcí definovaných na Ω ,
- soustavou obyčejných diferenciálních rovnic 1. řádu

$$\frac{dx_i}{dt} = f_i(x_1, x_2, \dots, x_n),$$

kde $i = 1, 2, \dots, n$.

Proměnné x_1, x_2, \dots, x_n se nazývají stavové proměnné dynamického systému a výše uvedené rovnice představují pohybové rovnice tohoto systému. Přírozené číslo n nazveme dimenzí uvažovaného systému. O stavovém prostoru se předpokládá, že je jistou otevřenou podmnožinou n -rozměrného euklidovského prostoru R^n . Okamžitý stav dynamického systému je pak reprezentován v tomto stavovém prostoru bodem, jehož kartézské souřadnice odpovídají hodnotám jednotlivých stavových proměnných. Pro dynamický systém s danými pohybovými rovnicemi platí, že okamžitá rychlost změny kterékoli stavové proměnné v daném okamžiku nezávisí explicitně na čase, ale pouze na okamžitých hodnotách stavových proměnných. Takovým dynamickým systémům říkáme autonomní. [3]

Definice [3]: Řešením dynamického systému je množina reálných funkcí $\{x_1(t), x_2(t), \dots, x_n(t)\}$, které mají následující vlastnosti:

- funkce $x_i(t)$, $i = 1, 2, \dots, n$, jsou definovány na nějakém nedegenerovaném otevřeném intervalu $T \subseteq R$;
- derivace $\frac{dx_i(t)}{dt}$, $i = 1, 2, \dots, n$, jsou také definovány na intervalu T ;
- množina funkcí $\{x_1(t), x_2(t), \dots, x_n(t)\}$ splňuje soustavu diferenciálních rovnic v každém bodě $t \in T$.

Křivky $C(t) = \{x_1(t), x_2(t), \dots, x_n(t); t \in T\}$, které jsou určeny partikulárními řešeními daného dynamického systému, se nazývají trajektorie (orbity) tohoto systému. Množina všech trajektorií dynamického systému se nazývá jeho fázovým portrétem. Mezi trajektoriemi daného dynamického systému zaujímají zvláštní postavení [3]:

1. kritické body, pro něž platí $x_i(t) = c_i$, přičemž konstanty c_i jsou řešeními soustavy rovnic

$$f_i(x_1, x_2, \dots, x_n) = 0 \quad (i = 1, 2, \dots, n);$$

2. periodické trajektorie s periodou $\tau > 0$, které splňují pro všechna i , $i = 1, 2, \dots, n$, podmínky

$$x_i(t) \neq x_i(0) \text{ pro } 0 < t < \tau,$$

$$x_i(\tau) = x_i(0).$$

4.2 Metody numerické integrace

Abstraktním modelem spojitého dynamického systému je obecně soustava diferenciálních a algebraických rovnic, proto je třeba věnovat zvláštní pozornost numerickým metodám pro jejich řešení. Numerické řešení algebraických ani transcendentních rovnic nepředstavuje zpravidla vážnější problém, pokud máme k dispozici vhodnou univerzální metodu, např. Newtonovu. Naproti tomu však různorodost modelovaných systému vyžaduje, aby byly programové prostředky pro spojitou simulaci vybaveny řadou univerzálních i speciálních metod pro numerickou integraci. [3]

V této kapitole jsem se zabýval pouze numerickými metodami řešení obyčejných diferenciálních rovnic a jejich soustav.

Uvažujme pro jednoduchost soustavu n obyčejných diferenciálních rovnic prvního řádu

$$\frac{dy_i}{dt} = f_i(t, y_1, y_2, \dots, y_n)$$

s počátečními podmínkami $y_i = y_{i0}$, kde $i = 1, 2, \dots, n$.

Tuto počáteční (Cauchyho) úlohu budeme zapisovat v maticovém tvaru

$$y_0 = f(t, y)$$

s počáteční podmínkou $y(t_0) = y_0$. Dále předpokládejme, že jsou splněny podmínky existence a jednoznačnosti řešení v oblasti $\Omega \times I$, kde Ω je stavový prostor uvažovaného systému a I nějaký interval reálných čísel. [3]

Numerickým řešením počáteční úlohy se rozumí posloupnost $\{y_i\}$ hodnot

$$y_0 = y(t_0), y_1 = y(t_1), \dots, y_k = y(t_k),$$

které odpovídají hodnotám t_i , $i = 1, 2, \dots, k$, nezávisle proměnné t (zpravidla času) v intervalu $\langle t_0, t_k \rangle$. Hodnota výrazu $h = t_{i+1} - t_i$ přitom udává velikost integračního kroku. Exaktní řešení uvažované počáteční úlohy budeme značit

$$Y_i = Y(t_i), i = 1, 2, \dots, k.$$

Cílem numerických metod je nalezení numerického řešení. Přitom se požaduje, aby posloupnost $\{y_i\}$ konvergovala pro $h \rightarrow 0$ k exaktnímu řešení $Y(t_i)$. Rozlišujeme dva základní typy metod řešení soustavy [3]:

- Metody, kde se hodnoty funkce $f(t, y)$ počítají jen v bodech $[t_i, y_i]$, přičemž y_i je hodnota numerického řešení v bodě $t = t_i$. Do této skupiny patří především tzv. **víceuzlové metody**.
- Metody, kde se hodnoty funkce $f(t, y)$ počítají navíc v bodech ležících mezi jednotlivými body $[t_i, y_i]$, $i = 1, 2, \dots, k$. Zástupci těchto metod jsou především metody **Rungeho-Kutty**.

V obou případech je posloupnost hodnot y_i výsledkem postupné extrapolace, přičemž již výchozí hodnoty v každém kroku jsou zatíženy lokální chybou. Tato chyba je součtem dvou příspěvků [3]:

- chyby metody (truncation error), která je způsobena respektováním jen konečného počtu členů Taylorova rozvoje funkce f ,
- zaokrouhlovací chyby (rounding error), jež souvisí s omezenou délkou slova v paměti počítače.

Chyba jednoho kroku pak přirozeně ovlivňuje výsledky kroku následujících. Celková chyba ε po realizaci n kroků, tzv. akumulovaná chyba po n krocích, je dána vztahem

$$\varepsilon = Y_n - y_n.$$

Kvalita použité numerické metody se hodnotí podle těchto základních kritérií [3]:

- přesnost metody (velikost lokální chyby a prostředky pro její odhad),
- stabilita metody,
- časová náročnost výpočtu,
- nároky na operační paměť počítače.

Metoda je absolutně stabilní pro daný krok h a danou soustavu diferenciálních rovnic, jestliže chyba vzniklá při výpočtu hodnoty y_n se nezvětšuje při výpočtu následujících hodnot y_k , $k > n$. K vyšetřování absolutní stability se používá testovací rovnice $y_0 = \lambda y$, kde λ je konstanta (obecně komplexní číslo). Množina hodnot

$$\tilde{h} = h \lambda,$$

pro něž je metoda absolutně stabilní, se nazývá obor absolutní stability příslušné soustavy.

4.2.1 Runge-Kutty

Těmto metodám se také říká jednouzlové, protože k výpočtu hodnoty y_{n+1} stačí znát pouze hodnotu y_n v bezprostředně předcházejícím uzlu. Vychází se obecně ze vztahu

$$y_{n+1} - y_n = \sum_{i=1}^p w_i k_i,$$

kde w_i jsou konstanty a

$$k_i = hf(t_n + a_i h, y_n + \sum_{j=1}^{i-1} b_{ij} k_j)$$

pro $i = 1, 2, \dots, p$, $h = t_{n+1} - t_n$, a_i a b_{ij} jsou konstanty, přičemž $a_1 = 0$. Metoda popsaná výše uvedeným vztahem se nazývá p -hodnotová, protože používá právě p hodnot funkce $f(t, x)$. Konstanty w_i , a_i , b_{ij} se spočtou tak, aby získané řešení souhlasilo s Taylorovým rozvojem v bodě $[t_n, y_n]$ až do P -té mocniny integračního kroku h včetně. Taková metoda se pak nazývá metoda **Rungeho-Kutty rádu P** . Pro $p \leq 4$ zřejmě platí $P = p$. [3]

K odhadování přesnosti metod Rungeho-Kutty se používá tzv. metoda polovičního kroku, tj. dvojí výpočet: jednak s krokem h , jednak s krokem $2h$. Všechny metody Rungeho-Kutty mají ohraničený obor absolutní stability; jeho rozsah se zvětšuje s rostoucím řádem metody. Jejich implementace na počítači je jednoduchá, integrační krok lze libovolně měnit. Mají přibližně stejnou, často i vyšší přesnost než metody prediktor-korektor stejného řádu (viz následující odstavec). [3]

4.2.2 Víceuzlové metody

Numerická metoda se nazývá k -uzlová, jestliže k výpočtu hodnoty y_{n+1} používá právě k aproximací bezprostředně předcházejících. Vychází se z obecného vztahu

$$y_{n+1} = \sum_{i=0}^r a_i y_{n-i} + h \sum_{i=1}^r b_i y'_{n-i},$$

kde a_i, b_i jsou konstanty. V uvedeném vztahu platí $k = r + 1$. [3]

Uvedené metody nejsou samostartující. Nejprve je nutno spočítat prvních k hodnot některou z metod typu Rungeho-Kutty nebo jinou jednoduzlovou metodou. Víceuzlová metoda je řádu právě p , jestliže je přesná pro polynomy stupně p . Přitom se vždy požaduje $p \geq 2$. Některé z koeficientů a_i, b_i mohou být rovny nule. V podstatě se rozlišují tři základní skupiny víceuzlových metod [3]:

1. Je-li $b_{-1} = 0$, pak hodnota y_{n+1} je vyjádřena jako lineární kombinace již známých aproximací y_i . Takové metody se nazývají explicitní (přímé nebo prediktorového typu).
2. Je-li ovšem $b_{-1} \neq 0$, pak výše uvedený vztah představuje implicitní rovnici pro y_{n+1} , protože $y'_{n+1} = f(t_{n+1}, y_{n+1})$, a takovou rovnici lze řešit jen iteračně. Metody této skupiny se nazývají implicitní (nepřímé, iterační nebo korektorového typu).
3. Kombinací obou předchozích typů vznikají metody prediktor - korektor. Jejich princip je jednoduchý. Pro první výpočet (predikci) y_{n+1}^P se používá formule explicitní. Následně se určí derivace $y'_{n+1} = f(t_{n+1}, y_{n+1}^P)$ a pak se hledané řešení zpřesňuje (koriguje) pomocí implicitní formule (počítá se y_{n+1}^C). Dvojice metod prediktor-korektor se vybírá tak, aby obě vybrané metody měly stejný řád.

4.2.3 Tuhé (stiff) soustavy

Soustava obyčejných diferenciálních rovnic se nazývá tuhou (stiff) soustavou, jestliže vlastní čísla λ i její Jacobiho matice

$$J = \left(\frac{\partial f_i}{\partial x_j} \right), \quad i, j = 1, 2, \dots, n,$$

jsou značně rozdílná. Je zřejmé, že vlastní čísla matice J závisejí na čase a v průběhu integrace se tedy mění. Tuhá soustava obyčejných diferenciálních rovnic má tyto vlastnosti [3]:

- $\operatorname{Re} \lambda_j < 0, j = 1, 2, \dots, n$, pro všechny hodnoty t ,

- poměr $R = \frac{\max_j |\lambda_j|}{\min_j |\lambda_j|}$

je velké číslo (v typických úlohách z praxe řádu 10^3 až 10^6).

Poměr R je charakteristikou soustavy obyčejných diferenciálních rovnic, nazývá se jejím tuhostním poměrem. U většiny numerických metod stabilita metody vyžaduje omezení integračního kroku h ve tvaru $|h\lambda| < K$, kde K je konstanta charakteristická pro danou metodu a λ je v absolutní hodnotě největší vlastní číslo Jacobiho matice. Pro velké hodnoty λ je tedy nutno volit malý integrační krok.

K řešení tuhých soustav jsou tedy vhodné metody, jejichž oblast absolutní stability je celá levá polorovina, tj. $\operatorname{Re} \lambda h < 0$. Takové metody se nazývají A-stabilní. Pro tyto metody platí [3]:

- Explicitní víceuzlové metody nemohou být A-stabilní.
- A-stabilní implicitní metody mohou být nejvýše 2. řádu.

Požadavek A-stability je příliš silný, neumožňuje používat víceuzlových metod řádu vyššího než 2. Proto byly zavedeny tzv. tuhostně stabilní metody.

Na závěr této kapitoly uvedu několik poznámek k hodnocení jednotlivých numerických metod podle různých kritérií.

- Přesnost metody. Metody vyšších řádů jsou obecně přesnější (mají menší lokální chybu) než metody nižších řádů téhož typu. U víceuzlových metod jsou metody implicitní přesnější než explicitní.
- Stabilita metody. Pro metody Rungeho-Kutty platí, že s rostoucím řádem metody se zvětšuje oblast její absolutní stability. Naproti tomu u Adamsových explicitních metod a metod prediktor - korektor se oblast absolutní stability zmenšuje s rostoucím řádem metody.

- Časová náročnost výpočtu. Rychlost řešení je pro danou hodnotu integračního kroku h nepřímo úměrná řádu metody. Víceuzlové metody jsou při daném h obecně rychlejší než metody jednouzlové.
- Nároky na paměť. Tyto nároky jsou obecně přímo úměrné řádu metody. Pro metody téhož řádu platí, že víceuzlové metody jsou náročnější než metody jednouzlové. U víceuzlových metod mají implicitní metody větší nároky na paměť než metody explicitní.

V této kapitole jsem vycházel převážně z [3].

