**Dolní propust**

**Milan Poláček**

# Úvod

V minulých úlohách (například ve třetí úloze) jsme se zaobírali pulsním tokem. Úkolem bylo vygenerovat pulsy toku. Nyní si vyzkoušíme tyto pulsy vyhladit primitivní aproximací dolní propusti s rozprostřenými parametry.

|  |
| --- |
|  |
| **Obrázek 1: Blokové schéma hotového modelu** |

# Zadání

1. Zkonstruujte model dolní propusti (***DpBlock***) – z bloků odporu a poddajnosti. Dostaneme se o úroveň níže, kde nataháte konektory a místo psaní rovnic nataháte i bločky. Můžete využít naší komoru, kde na vstup elastance dáte blok konstanty – například takto:

|  |
| --- |
|  |
| **Obrázek 2: Zapojení submodelu cévy – dolní propust. Využijeme normální konektory, do kterých jsou zapojeny vnitřní komponenty.** |

1. Vytvořte model dolní propusti v rovnicovém vyjádření (*DpEq*) – místo dvou bloků bude obsahovat pouze rovnice. Kolik jich budeme potřebovat? Pozor, nyní rozhodně neplatí, že vtok = výtok, neboť část toku se v bloku může akumulovat! Kolik jsme ušetřili rovnic oproti blokovému zapojení? Kam se poděly, když máme stejnou funkcionalitu? Odpovězte do zprávy.
2. Porovnejte chování obou bloků ve vlastním testovacím zapojení. (Např. je zapojte mezi zdroje tlaku.) Průběhy tlaků a objemů musí být stejné!
3. Vysvětlete, proč musíme použít ještě jeden odpor za cévou a proč nemůžeme přímo spojit dvě poddajnosti.
4. Vytvořte model céva, který bude jednobranem2 našeho konektoru a bude obsahovat:
   1. pole *DpEq* o délce *num*. Prvky budou parametrizované  
      *(each R=paramR, each C=paramC),*kde paramR a paramC jsou zas parametry celého modelu céva. Ty určují parametry bločků odporu a