5 Programovací prostředky pro simulaci

V této kapitole jsem se zabýval analýzou možností výběru vhodného programovacího prostředku pro simulaci. Je zřejmé, že simulační programy jsou velmi často složité – výpočet podle nich totiž musí být analogií nějakého složitého děje, který chceme poznat a který bývá výslednicí velmi komplexních časoprostorových a jiných vztahů. Pro překonávání potíží se sestavováním složitých programů existuje pět způsobů, které lze v principu aplikovat i na implementaci simulačních programů [3]:

- sestavování programů, do nichž zabudujeme předem připravené podprogramy, které se chovají navenek zdánlivě jednoduše a názorně a které při tom provádějí dosti složité výpočty (tedy použití podprogramů jako stavebních kamenů programu);
- použití předem připraveného „univerzálního“ programu, který je řízen daty tak, že některá z nich jeho běh směřují různými cestami;
- použití programovacích jazyků tak, že texty v nich sestavené a popisující více méně čitelně požadavky na to, co se má počítat, jsou interpretovány speciálním programem, tzv. interpretačním programem neboli interpretem;
- použití programovacích jazyků tak, že texty v nich sestavené a popisující více méně čitelně požadavky na to, co se má počítat, jsou překládány do odpovídajících programů ve strojovém kódu;
- použití objektově orientovaných programovacích prostředků jakožto základu pro definice adekvátních problémově orientovaných programovacích prostředků.

Hned na počátku dalšího rozboru uvedu, že třetí bod lze z následujících úvah vypustit, neboť — ačkoliv je interpretace textu historicky chápána jako samostatný způsob řešení potíží programování — jde o zvláštní případ bodu 2, což se v posledních letech stále více prosazuje: interpretace textu u programovacích jazyků

interpretačním programem je zvláštním případem zpracování vstupních dat „univerzálním“ programem.

Dnes lze vyloučit z dalších úvah i bod první, protože přípravu podprogramů vhodných pro pomoc v některých oblastech může lépe nahradit posílání zpráv mezi jednotlivými objekty s tím, že reakce příjemců zpráv jsou adekvátní vyvolání podprogramu, mezi jejichž parametry jsou tito příjemci. Např. vyvolání podprogramu $F(X, Y)$ lze nahradit posláním zprávy objektu X , aby provedl metodu M s parametrem Y , kde M je jen nepatrně preformulovaná F . V minulosti (v šedesátých a sedmdesátých letech) bylo vyvinuto několik set „balíků podprogramů“, které podporovaly tvorbu počítačových simulačních modelů, z nichž aplikace některých přetrvává někde až dodnes (např. GASP). Univerzální či spíše univerzálnější „simulátory“, pomáhající podle bodu 2, se uplatňují v různých aplikačních oborech (např. v tvorbě počítačů a ve strojírenské výrobě). Dnes obvykle tyto simulátory reagují na kombinovaná alfanumerická a grafická vstupní data, jejichž rozmanitost je tak velká, že lze těžko vyslovit nějaké společné zásady pro simulační prostředky sledující zásadu 2; shrnující popis je znemožňován i prudkým vývojem těchto prostředků. U nás jsou z nich známy např. TAYLOR a WITNESS pro simulaci diskrétních výrob, ve světě však mají podobné prostředky nejrůznější obory lidské činnosti, např. dispečeri energetických sítí, letištní odborníci nebo pracovníci v organizaci zdravotnictví. [3]

Poslední dva způsoby jsou popsány podrobněji v následujících kapitolách.

5.1 Základní klasifikace simulačních jazyků

V systémech existují obecně transakce a permanentní prvky čili aktivity. Systémy, které obsahují prvky obou těchto kategorií, se nazývají systémy typu AT (zkratka Aktivita-Transakce) nebo zkráceně AT-systémy. Lze si představit systémy tak proměnné v čase, že v nich žádné aktivity nejsou, tj. že všechny jejich prvky chápeme jako transakce. Tyto systémy nazýváme systémy typu T. Existuje však i opačný extrém, systémy typu A čili A-systémy, v nichž nejsou žádné transakce. V takových systémech zanedbáváme v podstatě všechny strukturní změny a celý děj, který v nich v čase probíhá, abstrahujeme do nějakých změn hodnot atributu (většinou jde o atributy numerické, resp. booleovské). Necht' L je simulační jazyk

a C jeho sémantická třída. Když C obsahuje pouze A-systémy, řekneme, že L je jazykem základního typu A. Když C obsahuje pouze T-systémy, řekneme, že L je jazykem základního typu T. V ostatních případech řekneme, že L je jazykem základního typu AT. Sémantická třída jazyka základního typu AT nemusí být omezena na AT-systémy. Existují sice jazyky základního typu AT, v nichž lze popsat pouze systémy obsahující aspoň jednu aktivitu a aspoň jednu transakci, avšak obecně platí, že sémantická třída jazyka základního typu AT může obsahovat i systémy typu A (což jsou v mnoha případech degenerované případy systému typu AT) a méně častěji i systémy typu T. Více o těchto jazycích naleznete v [3].

5.2 Jazyky pro objektově orientované programování

Objektově orientované programování (OOP) je proces tvorby počítačových programů s pomocí reprezentace znalostí ve formě tříd, to jest ve formě abstraktních entit s mnoha důležitými vlastnostmi. [3]

Dnes nejčastěji používané prostředky pro OOP jsou C++, JAVA nebo novější verze Pascalu. Tyto prostředky vycházejí z více či méně radikálního pojetí OOP, dle něhož jediná možnost dění v počítači spočívá ve vysílání zpráv a v reakcích adresátu na ně. Kvaziparalelní systémy se v těchto jazycích nepřipouštějí. Při tomto přístupu se tvorba simulačních prostředků omezuje na prostředky s elementárními transakcemi. Jinými slovy, při použití prostředků pro OOP, které jsou dnes rozšířeny, se sestavují konfigurace tříd, pomocí nichž jsou realizovány různé jazyky, v nichž existují jednak pasivní transakce (transakce odpovídající prvkům, které „pozorujeme“ na simulovaném systému, ale bez vlastních životních pravidel), jednak elementární transakce, které nesou útržky děje v systému. Pasivní transakce jsou instance tříd, zatímco metody, které tyto transakce provádějí (na podněty odjinud), odpovídají elementárním transakcím. [3]

5.3 Použití software pro simulaci

Poslední možností při vytváření modelů pro simulaci je použití již hotového softwarového produktu. Tato metoda je dnes hojně využívána, protože není tak náročná jako vytváření simulačního jádra například pomocí OOP. K této metodě jsem se uchýlil i já, jelikož jeden takovýto software mám k dispozici a pracoval jsem s ním při studiu. V této kapitole se blíže s tímto programem seznámíme, ale ještě předtím ukážu, že existuje spousta alternativních programů a každý je použitelný pro jiné případy.

5.3.1 Matlab a Simulink

Matlab je integrované prostředí pro technické výpočty, modelování a simulace, měření a testování, řídicí techniku, zpracování signálů, komunikaci, zpracování obrazu a videa, vizualizaci dat atd. Matlab rovněž disponuje specializovanými nástroji pro finanční analýzy, modelování, simulace a optimalizace. [2]

Simulink je nadstavba Matlabu pro simulaci a modelování dynamických systémů, který využívá algoritmy Matlabu pro numerické řešení nelineárních diferenciálních rovnic. Poskytuje uživateli možnost rychle a snadno vytvářet modely dynamických soustav ve formě blokových schémat a rovnic. [2]

5.3.1.1 Matlab

Jelikož jsem hlavní část práce vytvářel v Simulinku, zmíním se o Matlabu jen okrajově. Více informací o tomto programu je možné získat například v [2].

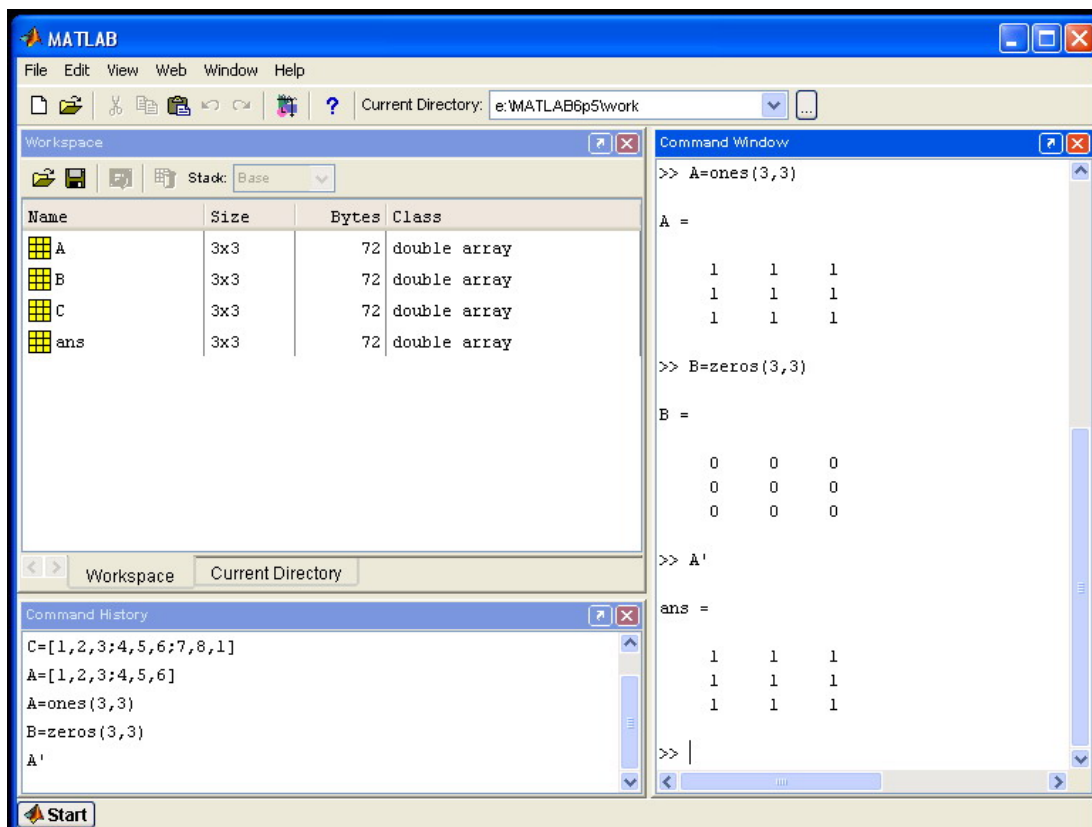
Matlab je výkonné, interaktivní prostředí pro vědecké a inženýrské výpočty a vizualizaci dat. Matlab integruje numerickou analýzu, maticové výpočty, zpracování signálů a grafiku do uživatelsky příjemného prostředí, ve kterém se problémy a řešení zapisují stejně jako v matematice, tedy bez tradičního programování. Matlab je interaktivní systém, jehož základním datovým prvkem je matice, u které se nezadáva rozměr. To nám umožňuje řešit mnoho numerických problémů podstatně rychleji než při použití klasických programovacích jazyků

(Fortran, Basic nebo C). Snad nejocetovanější vlastností Matlabu je jeho snadná rozšiřitelnost, která nám umožňuje doplňovat systém o napsané funkce (M-soubory) i celé aplikace. K Matlabu si můžeme navíc pořídit celou řadu specificky zaměřených nadstaveb, tzv. toolboxů, což je kolekce M-souborů, která je určena pro řešení jistých tříd problémů. V současnosti jsou k dispozici nadstavby pro následující okruhy problémů [11]:

- zpracování signálů,
- zpracování obrazů,
- teorie řízení,
- identifikace systémů,
- optimalizace,
- neuronové sítě,
- spliny,
- statistika,
- symbolická matematika.

Grafické prostředí

Matlab zobrazuje veškeré grafické výstupy v jednotlivých grafických oknech. Grafická okna mají své vlastnosti, které můžete modifikovat příkazy a funkcemi Matlabu. Počet grafických oken není nijak omezen.



Obr. 1 - Hlavní okno Matlabu

Matice

Matlab pracuje v podstatě s jedním typem dat a tím je matice (reálná nebo komplexní); tj. všechny proměnné jsou matice. V některých situacích jsou matice řádu jedna interpretovány jako skaláry a matice s jedním řádkem nebo sloupcem jako vektory. [11]

Příkazy a proměnné

Matlab je výrazový jazyk; výrazy, které napíšete, jsou interpretovány a vyhodnoceny. Příkazy Matlabu jsou obvykle ve tvaru *proměnná = výraz* nebo jenom *výraz*. Výraz je obvykle složen z operátorů, funkcí a jmen proměnných. Výsledkem vyhodnocení výrazu je matice, která se zobrazí na obrazovce a je přiřazena do proměnné. Pokud název proměnné a rovnítko chybí, je automaticky vytvořena proměnná *ans* (answer), do které je výsledek uložen. [11]

M-soubory (M-files)

Matlab se obvykle používá v příkazovém módu; když zadáte jednořádkový příkaz, Matlab ho okamžitě provede a zobrazí výsledky. Kromě toho může Matlab

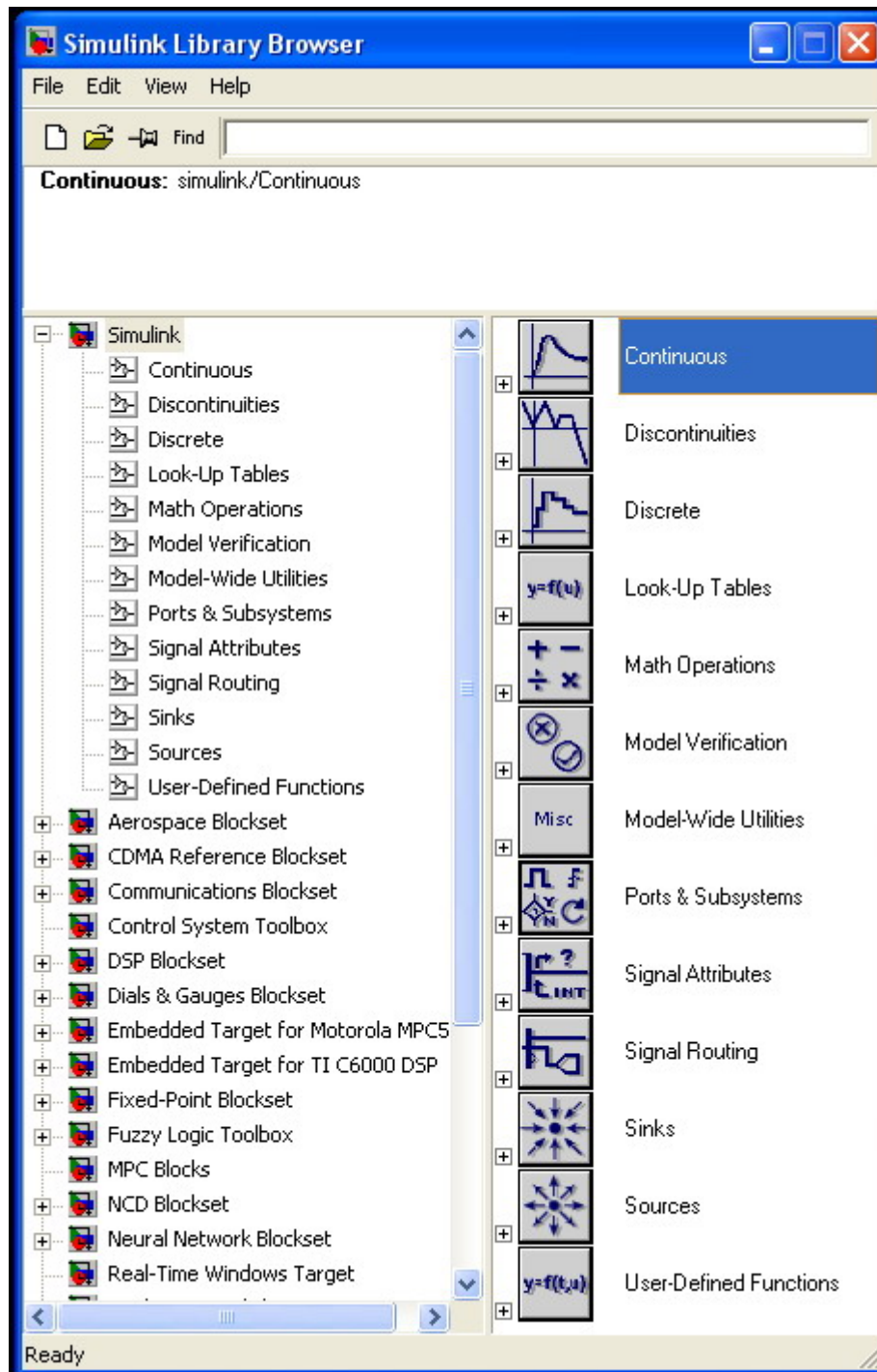
také spouštět posloupnosti příkazů, které jsou uloženy v souborech. Soubory, které obsahují příkazy Matlabu, se nazývají M-soubory, neboť mají příponu '.m'. Např. soubor s názvem **bessel.m** obsahuje příkazy Matlabu, které vypočtou hodnotu Besselovy funkce. M-soubory obsahují posloupnost normálních příkazů Matlabu, které se mohou dále odkazovat na jiné M-soubory. M-soubor může volat rekurzivně sám sebe. M-soubor můžeme vytvořit libovolným textovým editorem. Rozlišujeme dva typy M-souborů: skripty a funkce. Skripty, neboli skriptové soubory, automatizují dlouhé posloupnosti příkazů. Funkce, neboli funkční soubory, poskytují Matlabu rozšiřitelnost. Funkce nám umožňují přidávat nové funkce k funkcím existujícím. Za svoji oblíbenost vděčí Matlab v mnohém právě své schopnosti vytvářet nové funkce, které řeší uživatelem specifikované problémy. Skripty i funkce jsou obyčejné textové (ASCII) soubory. [11]

Tím bych výčet některých vlastností Matlabu ukončil a zaměřím se spíše na Simulink.

5.3.1.2 Simulink

Simulink je nadstavba Matlabu. Jedná se v principu o grafický "analogový" počítač. Simulink ke svému výpočtu využívá procedury Matlabu, které jsou dostatečně robustní a efektivní. Simulink je tedy určen na řešení - simulaci chování dynamického systému, pokud známe jeho matematický popis. [12]

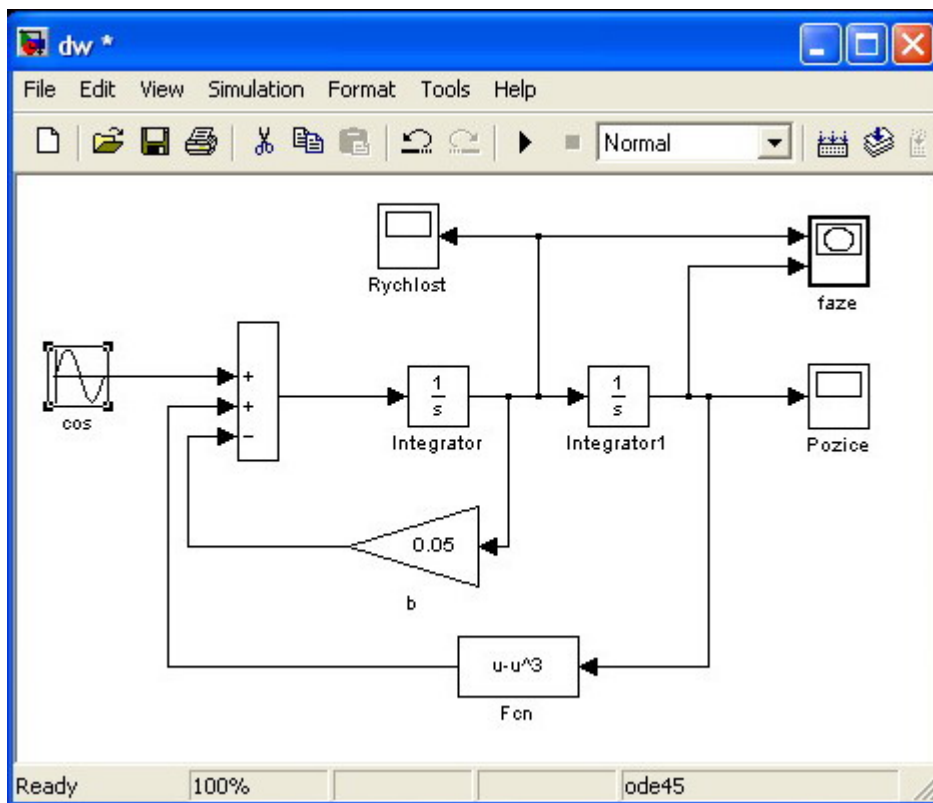
Vlastní práce se Simulinkem je jednoduchá. Simulink se spouští z Matlabu příkazem *simulink*. Poté se nám na obrazovku otevře okno se všemi nainstalovanými knihovnamí simulinku. Jejich počet je vždy závislý na nainstalovaných toolboxech Matlabu. Z těchto knihoven můžeme pomocí myši přetahovat jednotlivé bloky, které jsou určeny matematickým popisem modelu, do vlastního okna modelu. Jednotlivé bloky lze propojit pomocí myši, a pokud je vše v pořádku, lze spustit simulaci. Před spuštěním simulace je dobré vždy přezkontrolovat její parametry jako např. koncový čas, algoritmus řešení, krok a další. [12]



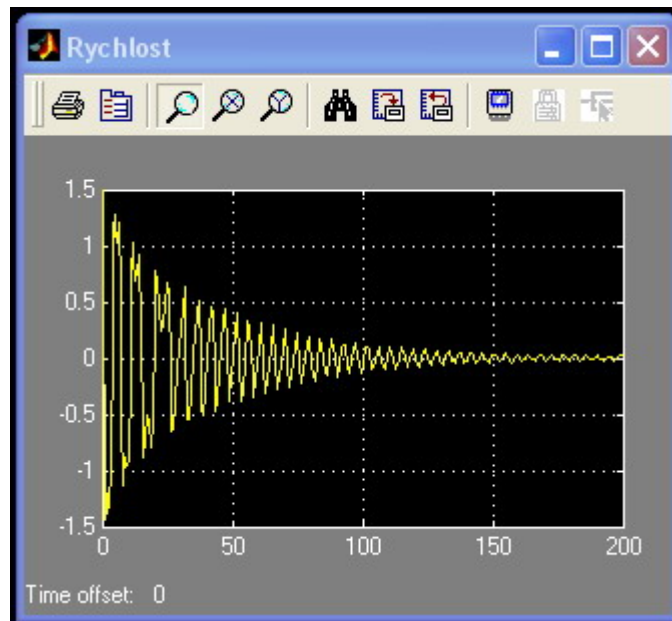
Obr. 2 - Simulink - okno s knihovnamí

Parametry jednotlivých bloků (hlavně počátečních hodnot a zesílení) lze nastavit přímo v simulinkovém modelu nebo pomocí Matlabu (nejčastěji vhodným skriptem). Tento skript je nutné spustit před vlastním spuštěním výpočtu v Simulinku. U složitějších modelů se plně osvědčí nastavování skriptem. Také výsledky řešení (simulace) je vždy lepší uložit do souboru s příponou mat

a zpracovat poté pomocí skriptu. Zobrazovací elementy Simulinku lze využívat pouze k prvotní vizuální kontrole. U složitých systémů je možné celý model koncipovat jako jednotlivé malé podsystémy, které jsou mezi sebou propojeny. Lze tak snáze celý model odlaďovat a také snáze se v něm orientovat. Na následujících obrázcích vidíte okno s vyřešeným příkladem a výsledným grafem. [12]



Obr. 3 - Simulink - okno s příkladem



Obr. 4 - Simulink - graf

Mezi základní knihovny Simulinku (viz **Obr. 2**) patří knihovna spojitých (continuous), diskrétních (discrete), zdrojových (sources), zobrazovacích (sinks), matematických (math), nelineárních (nonlinear) signálových a subsystémových (signals & subsystems) bloků. Tyto jednotlivé bloky lze táhnutím myši přenést do okna modelu a poklepáním na ně nastavit jejich parametry. Spojování bloků se provádí také myší a to táhnutím myši se stisknutým levým tlačítkem od místa výstupu či vstupu. Pokud potřebujeme signál rozvétvit, je vždy lepší táhnout od vstupu následujícího bloku na již propojenou linii, kde se signál automaticky naváže. Pokud chceme vložit k modelu doprovodný text, klikneme dvakrát na místo, kde má text být a zobrazí se nám malé editační okno. Stejně lze upravovat i popisy u jednotlivých bloků. Pozor musíme v tomto případě dát na to, aby každý blok měl jiný název. [12]

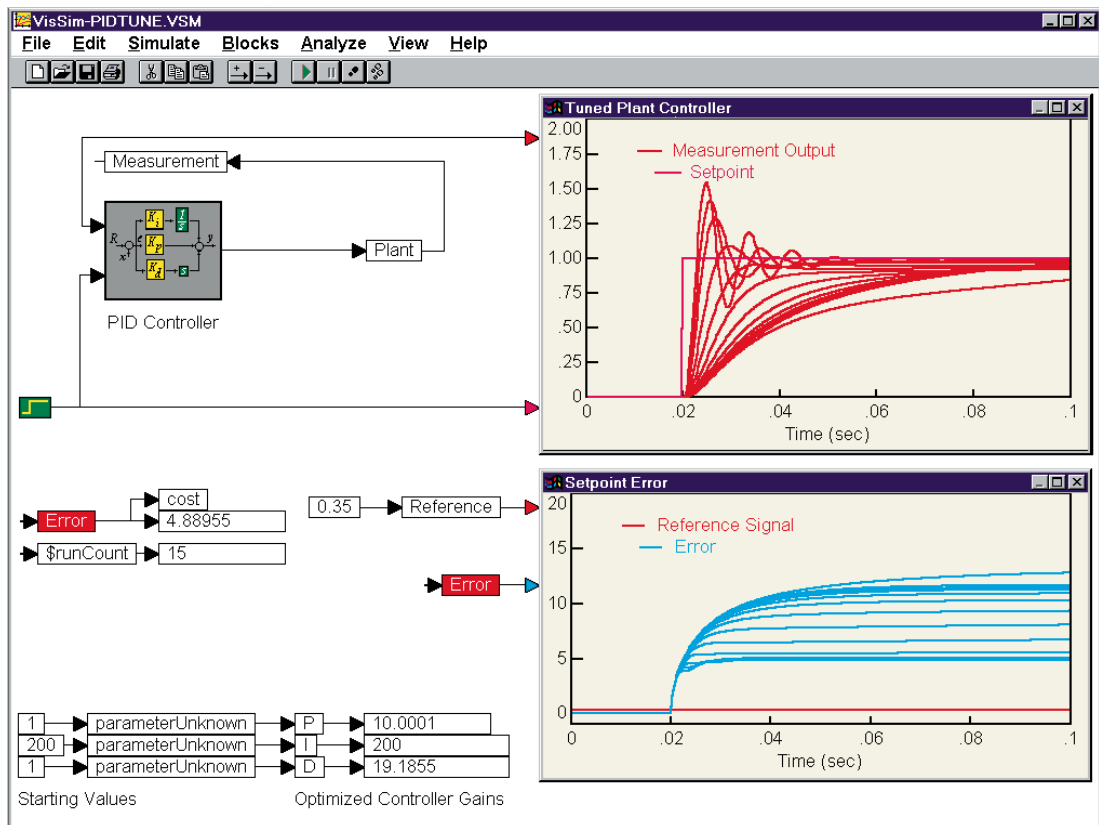
5.3.2 Jiný software pro spojitou simulaci

V této podkapitole poukážu na to, že Simulink není jediná volba při výběru software pro modelování dynamických systémů. Proto jsem vybral dva programy, které mají podobný účel jako Simulink a krátce je popíšu.

5.3.2.1 VisSim

S VisSim jsem neměl možnost se blíže seznámit, proto uvedu jen oficiální firemní popis tohoto produktu [6]:

VisSim je moderní vizualizační prostředí pro komplexní dynamický simulační systém, nyní v nejnovější verzi 6.0. Program kombinuje jasnost, jednoznačnost a názornost grafiky blokových diagramů s mohutným výpočetním jádrem. Přináší rychlé a přesné řešení pro lineární, nelineární, kontinuální, diskrétní funkce, multiplikační a smíšené úlohy. Kromě toho integrovaná vývojová platforma umožňuje volný přechod mezi tvorbou modelu, simulací, optimalizací a validizací. Tak jako Mathcad VisSim eliminuje tradiční programování, takže není potřeba ztrácet čas dlouhou přípravou a studiem před vlastním použitím programu. Pro specializované technické úlohy nabízí VisSim rozsáhlý set firemních produktů zaměřených na linearizaci a frekvenční analýzu, modelování komunikačních systémů, simulaci neuronových sítí a další. Produkty VisSim/Comm tvoří skupinu zaměřenou na rychlé, efektivní a jednoduché modelování komunikačních systémů. Zabudované nástroje umožňují integraci a spolupráci s programem Mathcad. Program je nabízen jak pro komerční organizace, tak se zvýhodněním pro akademické nekomerční zájemce. Je dostupný v jednorázové verzi nebo floating síťové verzi. Na obrázku 5 můžete vidět jak vypadá hlavní okno VisSim.

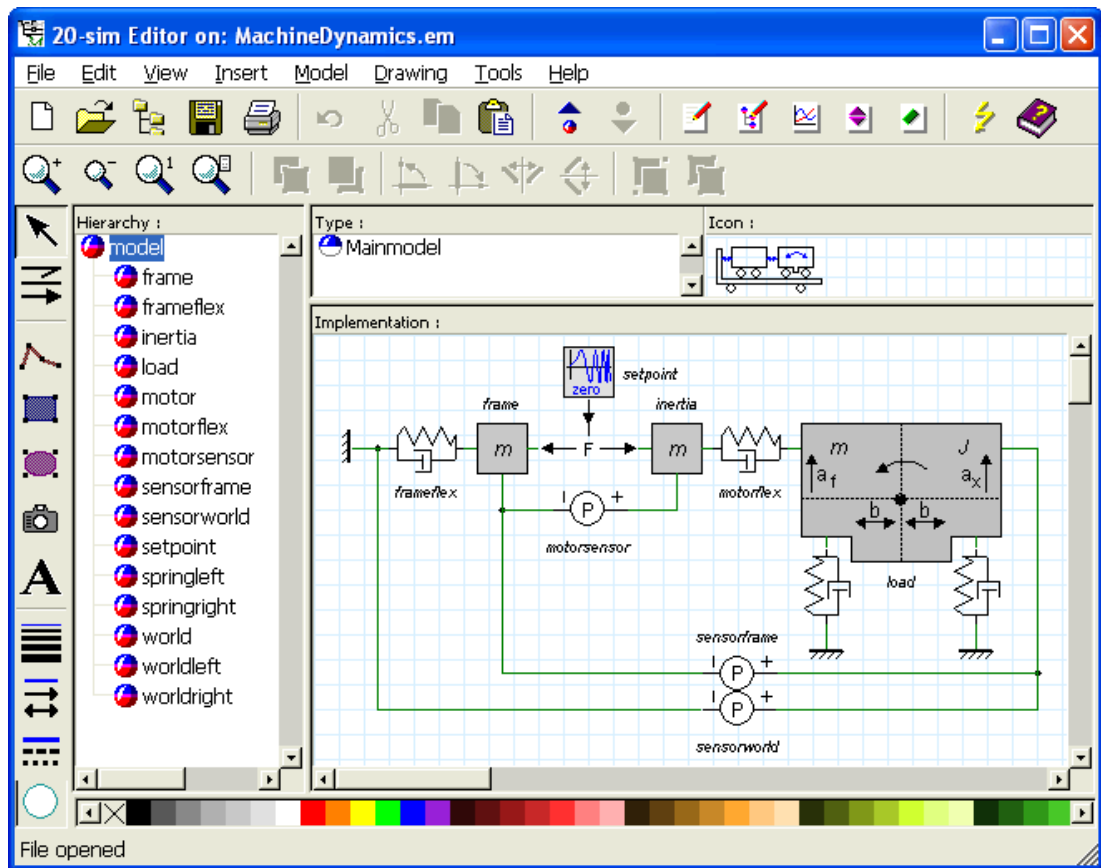


Obr. 5 - VisSim

5.3.2.2 20-Sim

20-sim je program pro modelování a simulaci, který běží pod Microsoft Windows. Je s ním možné modelovat chování dynamických systémů, například elektrických, mechanických, hydraulických nebo jakoukoliv jejich kombinaci.

20-sim plně podporuje grafické modelování, umožňuje navrhovat a analyzovat dynamické systémy intuitivní a uživatelsky příjemnou cestou. Podporuje také používání komponent. To umožňuje zadávat modely jako jakýsi inženýrský náčrtek výběrem komponent z knihovny a jejich následné spojení. [5]



Obr. 6 - 20-sim

Samozřejmě, že existuje ještě celá řada dalších produktů pro modelování spojitých systémů. Při jejich výběru musíme brát v potaz mnoho hledisek. Mělo by nás především zajímat, zda daný software zvládne svými prostředky vyřešit úlohu, kterou požadujeme, zda je finančně dostupný, zda je ještě v budoucnu šance na jeho využití.

6 Fyzikální analogie

V této kapitole se dostáváme k hlavnímu tématu práce, kterým je fyzikální analogie.

Pokud chceme popsat nějaký dynamický systém, tak za použití zákonů fyziky tento model popíšeme diferenciální rovnicí. Daná diferenciální rovnice představuje matematický model, který určuje průběh sledovaného příznaku. Samotný matematický model však nedává řešiteli informaci v podobě, kterou potřebuje pro vyhodnocení zkoumaného děje. Tu získáme teprve řešením matematického modelu. Řešení zpravidla provádíme s využitím numerických metod řešení diferenciálních rovnic na číslicových počítačích ve vhodných simulačních programech. V simulačním programu vytvoříme simulační model, který představuje odvozený matematický model formulovaný pomocí prostředků použitého simulačního programu. Může to být grafické schéma znázorňující strukturu simulačního modelu u blokově orientovaných simulačních programů s grafickým vstupem, mohou to být příkazové řádky simulačního programu popisujícího odvozený matematický model.

Způsoby modelování se liší podle kritéria přiřazení modelu k originálu. Můžeme vycházet z **podobnosti a analogie**. [1]

Podobnost představuje jednoznačné vzájemné přiřazení mezi různými systémy v jejich struktuře, vlastnostech a chování. [1]

Fyzikální podobnost vyjadřuje podobnost mezi systémy a procesy stejné podstaty a vedle geometrické podobnosti zahrnuje podobnost parametrů a stavových veličin. [1]

Matematická podobnost vyjadřuje podobnost mezi systémy a procesy majícími stejný matematický popis. [1]

Analogie pak znamená matematickou podobnost fyzikálně odlišných systémů a procesů. [1]

Fyzikální analogie tedy představuje rozšíření pojmu fyzikální podobnost na fyzikálně různé jevy, které mají různý matematický popis. Dnes je fyzikální analogie hojně používaná například v termodynamice, pružnosti, hydrodynamice, aplikované matematice nebo elektrotechnice. Nejčastěji se však používají analogy elektrické. Ty pracují na principu matematické podobnosti mezi fyzikálním polem nebo obvodem a jejich elektrickým modelem. Kromě elektrických analogií se však v experimentální technice někdy používá hydraulické, pneumatické, chemické, hydrodynamické, tepelné, mechanické, optické aj. Některé z nich se již dnes nepoužívají a ztratily na významu, jiné však zůstaly v důsledku svého zdokonalování rovnocenným prostředkem ostatním progresivním metodám (např. numerickým). V některých případech řešení složitých úloh mohou mít i výjimečné postavení, zejména díky vlastnostem jako je jednoduchost, nenáročnost a bezprostřednost.

Pokud volíme analogii, tak kromě výše uvedených vlastností rozhoduje především přesnost, dostupnost, charakter úlohy a měřitelnost veličin na modelu.

V následující tabulce naleznete příklady různých fyzikálně analogických ustálených polí s vnitřními zdroji popisovaných Poissonovskou rovnicí.

Tabulka 1 – Analogie ustálených fyzikálních polí. Zdroj: [7]

	Soustava	fyzikální pole	Základní rovnice	Popis veličin	Schéma
1	Mechanická	Mechanická deformace membrány zatížené přetlakem	$\nabla^2 z(x, y) = -\frac{p}{S}$	z – průhyb membrány, p – vnitřní přetlak, $S(N.m^{-1})$ - napjatost membrány, x, y - souřadnice	
2	Mechanická	Smyková napětí ve zkrucovaném hřídeli	$\nabla^2 F(x, y, z) = -2G\vartheta$ $\tau_{yx} = \pm \frac{\partial F}{\partial z}, \tau_{zx} = \mp \frac{\partial F}{\partial z}$	F – funkce napětí, G – modul pružnosti ve smyku, ϑ - zkrut, τ - smyková napětí	
3	Hydraulická	Proudění vazké nestlačitelné tekutiny potrubím	$\nabla^2 w(x, y, z) = -\frac{1}{\eta} \frac{\partial p}{\partial x}$	w – rychlost proudění, η - dynamická vazkost, $\frac{\partial p}{\partial x}$ - tlakový spád ve směru proudění	
4	Hydraulická	Vírové proudění ideální nestlačitelné tekutiny	$\nabla^2 \Phi(x, y, z) = -2\omega$ $w = -grad\Phi$	Φ - rychlostní potenciál, ω - úhlová rychlost, w – místní rychlost	

5	Hydraulická	Proudění ideální nestlačitelné tekutiny za zřídly či propady	$\nabla^2 \Phi(x, y, z) = \pm \frac{Q_v}{L^3}$ $w = -grad \Phi$	Φ – rychlostní potenciál, Q_v – vydatnost zřídla (+) či pohltivost propadu (-), L – délkový rozměr	
6	Hydraulická	Filtrace tekutin při působení zdrojů či propadů	$\nabla^2 h(x, y, z) = \frac{v_0}{kh_0}$ $\Phi = -kh, w = -grad \Phi = k grad h$ $h = h_0 + z$	h – výška spodní vody, h_0 – tlaková výška, z – geodetická výška, v_0 – infiltrace, k – součinitel filtrace	
	Tepelná	Hmotnost pole s vnitřními zdroji či propady	$\nabla^2 c(x, y, z) = -j''' D^{-1}$ $j'' = -D grad c$	c – molární koncentrace, j'' – hustota hmotnostního toku, j''' – objemová hustota hm. toku, D – difúzní součinitel	
8	Tepelná	Teplotní pole s vnitřními zdroji či propady	$\nabla^2 T(x, y, z) = \mp \frac{q'''}{\lambda}$ $q'' = -\lambda grad T$	T – teplota, q'' – hustota tepelného toku, q''' – objemová hustota tep. toku, λ – tepelná vodivost	
9	Elektrická	Neutronové pole	$\nabla^2 \rho_{1,2}(x, y, z) = \pm k_{1,2}$	ρ – monokinetický tok neutronů, k_1 – konstanta aktivní zóny 1, k_2 – konstanta odražeče 2	

10	Elektrická	Elektrické pole s vnitřními zdroji či propady	$\nabla^2 V(x, y, z) = -\rho J''''$ $J'' = -\gamma \text{grad } V$	V – elektrický potenciál, ρ – měrný odpor, J'' – proudová hustota, J'''' – objemová proudová hustota	
11	Elektrická	Elektrostatické pole	$\nabla^2 V(x, y, z) = -\frac{\rho}{\epsilon}$ $D = -\epsilon \text{grad } V$	V – elektrický potenciál, ρ – objemová hustota náboje, ϵ – permitivita, D – elektrická indukce	
12	Magnetická	Magnetické pole v rovině x, y	$\nabla^2 A_z(x, y) = -\mu J$ $ B = \text{grad } A_z $	A_z – magnetický potenciál, J – proudová hustota, μ – permeabilita, B – magnetická indukce	

6.1 Modelování spojitých systému pomocí fyzikální analogie

Modelování spojitých systémů s využitím fyzikální analogie vychází z izomorfie různých fyzikálních soustav. Patří mezi ně například elektrická, mechanická, pneumatická nebo hydraulická fyzikální soustava. V každé z nich lze definovat dvojici komplementárních veličin extenzivní (průtokové) $i(t)$ a intenzivní (rozdílové) $e(t)$, jejichž součin představuje okamžitý výkon

$$P(t) = i(t)e(t)$$

Celkovou energetickou bilanci systému v průběhu časového intervalu $\langle t_1, t_2 \rangle$ stanovíme podle vztahu

$$E(t_1, t_2) = \int_{t_1}^{t_2} P(t) dt = \int_{t_1}^{t_2} i(t)e(t) dt$$

Kromě výše uvedených veličin lze fyzikální vlastnosti soustav vyjádřit pomocí definice základních impedancí typu odpor, kapacita nebo indukčnost, které určují 3 základní typy závislosti extenzivní a intenzivní veličiny. [1]

Odpor představuje prvek, na kterém je závislost mezi veličinami $i(t)$ a $e(t)$ obecně určena vztahem

$$e(t) = Ri(t)$$

Obecná **kapacita** C realizuje závislost veličin $i(t)$ a $e(t)$ určenou vztahem

$$i(t) = C \frac{de(t)}{dt}$$

Indukčnost L představuje transformaci veličin $i(t)$ a $e(t)$ podle vztahu

$$i(t) = L \frac{di(t)}{dt}$$

Vzájemným propojením impedancí definovaných typů a pomocí řízených zdrojů intenzivní a extenzivní veličiny lze sestavit odporové schéma různých prvků v různých fyzikálních soustavách. Pro odvození jejich matematického popisu lze použít zobecněné Kirchhoffovy zákony. [1]

- Součet extenzivních veličin v daném uzlu je roven nule $\sum_{j=1}^k i_j(t) = 0$.
- Součet všech intenzivních veličin jednotlivých impedancí a zdrojů v uzavřené smyčce je roven nule $\sum_{j=1}^m e_j(t) = 0$.

Použití uvedené metodiky popisu spojitých systémů pomocí zavedených typů impedancí je výhodné, zejména pokud je použitý simulační program vybaven odpovídajícími makromodely, které lze propojit do různých topologických struktur. Pomocí uvedených makromodelů tak lze zadat popisy různých soustav. [1]

V následující tabulce jsou obecným veličinám přiřazeny konkrétní fyzikální veličiny vybraných systémů a jejich vlastnostem přiřazeny jejich zavedené názvy.

Tabulka 2 – Přiřazení fyzikálních veličin obecným veličinám. Zdroj: [1]

Soustava	Fyzikální veličina		Impedance		
	extenzivní	intenzivní	Odpor	Kapacita	indukčnost
	$i(t)$	$e(t)$	$\frac{e(t)}{i(t)}$	$\frac{i(t)}{\frac{de(t)}{dt}}$	$\frac{e(t)}{\frac{di(t)}{dt}}$
Elektrická	Elektrický proud $i(t)$ [A]	Elektrické napětí $u(t)$ [V]	Elektrický odpor R [Ω]	Elektrická kapacita C [F]	Elektrická indukčnost L [H]
Mechanická	Síla $F(t)$ [N]	Rychlost $v(t)$ [m/s]	Translační vodivost $G = \frac{1}{b}$ [kg.s ⁻¹]	Hmotnost m [kg]	Mechanická poddajnost $\delta = \frac{1}{k}$ [m/N]
Hydraulická	Průtok $Q(t)$ [m ³ /s]	Tlak $p(t)$ [Pa]	Hydraulický odpor R [Pa.s.m ⁻³]	Pneumatická kapacita $C = \frac{1}{D}$ [m ³ .Pa ⁻¹]	Hydraulická indukčnost H [m ³ .Pa ⁻¹ .s ⁻¹]
Pneumatická	Průtok $Q(t)$ [m ³ /s]	Tlak $p(t)$ [Pa]	Pneumatický odpor R [Pa.s.m ⁻³]	Pneumatická kapacita $C = \frac{1}{D}$ [m ³ .Pa ⁻¹]	Pneumatická indukčnost H [m ³ .Pa ⁻¹ .s ⁻¹]
Tepelná	Tepelný tok $\Phi(t)$ [W]	Absolutní teplota $T(t)$ [K]	Tepelný odpor R [K.s.J ⁻¹]	Tepelná kapacita C [J.K ⁻¹]	-

U tepelných systémů je v praxi zažitá analogie k elektrickým systémům, kde napětí u je analogické absolutní teplotě T (intenzivní veličina $e(t)$) a proud i je analogický tepelnému toku Φ (extenzivní veličina $i(t)$). Součin uvedených proměnných však nemá fyzikální rozměr výkonu, jelikož tepelný tok Φ má již rozměr výkonu. Z uvedených důvodů se někdy definují intenzivní a extenzivní veličiny jako absolutní teplota T a tok entropie $\frac{dS}{dt}$. [1]

6.2 Elektrické analogy fyzikálních obvodů

V této podkapitole bych se rád zmínil o elektrických obvodech a jejich analogiích. Elektrické obvody složené ze zdrojů, odporů a indukčností dodnes neztratily na významu. Možnost experimentálního zkoumání dynamických soustav na ekvivalentním elektrickém obvodu je stále z hlediska časové a ekonomické náročnosti i přesnosti řešení obvykle výhodnější než vyšetřování vlastních soustav.

6.2.1 Elektrické analogy dynamických obvodů

Elektrické analogy statických obvodů si dovoluji vynechat, protože ty nejsou tématem této práce a spíše se zaměřím na obvody dynamické.

Ty pracují na principu analogie mezi dynamickým mechanickým a elektrickým obvodem. Obvody jsou popsány analogickými diferenciálními rovnicemi. Matematická podobnost je nutnou, nikoli postačující podmínkou. [7]

Mezi základní druhy elektromechanických analogií patří například mechanická kmitavá soustava s jedním stupněm volnosti sestávající z hmotnosti m pružiny o lineární poddajnosti c a tlumiče s konstantou tlumení k . Na tuto soustavu působí dynamická síla $F(\tau)$. Základní rovnice popisující soustavu má tvar

$$m\ddot{x} + k\dot{x} + cx = F(\tau)$$

Pro analogický rezonanční elektrický obvod se sériově řazenými prvky platí rovnice

$$C\ddot{U} + G\dot{U} + L^{-1}U = \dot{I}(\tau)$$

V případě paralelně řazených prvků v elektrickém obvodu dostaneme

$$L\ddot{I} + R\dot{I} + C^{-1}I = \dot{U}(\tau)$$

Zpracování tohoto příkladu a ukázky analogií naleznete v následující kapitole.

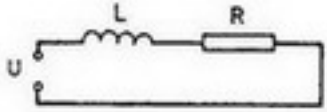
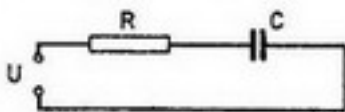
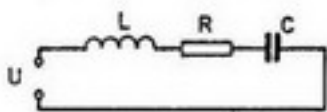
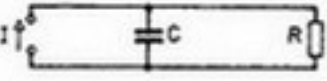
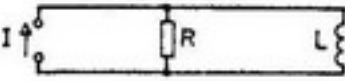
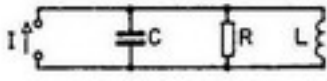


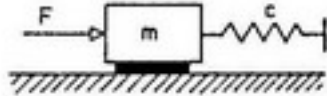
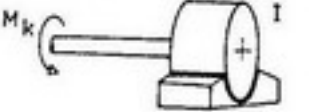
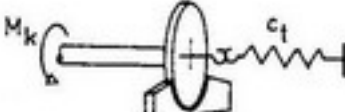
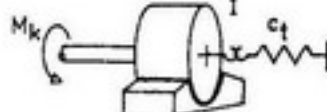
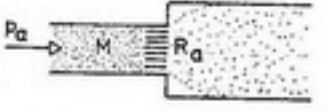
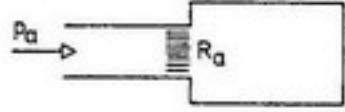

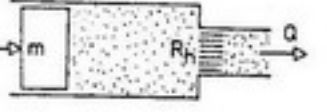
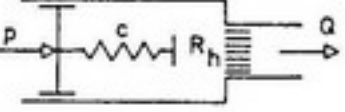

Z pozorování odpovídajících si rovnic pro tři uvedené obvody vyplývají dva základní druhy analogií. Jedná se o napěťovou a proudovou analogii. U napěťové analogie odpovídá rychlosti elektrický proud I a síle F elektrické napětí U . U analogie proudové je tomu naopak.

Elektromechanická analogie předpokládá zidealizované rovinné soustavy s jedním stupněm volnosti, řetězce, v nichž každý prvek je představován vždy jedním parametrem. Uvedená zjednodušení obvykle nevedou k podstatnějšímu ovlivnění požadované přesnosti.

Z uvedených druhů analogií je více rozšířena napěťová, při níž působícím silám nebo momentům odpovídají na modelu elektrická napětí, jež jsou snáze a přesněji měřitelná. Na druhé straně předností proudové analogie je jednoduchost a názornost při sestavování modelového schématu. [7]

V následující tabulce je uveden přehled analogických dynamických obvodů, a to tlumeného bez kapacitance nebo indukance a kmitavého obvodu s tlumením. Porovnávají se schémata a základní rovnice obvodu elektrického, mechanického, přímočarého, mechanického rotačního, akustického a hydraulického.

Tabulka 3 – Analogické dynamické obvody. Zdroj: [7].

		tlumený obvod bez kapacity	tlumený obvod bez indukčnosti	kmitavý obvod s tlumením	veličiny	
		$Y = \alpha \dot{X} + \beta X$	$Y = \beta X + \gamma^{-1} \int X d\tau$	$Y = \alpha \dot{X} + \beta X + \gamma^{-1} X d\tau$		
Elektrický obvod	napětový				X – elektrický proud I , Y – elektrické napětí U	α - indukčnost L , β - elektrický odpor R , γ - elektrická kapacita C
	proudový				X – elektrické napětí U , Y – elektrický proud I	α - elektrická kapacita C , β - elektrická vodivost $G = R^{-1}$, γ - indukčnost L
Mechanický obvod	posuvný				X – rychlost $v = \dot{X}$, Y – síla F	α - hmotnost m , β - mechanický odpor R_m , Konstanta tlumení k , γ - lineární poddajnost c
	rotační				X – úhlová rychlost $\omega_r = \dot{\phi}$, Y – Kroučící moment M_k	α - moment setrvačnosti I , β - mechanický odpor R_r , γ - korzní poddajnost c_t
Akustický obvod					X – objemová rychlost q , Y – akustický tlak p_a	α - hmotnost M , β - akustický odpor R_a , γ - akustická poddajnost c_a
Hydraulický obvod					X – rychlost v , Y – Síla F	α - hmotnost m , β - hydraulický odpor R_h , γ - lineární poddajnost c

Obdobně jako mechanické modely lze modelovat i hydraulické a pneumatické obvody na elektrickém analogu. Ke každému hydraulickému nebo pneumatickému prvku lze najít analogický prvek elektrického obvodu. Podstatný rozdíl mezi hydraulickým a pneumatickým obvodem může však být vyvolán stlačitelností pracovní látky.

7 Tvorba modelů a ukázky analogií

V této kapitole se dozvíme, jak se vytvářejí matematické modely pomocí fyzikálních zákonů a konečně zde také nalezneme ukázky fyzikálních analogií na modelech vytvořených v programu Simulink.

Protože jsem se rozhodl, že analogie budu ukazovat na mechanické a elektrické soustavě, zaměřím se právě na tyto dvě i při popisu tvorby matematických modelů.

7.1 Tvorba matematických modelů mechanických systémů

Pokud chceme sestavit matematický model mechanického systému, který tvoří hmotné objekty a vazby mezi nimi, musíme stanovit vztahy mezi jejich pohybem a působícími silami pomocí zákonů dynamiky. Musíme však brát v potaz, že určení jednotlivých parametrů modelu nemusí být až tak jednoduché, zpravidla vychází z dokumentace modelovaného objektu, někdy musí být stanoveny experimentálně. S cílem analyzovat dynamické vlastnosti strojů a zařízení bude nejčastěji vytvářený matematický model představovat popis soustavy dokonale tuhých těles, tzn. soustavy vzájemně vázaných hmotných bodů. Tato cesta tvorby matematického modelu mechanického systému vychází z předpokladu soustředění hmotností a působících sil do jediného bodu, tzv. **hmotného bodu**. Dále často předpokládáme, že těleso je dokonale tuhé, chápané jako soustava hmotných bodů, které nemění svoji konfiguraci. Skutečné vlastnosti těles – jejich poddajnost – rovněž popisujeme jako soustředěnou vlastnost a modelujeme pomocí poddajnosti, resp. tuhosti, tlumení apod. Matematické modely, které jsou vytvářeny za předpokladu a současně i za velmi důležitého zjednodušení soustředění vlastností a silového působení do hmotného bodu, se nazývají **modely se soustředěnými parametry**. [1]

Mohou ale nastat i případy, kdy je nutno brát v potaz například hmotnost, tuhost spojitě rozloženou, pak hovoříme o systému se spojitě rozloženými parametry, které se také označují jako kontinuum. Matematické modely, které popisují tyto vlastnosti, jsou vyjádřeny pomocí parciálních diferenciálních rovnic.

Pokud tedy chceme sestavit matematický model mechanického systému, je nutné vycházet ze tří Newtonových zákonů. [1]

První Newtonův zákon

Hmotný bod se pohybuje rovnoměrně přímočaře nebo je v klidu, není-li nucen vnější silou tento pohybový stav měnit.

Druhý Newtonův zákon

Součin hmotnosti hmotného bodu a jeho zrychlení je roven působící síle

$$m \cdot a = F$$

Třetí Newtonův zákon

Vzájemný účinek dvou těles je stejný, ale opačného smyslu.

Jen Newtonovy zákony ale na vytvoření matematického modelu nestačí. Musíme ho hledat také ve tvaru pohybové rovnice, která vyjadřuje vztah mezi změnou pohybu uvažovaného hmotného bodu a působícími silami. Sestavení pohybové rovnice lze provést dvěma možnými způsoby. Jedná se o **aplikaci druhého Newtonova zákona** nebo **pomocí d'Alembertova principu**. [1]

Pokud se rozhodneme pro první způsob, tedy **pomocí druhého Newtonova zákona**, sestavujeme pohybovou rovnici pro obecnou polohu tělesa, kdy uvažujeme všechny síly, které na vyšetřovaný bod působí. Těleso uvolníme a vyhodnotíme všechny působící síly, pokud je bod vázán kinetickými vazbami, musíme uvážit i účinky vazeb, tedy reakční síly. Reakční sílu spolu s působícími silami ve vektorovém součtu definují výslednou sílu

$$m \cdot a = \sum_i F_i$$

Pohybová rovnice je tedy obecně vektorová. Podle konkrétní úlohy ji rozepíšeme na příslušný počet rovnic, jejichž počet závisí na tom, zda jde o pohyb po přímce, v rovině nebo v prostoru. [1]

Při použití druhého způsobu, tedy d'Alembertova principu musíme zavést setrvačnou sílu

$$D = -m \cdot a$$

kteřou je nutné přiřadit k působícím silám. Tento součin zrychlení a hmotnosti nazval francouzský matematik a fyzik Jean Le Rond d'Alembert setrvační silou, neboť je podle definice orientována proti smyslu zrychlení a lze ji chápat jako odpor hmoty proti jejímu urychlování, jako sílu vyjadřující tendenci hmotného bodu setrvávat v rovnoměrném přímočarém pohybu nebo v klidu. Po zavedení této myšlené síly, která působí na hmotný bod vedle výsledné síly

$$F = \sum_i F_i$$

a je s ní ve stavu rovnováhy, můžeme tudíž pohybovou rovnici sestavit z podmínek rovnováhy, pomocí pravidel statiky

$$\sum_i F_i + D = 0$$

Je zajímavé, že pohybové rovnice sestavené pomocí Newtonova pohybového zákona nebo d'Alembertova principu jsou shodné, jen se při sestavování vychází z jiných principů. [1]

7.1.1 Popis prvků modelu mechanického systému

Nyní bych rád popsal několik prvků, z kterých se může skládat model mechanického systému. Mezi tyto vzájemně vázané stavební prvky patří například hmotný bod, pružina, tlumič, zdroj síly a rychlosti u soustav konajících přímočarý pohyb a kotouč, torzní pružina, torzní tlumič, zdroj momentu a úhlové rychlosti u soustav konajících rotační pohyb. Jelikož modely, které jsem vytvářel, představují systémy konající pohyb přímočarý, zaměřím se spíše na popis těchto prvků. [1]

Prvním z prvků, které popíši, je hmotný bod. Ten představuje soustředěnou hmotnost tělesa umístěnou do jeho těžiště a vyjadřuje setrvačné účinky tělesa.

Pro hmotný bod platí pohybová rovnice při působení vnější síly F .

$$m \cdot a = F,$$

nebo také

$$m \cdot \frac{dv}{dt} = F,$$

$$m.\ddot{x} = F$$

Kinetická energie pohybujícího se hmotného bodu rychlostí v je rovna

$$W_K = \frac{1}{2} m.v^2 .$$

Dalším prvkem je **pružina** nebo také **poddajnost**. Síla přenášená pružinou je úměrná jejímu stlačení (protažení).

$$F_K = k.x ,$$

kde k je tuhost pružiny. Převrácená hodnota tuhosti k se označuje poddajnost

$$\delta = \frac{1}{k} .$$

Pružina je schopna akumulovat potenciální energii

$$W_P = \frac{1}{2} k.x^2 .$$

Síla přenášená **tlumičem** je úměrná rychlosti

$$F_b = b.v ,$$

nebo také

$$F_b = b.\dot{x} .$$

Tlumič je prvek, na kterém se energie soustavy maří, přeměňuje se v tepelnou formu energie.

7.2 Tvorba matematických modelů elektrických systémů

V této podkapitole jsem se zabýval elektrickými systémy. Ty jsou složeny z jednotlivých prvků tak, aby jimi mohl procházet elektrický proud. Elektrickým obvodem potom nazveme elektrický systém, pokud lze elektrické děje popsat dostatečně pomocí elektrických proudů a napětí.

Při analýze elektrických systémů pracujeme s elektrickým obvodem, který je schématicky znázorněn pomocí základních ideálních aktivních a pasivních prvků.

Mezi **aktivní prvky**, které jsou zdroje elektrické energie, pak patří zdroje napětí a zdroje proudu.

Pasivní prvky jsou spotřebiče elektrické energie. Patří mezi ně tyto ideální prvky: rezistory realizující odpor R , kapacitory nebo také kondenzátory realizující kapacitu C a induktory (cívky) realizující indukčnost L .

Veličiny, které sledujeme, jsou pak **elektrické napětí** $u(t)$ a **elektrický proud** $i(t)$. Součin těchto veličin, tedy součin proudu protékajícího prvkem a úbytku napětí na daném prvku určuje okamžitý výkon daného prvku:

$$P(t) = u(t) \cdot i(t).$$

Výše uvedené prvky mají také své vlastnosti. Jsou určeny jejich charakteristikami popisujícími závislost jedné veličiny na druhé

$$\begin{aligned}u &= f(i), \\i &= g(u), \\F(i, u) &= 0.\end{aligned}$$

Co se týče ideálních prvků, tak ty mají pouze jednu vlastnost (kapacita, odpor, indukčnost) a jsou vázány pouze s jedním druhem energie. Systémy tak můžeme rozložit na zapojení ideálních prvků a vytvořit tak model se soustředěnými parametry. Přijetí této představy při vytváření modelu představuje jedno z prvních přijatých a velmi důležitých zjednodušení. Vedle toho se můžeme setkat se systémy, kde použití soustředěných parametrů bude nemožné nebo omezené.

7.2.1 Popis prvků modelu elektrického systému

Prvním prvkem, který zmíním je **ideální zdroj napětí**. Je to zdroj, který má na svých svorkách obecně časově proměnné napětí $u(t)$ nezávislé na odebrání proudu. Může dodávat neomezeně velký výkon, proud může dosáhnout nekonečně velké hodnoty. Ideální zdroje napětí lze spojovat do série a jejich napětí se přitom sčítají. [1]

Další prvek elektrických systémů je **ideální zdroj proudu**. Ten dodává do připojených spotřebičů proud $i(t)$ nezávislý na vlastnostech těchto spotřebičů. Může dodávat nekonečně velký výkon, napětí na svorkách může dosáhnout nekonečně velkých hodnot. Ideální zdroje proudu lze spojovat paralelně a jejich proudy se sčítají. [1]

Ideální rezistor je prvkem, na kterém se elektrická energie mění v teplo. Závislost mezi napětím a proudem je vyjádřena Ohmovým zákonem:

$$u(t) = R \cdot i(t),$$

kde R je činný odpor [Ω]. Okamžitý výkon je roven

$$P = u(t) \cdot i(t) = R i^2(t) = \frac{u^2(t)}{R}.$$

Prvek schopný akumulovat energii elektrického pole se nazývá **ideální kapacitor**. Proud protékající kapacitorem je s napětím na kapacitoru vázán vztahem

$$i = C \frac{du(t)}{dt},$$

kde C je kapacita [F]. Akumulovaná energie je rovna

$$W_c = \frac{1}{2} \cdot C \cdot u^2(t).$$

Posledním prvkem elektrických obvodů, který popíšu, je **ideální induktor**. Ten akumuluje pouze energii magnetického pole. V technické praxi je představován cívkou. Okamžité napětí induktoru je s protékajícím proudem vázáno vztahem

$$u = L \frac{di}{dt},$$

kde L je indukčnost [H]

Akumulovaná energie je vyjádřena tímto vztahem

$$W_m = \frac{1}{2} \cdot L \cdot i^2(t).$$

Magnetické pole je charakterizováno okamžitou hodnotou magnetického toku, pro cívku s N závitů platí vztah

$$\Phi = \frac{L \cdot i(t)}{N}.$$

Analýza elektrických obvodů představuje stanovení hodnot proudů protékajících jednotlivými prvky a napětí na jednotlivých prvcích. Pokud chceme stanovit hodnoty uvedených veličin, pak budeme potřebovat příslušný počet lineárně nezávislých rovnic, které se sestaví vhodnou aplikací Kirchhoffových zákonů. Systematický přístup k jejich použití vychází z aplikace poznatků teorie grafů na elektrické obvody, které zkoumáme.

Kirchhoffovy zákony jsou tedy důležitým teoretickým základem pro popis elektrických obvodů. První zákon říká, že součet proudů v řezu je roven nule

$$\sum_i i_k(t) = 0.$$

Druhý zákon pak tvrdí, že součet napětí ve smyčce je roven nule

$$\sum_k u_k(t) = 0.$$

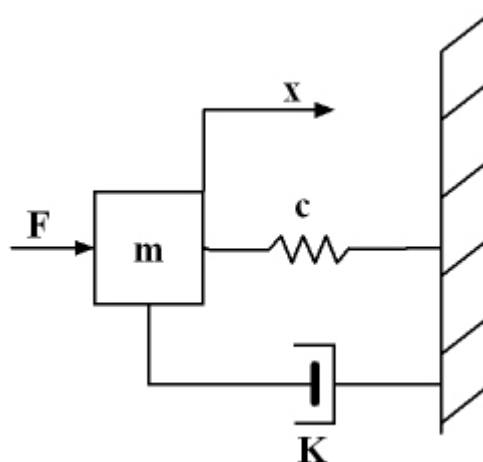
Při aplikaci těchto zákonů musíme v obvodu vyznačit orientace obvodových veličin, kladné smysly obvodových veličin pasivních prvků, daných napětí a proudů.

Takže nyní již víme, jakým způsobem se vytváří matematické modely mechanických a elektrických soustav a z jakých prvků se skládají.

7.3 Ukázka analogií na modelech – první příklad

Jak jsem již uvedl, zaměřím se na analogie elektromechanické. Pokusím se tedy ukázat, že nějaká mechanická soustava má stejný matematický popis jako jiná soustava elektrická.

Prvním analogem, který ukáží, bude **mechanická kmitavá soustava s jedním stupněm volnosti a rezonanční elektrický obvod se sériově řazenými prvky**.



Obr. 7 - Mechanická kmitavá soustava s jedním stupněm volnosti

Mechanická soustava z obrázku 7 se skládá z hmotnosti m , pružiny o lineární poddajnosti c a tlumiče k . Působí na ní dynamická síla $F(\tau)$.

Těleso vykonává pohyb po jedné souřadnici. Po uvolnění tělesa získáme pohybovou rovnici

$$m \cdot \frac{d^2 x}{dt^2} = \sum_i F_i,$$

kde F_i jsou rovny

$$F = m \cdot a,$$

$$F_c = c \cdot x,$$

$$F_k = k \cdot \dot{x},$$

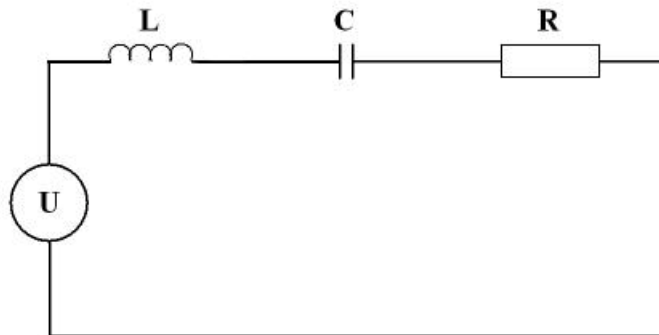
Pokud dosadíme do výše uvedené pohybové rovnice, dostaneme

$$m \cdot \frac{d^2 x}{dt^2} + k \cdot \frac{dx}{dt} + c \cdot x = F.$$

Po úpravě dostaneme

$$m \cdot \ddot{x} + k \cdot \dot{x} + c \cdot x = F$$

Druhým systémem je **analogický rezonanční elektrický obvod se sériově řazenými prvky** (viz Obrázek 8).



Obr. 8 - Rezonanční elektrický obvod se sériově řazenými prvky

Jeho rovnice má následující tvar

$$C\ddot{U} + G\dot{U} + L^{-1}U = \dot{I}(\tau)$$



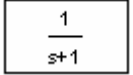



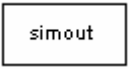




Všimněte si, že rovnice má stejný tvar jako v případě mechanického systému (fyzikální analogie). Co se liší, jsou veličiny a nepůsobí zde dynamická síla, ale zdrojem je elektrický proud.

7.3.1 Porovnání modelů vytvořených v Simulinku

V této podkapitole krátce popíšu, jakým způsobem jsem vytvářel modely v programu Simulink a následně porovnáám výsledné modely obou analogických soustav a ukážu některé grafy.

První věcí, kterou zde nalezneme, je přehled a popis bloků, které jsem použil při stavbě modelů. Předpokládám, že ne každý je plně seznámen s programem Simulink, proto to považuji za vhodné. Přehled bloků naleznete v následující tabulce.

Tabulka 4 – Popis bloků použitých v modelech

Knihovna	Blok	Obrázek bloku	Popis bloku
Continuous	Integrator		numerická integrace vstupního signálu
	Derivative		numerická derivace vstupního signálu
	Transfer Fcn		vyjádření lineární dif. rovnice v přenosovém tvaru
Math operation	Sum		součet vstupních signálů
	Gain		násobení vstupního signálu konstantou
Sinks	Scope		zobrazuje ve formě grafu časový průběh vstupních signálů
	To workspace		zaznamená do proměnné v pracovním prostoru Matlabu průběh signálu
Sources	Constant		zdroj signálu časově neproměnného
	Step		skoková změna signálu v def. čase
	Sine wave		signál popsany matematickou funkcí
	Ramp		signál má standardní tvar

Nyní víme, jaké bloky budou v modelech použity, proto se můžeme pustit do samotné tvorby.

Pokud se tedy rozhodneme řešit první systém, který je popsán lineární diferenciální rovnicí druhého řádu s konstantními koeficienty, existují dva způsoby. Je to metoda snižování řádu derivace a metoda postupné integrace. Já jsem se rozhodl použít metodu první a po úpravě jsem tedy dostal tuto rovnici pro systém první

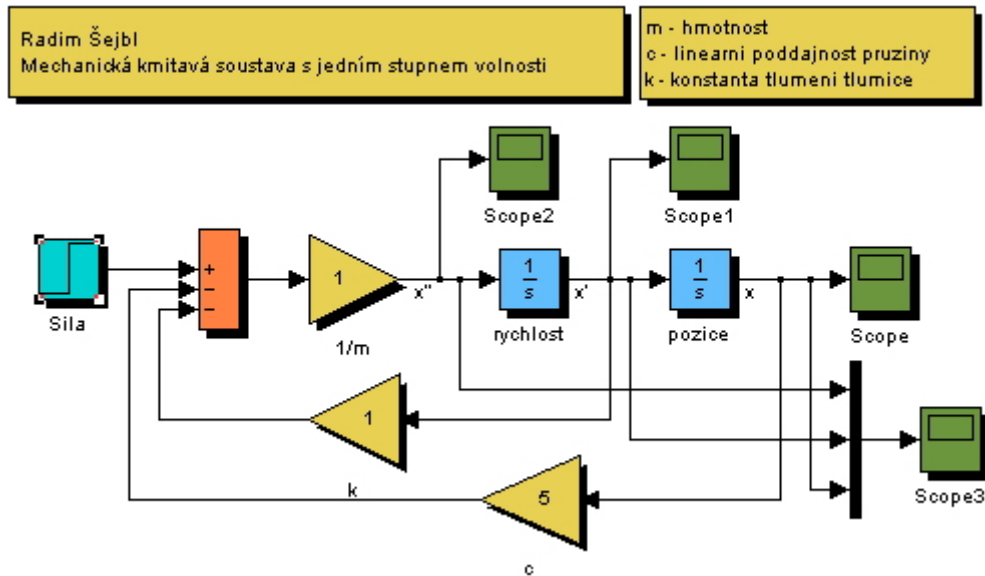
$$\ddot{x} = \frac{1}{m}(F - k \cdot \dot{x} - c \cdot x)$$

a pro systém elektrický

$$\ddot{U} = \frac{1}{C}(I - G \cdot \dot{U} - L^{-1} \cdot U).$$

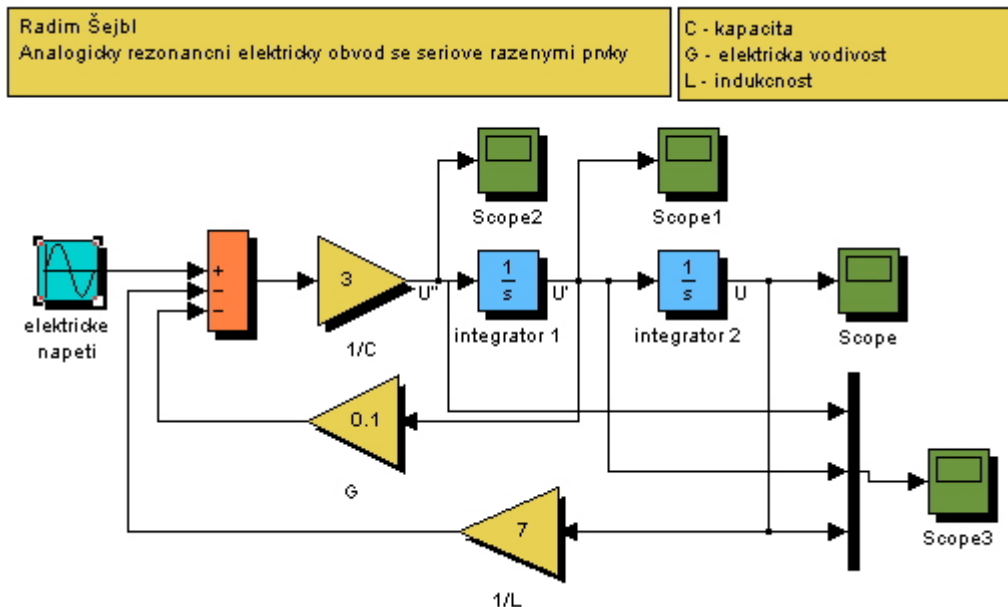
Takto upravené rovnice již není problém namodelovat v Simulinku. Přesný postup vytváření modelů přesahuje rozsah této Diplomové práce, proto si ho dovolím vynechat.

Model mechanické kmitavé soustavy vidíte na následujícím obrázku.



Obr. 9 - Model mechanické kmitavé soustavy

Pro porovnání uvedu i model rezonančního elektrického obvodu.



Obr. 10 - Model rezonančního elektrického obvodu

Z letného porovnání modelů je jasné, že jsou analogické. Jediné, co se liší, je zdroj. U mechanického systému se jedná o zdroj síly a u systému elektrického je to zdroj napětí.

Oba modely lze nalézt na přiloženém CD pod názvy *Příklad1_A.mdl* a *Příklad1_B.mdl* v adresáři *Modely*.

7.3.2 Analýza výsledků

U každého modelu jsou 4 bloky Scope, které zobrazují ve formě grafu časový průběh signálu. První blok (SC2) jsem umístil před první integrátor, druhý (SC1) za něj, třetí (SC) zobrazuje výsledný signál a konečně čtvrtý blok Scope (SC3) zobrazuje všechny 3 signály najednou pro lepší porovnání.

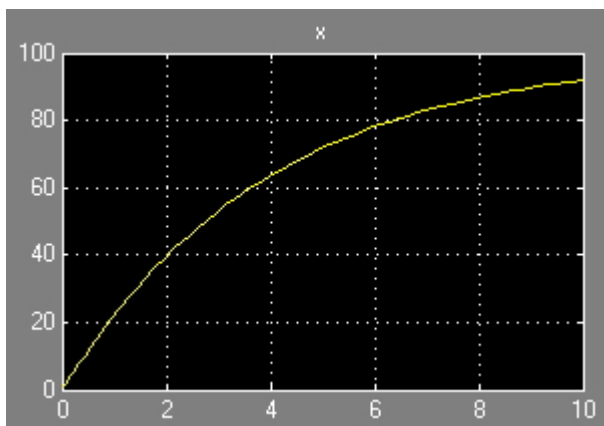
Výsledky pro první, mechanický model jsou zobrazeny v následující tabulce. Grafy zobrazují průběh signálu, který odpovídá následujícím hodnotám:

$$m = 1, c = 5 \text{ a } k = 1$$

Tabulka 5 – Příklad 1 - Model mechanické soustavy - výsledné grafy

Scope	Graf
SC2	
SC1	
SC	
SC3	

Validaci výsledků mohou provést pouhým posouzením výsledných grafů. Tedy jak jsem již napsal, jedná se o kmitavou soustavu a ta se při dostatečné působící síle rozkmitá a po chvíli se vrátí do klidového stavu. Pokud zvedneme k na dostatečně velkou hodnotu při stejné velikosti síly, soustava se nerozkmitá (viz následující obrázek). Z pouhé analýzy výsledných grafů lze usoudit, že vytvořený model funguje správně.

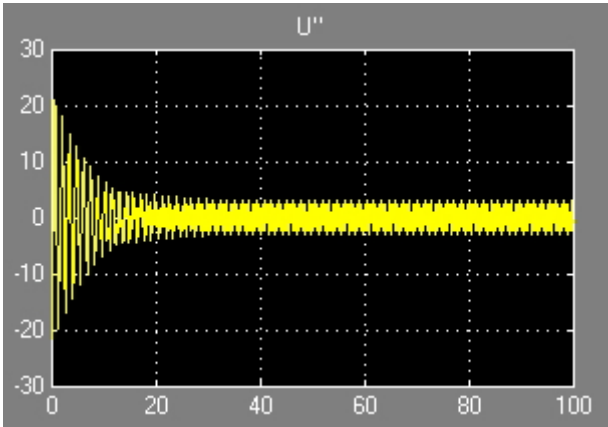
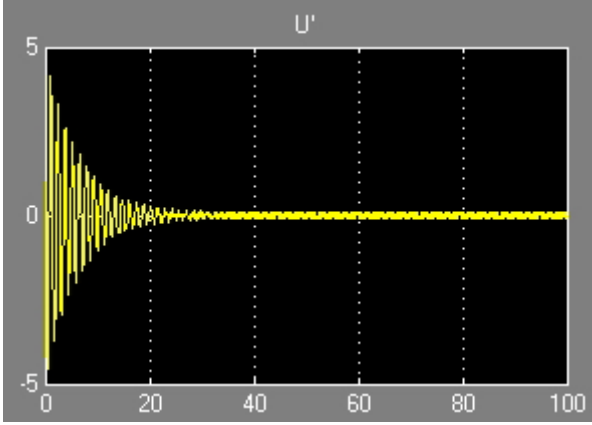
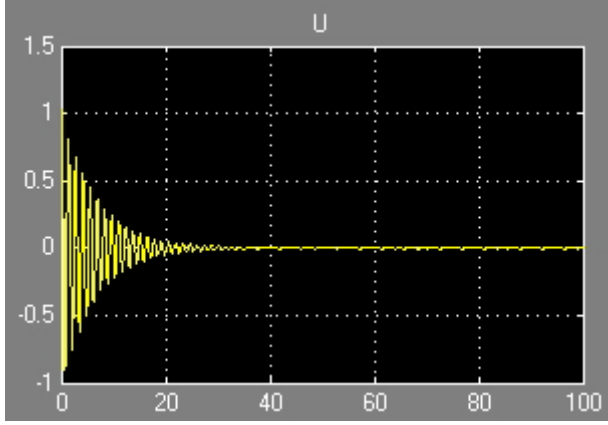
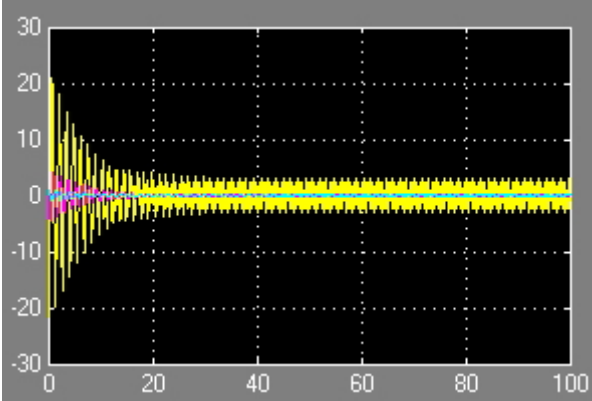


Obr. 11 - Mechanický model - výsledný graf ($k = 20$)

Výsledky pro druhý model (elektrický) zobrazuje tabulka 6. Grafy zobrazují průběh signálu, který odpovídá následujícím hodnotám:

$$C = \frac{1}{3}, G = 0,1, L = \frac{1}{7}$$

Tabulka 6 – Příklad 1 - Model elektrické soustavy - výsledné grafy

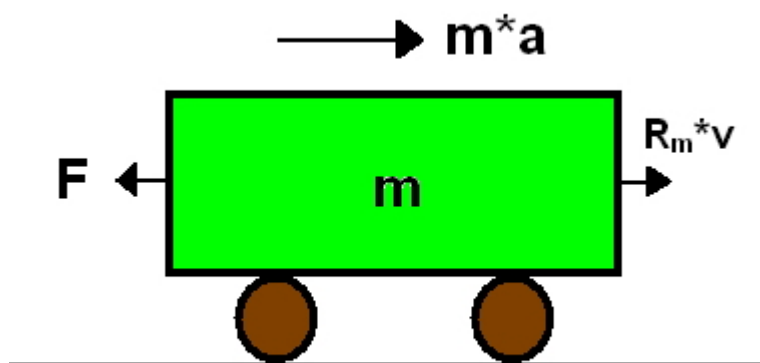
Scope	Graf
SC2	 <p>The graph shows a signal labeled U'' on a grid. The vertical axis ranges from -30 to 30 with major ticks every 10 units. The horizontal axis ranges from 0 to 100 with major ticks every 20 units. The signal starts at approximately 20 at time 0, exhibits damped oscillations, and stabilizes around 0 after approximately 20 time units.</p>
SC1	 <p>The graph shows a signal labeled U' on a grid. The vertical axis ranges from -5 to 5 with major ticks every 1 unit. The horizontal axis ranges from 0 to 100 with major ticks every 20 units. The signal starts at approximately 5 at time 0, exhibits damped oscillations, and stabilizes around 0 after approximately 20 time units.</p>
SC	 <p>The graph shows a signal labeled U on a grid. The vertical axis ranges from -1 to 1.5 with major ticks every 0.5 units. The horizontal axis ranges from 0 to 100 with major ticks every 20 units. The signal starts at approximately 1.5 at time 0, exhibits damped oscillations, and stabilizes around 0 after approximately 20 time units.</p>
SC3	 <p>The graph shows a signal labeled U on a grid. The vertical axis ranges from -30 to 30 with major ticks every 10 units. The horizontal axis ranges from 0 to 100 with major ticks every 20 units. The signal starts at approximately 20 at time 0, exhibits damped oscillations, and stabilizes around 0 after approximately 20 time units. Multiple colored traces (yellow, blue, magenta) are visible, all following the same general trend.</p>

Zde již bude **validace výsledků** horší. Z grafů lze vyčíst, že soustava generuje nějaký kmitavý signál, který se po nějakých 20 vteřinách ustálí, což by odpovídalo elektrickému proudu. Opět můžeme menšími experimenty s nastavováním hodnot dospět k názoru, že se soustava chová správně. Pokud budeme například zvyšovat C , signál se bude úměrně zvyšovat. Zvyšováním G naopak signál utlumíme.

7.3 Druhá ukázka elektromechanické analogie

Druhým příkladem, který předvedu, je elektrická analogie k mechanickému systému pohybu vozidla.

Jedná se o jednoduchý příklad (viz následující obrázek), kdy na těleso (v tomto případě vozidlo) o hmotnosti m působí nějaká síla F a proti působí mechanický odpor R_m .



Obr. 12 - Mechanický systém - pohyb automobilu

Pokud sečteme všechny síly působící na těleso, dostaneme následující rovnici

$$F = m \cdot \frac{dv}{dt} + R_m \cdot v$$

Po integraci zrychlení získáme rychlost:

$$a = \frac{dv}{dt} \equiv \int \frac{dv}{dt} = v$$

Výsledná diferenciální rovnice prvního řádu pak bude vypadat takto

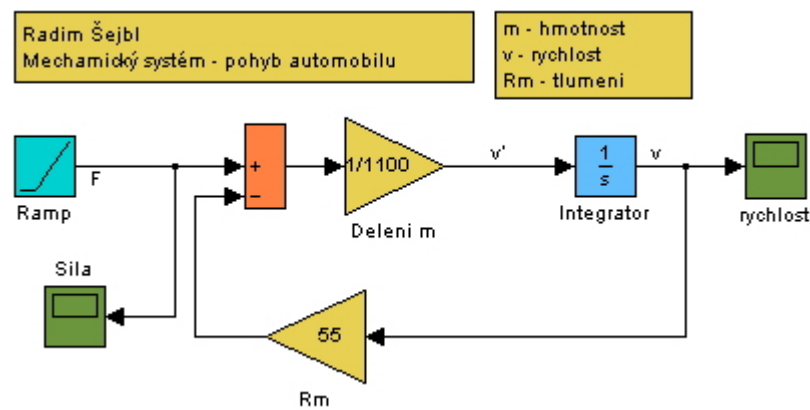
$$a = \frac{F - R_m \cdot v}{m}$$

Druhý systém je elektrický obvod, jehož zdrojem je elektrické napětí U . Jeho analogická rovnice k rovnici mechanického systému vypadá následovně:

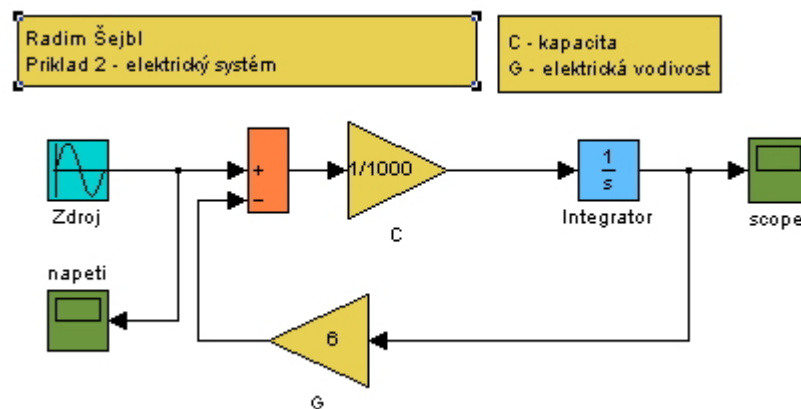
$$\dot{U} = \frac{I - G \cdot U}{C}$$

K nalezení analogické diferenciální rovnice je potřeba nelézt analogické fyzikální veličiny. V tomto případě síle F odpovídá elektrický proud I , mechanickému odporu R_m pak elektrická vodivost G atd. Menší přehled analogických fyzikálních veličin naleznete v tabulce 2, rozsáhlejší je možné hledat například v [7].

Modely obou soustav vypadají následovně:



Obr. 13 - Příklad 2 - Mechanický systém



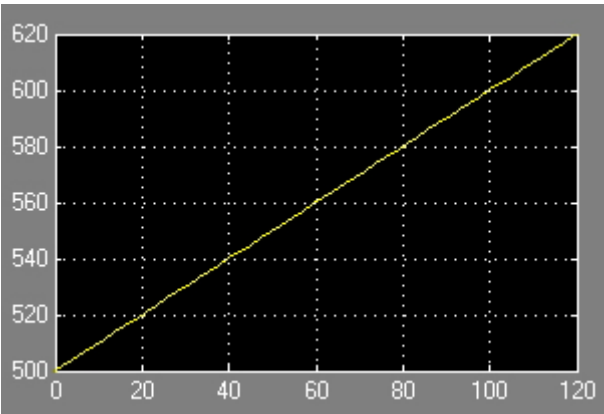
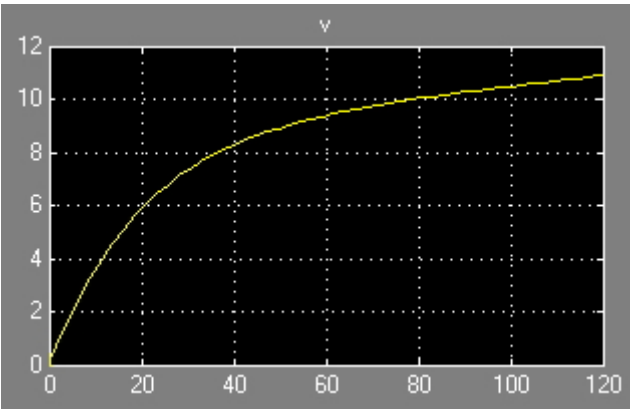
Obr. 14 - Příklad 2 - Elektrický systém

Z prvního letmého pohledu je zřejmé, že modely jsou fyzikálně analogické. Jediné, co se muselo v druhém modelu změnit, je zdroj a hodnoty jednotlivých veličin.

Opět je možné oba modely nalézt na přiloženém CD pod názvy *Příklad2_A.mdl* a *Příklad2_B.mdl* v adresáři *Modely*.

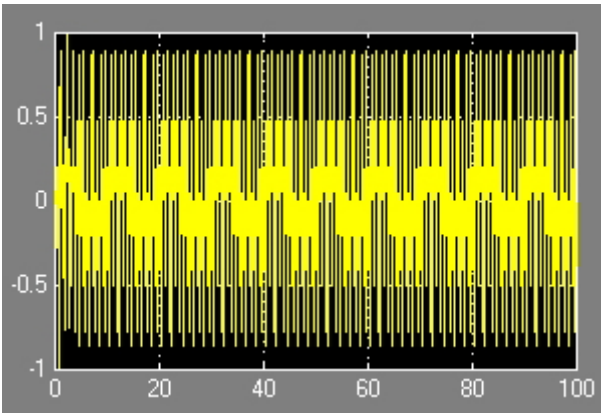
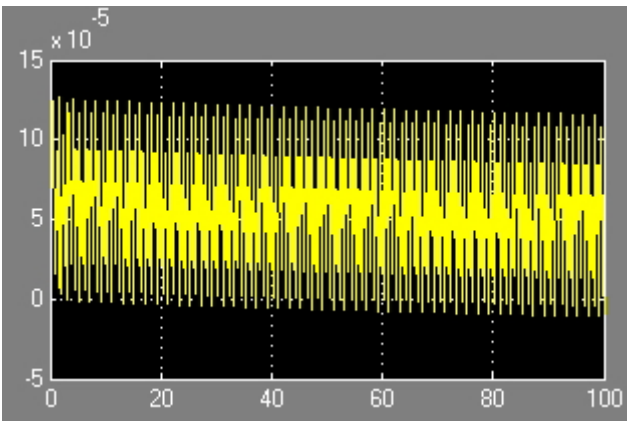
Výsledné grafy zobrazující průběh signálu jsou v následujících tabulkách (hodnoty jednotlivých veličin lze vyčíst z modelů):

Tabulka 7 – Příklad 2 - mechanická soustava - výsledné grafy

Scope	Graf
Sila	
Rychlost	

Validaci je opět možné provést pouhou analýzou výsledných grafů. U mechanického systému je jasně vidět, že model funguje správně. Při narůstající velikosti síly se postupně zrychluje daný objekt (vozidlo).

Tabulka 8 – Příklad 2 - elektrická soustava - výsledné grafy

Scope	Graf
Napětí	
Scope	

U elektrického systému z grafů můžeme vyčíst jasný kmitavý pravidelný signál, který by mohl odpovídat elektrickému proudu.

V této kapitole jsem na dvou příkladech pomocí modelů vytvořených v programu Simulink ukázal, jak fyzikální analogie vypadá, jak se nachází analogické systémy a především, že opravdu funguje a je možné ji snadno využívat.

8 Zpracování souborů z CD přílohy

Jak jste si mohli všimnout, všechny modely, které sloužily jako názorné ukázky analogií, je možné nalézt na přiloženém CD v adresáři *Modely*.

V následující tabulce je přehled všech přiložených souborů a jejich popis.

Tabulka 9 - Přehled souborů přiložených na CD

	Model	Simulink soubor
1.	Mechanická kmitavá soustava s jedním stupněm volnosti	<i>Příklad1_A.mdl</i>
2.	Rezonanční elektrický obvod se sériově řazenými prvky	<i>Příklad1_B.mdl</i>
3.	Mechanický systém - pohyb automobilu	<i>Příklad2_A.mdl</i>
4.	Analogický elektrický systém	<i>Příklad2_B.mdl</i>

Všechny modely byly vytvořeny v počítačové učebně DFJP, kde je nainstalován **Matlab verze 6.5**. Proto je nutné i všechny modely na příloze spouštět v této verzi programu Matlab. Kompatibilita se starými a bohužel i novějšími verzemi tohoto softwaru není zaručena.

Nejdříve je nutné soubor, se kterým chceme pracovat, přehrát do nějakého pracovního adresáře. Soubory je poté možné spustit dvěma způsoby.

1. Dvojitě poklepnutím myši na soubor automaticky otevře Matlab a Simulink a spustí příklad.
2. Nebo je možné nejdříve spustit Matlab a zadat do příkazového řádku příkaz „*simulink*“ nebo kliknout na ikonu Simulink v horním menu. Otevře se nám hlavní okno Simulinku. Pak už jen stačí zvolit *File – Open* (nebo *ctrl + O*) a vybrat soubor, který chceme otevřít.

9 Závěr

Hlavními cíly této práce bylo seznámení s problematikou fyzikální analogie popisu modelů spojité simulace. Nejdříve jsem popsal, jakými způsoby je možné řešit dynamické systémy a jaký vhodný software vybrat pro simulaci. Asi nejdůležitějšími částmi byl přehled analogických fyzikálních modelů a ukázky analogií na modelech vytvořených v programu Simulink. Tyto cíle se dle mého názoru splnit podařilo.

Co by tedy bylo vhodné si z této práce zapamatovat? Určitě je dobré vědět, že existuje několik základních matematických prostředků a metod pro simulaci, z nichž nejdůležitější z hlediska této práce jsou diferenciální a diferenciální rovnice. Spojitý simulační model, jeho definice a metody numerické integrace jsou zajisté také důležité teoretické základy. Neméně důležité je také vědět něco o produktu Matlab a jeho nadstavbě Simulink, což je software, bez kterého se těžko obejdeme při modelování dynamických systémů.

Ale hlavní téma této práce byla samozřejmě fyzikální analogie. Zde bychom měli věnovat pozornost především způsobu modelování spojitých systémů pomocí fyzikální analogie, elektrickým analogům fyzikálních obvodů a přehledům analogických dynamických obvodů a analogických fyzikálních veličin. Dvě ukázky elektromagnetických analogií na modelech vytvořených v programu Simulink pak poslouží jako výborná demonstrace toho, jak fyzikální analogie fungují.

Fyzikální analogie je dnes často používaná technika popisu a řešení modelů spojité simulace a není tedy důvod, aby se od ní v budoucnosti opouštělo. Její kladné vlastnosti jako jednoduchost, nenáročnost na použité prostředky a měřitelnost zaručují její budoucí vývoj. Právě k seznámení s touto technikou popisu modelů spojité simulace byla určena moje práce.

Seznam tabulek

<i>Tabulka 1 – Analogie ustálených fyzikálních polí. Zdroj: [7]</i>	32
<i>Tabulka 2 – Přiřazení fyzikálních veličin obecným veličinám. Zdroj: [1]</i>	37
<i>Tabulka 3 – Analogické dynamické obvody. Zdroj: [7]</i>	40
<i>Tabulka 4 – Popis bloků použitých v modelech</i>	51
<i>Tabulka 5 – Příklad 1 - Model mechanické soustavy - výsledné grafy</i>	54
<i>Tabulka 6 – Příklad 1 - Model elektrické soustavy - výsledné grafy</i>	56
<i>Tabulka 7 – Příklad 2 - mechanická soustava - výsledné grafy</i>	59
<i>Tabulka 8 – Příklad 2 - elektrická soustava - výsledné grafy</i>	60
<i>Tabulka 9 - Přehled souborů přiložených na CD</i>	61

Seznam obrázků

<i>Obr. 1 - Hlavní okno Matlabu</i>	22
<i>Obr. 2 - Simulink - okno s knihovnami</i>	24
<i>Obr. 3 - Simulink - okno s příkladem</i>	25
<i>Obr. 4 - Simulink - graf</i>	26
<i>Obr. 5 - VisSim</i>	28
<i>Obr. 6 - 20-sim</i>	29
<i>Obr. 7 - Mechanická kmitavá soustava s jedním stupněm volnosti</i>	48
<i>Obr. 8 - Rezonanční elektrický obvod se sériově řazenými prvky</i>	49
<i>Obr. 9 - Model mechanické kmitavé soustavy</i>	52
<i>Obr. 10 - Model rezonančního elektrického obvodu</i>	52
<i>Obr. 11 - Mechanický model - výsledný graf ($k = 20$)</i>	55
<i>Obr. 12 - Mechanický systém - pohyb automobilu</i>	57
<i>Obr. 13 - Příklad 2 - Mechanický systém</i>	58
<i>Obr. 14 - Příklad 2 - Elektrický systém</i>	58

Příloha – obsah přiloženého datového nosiče (CD)

Součástí této práce je přiložený datový nosič (CD) s následujícím obsahem:

/Modely/

- zde jsou uloženy všechny modely, které sloužily jako ukázka analogií. Tento adresář obsahuje tyto soubory:

Příklad1_A.mdl

Příklad1_B.mdl

Příklad2_A.mdl

Příklad2_B.mdl

Jejich obsah a možnosti spuštění jsou popsány v kapitole 8.

Seznam použitých informačních zdrojů

- [1] NOSKIEVIČ, P. *Modelování a identifikace systémů*, MONTNEX a.s., Ostrava, 1999, ISBN 80-7225-030-2
- [2] DUŠEK, F.; HONC D., *Matlab a simulink*, Fakulta chemicko-technologická, Univerzita Pardubice, 2005, ISBN 80-7194-776-8
- [3] KŘIVÝ, I.; KINDLER, E., *Simulace a modelování*, Přírodovědecká fakulta, Ostravská Univerzita, 2001
- [4] *WWW stránky produktu Matlab* [online] Dostupné z: <<http://www.mathworks.com/>>
- [5] *WWW stránky produktu 20-sim* [online] [cit. 2006-09-03] Dostupné z: <<http://www.20sim.com/>>
- [6] *WWW stránky produktu VisSim* [online] [cit. 2006-09-03] Dostupné z: <<http://www.vissim.com/>>
- [7] KUNEŠ, J.; VAVROCH, O.; FRANTA, V., *Základy modelování*, Státní nakladatelství technické literatury, Praha, 1989, ISBN 80-03-00147-1
- [8] DEBNEY, J., B.; HARMAN, T., L., *Mastering simulink*, University of Houston – Clear Lake, 2004, ISBN 0-13-142477-7
- [9] ANGOT, A. *Užitá matematika pro elektrotechnické inženýry*, Státní nakladatelství technické literatury, Praha, 1972
- [10] *Using Simulink* [online] Dostupné z: <<http://www-ccs.ucsd.edu/matlab/toolbox/simulink/simulink.html>>
- [11] HERINGOVÁ, B.; HORA, P., *MATLAB Díl I. – Práce s programem* [online] [cit. 2006-09-01] Dostupné z: <<http://www.cdm.cas.cz/czech/hora/vyuka/mvs/tutorial.pdf>>
- [12] *Simulink* [online] Dostupné z: <<http://volt.feld.cvut.cz/vyuka/DMS/matlab/simulink.html>>

ÚDAJE PRO KNIHOVNICKOU DATABÁZI

Název práce	Fyzikální analogie popisu modelů spojité simulace
Autor práce	Bc. Radim Šejbl
Obor	Aplikovaná informatika v dopravě
Rok obhajoby	2006
Vedoucí práce	Ing. Oktavián Strádal, Ph.D.
Anotace	<p>Tato práce je zaměřena na problematiku fyzikálních analogií modelů spojité simulace. Jsou zde také popsány různé způsoby a možnosti popisu a řešení spojitých systémů. Dále se zabývám otázkou doporučení vhodného softwaru pro simulaci dynamických systémů. Důležitou částí práce je přehled analogických fyzikálních modelů a ukázka analogií na modelech vytvořených v programu Simulink.</p>
Klíčová slova	Simulace, modelování, fyzikální analogie, Matlab, Simulink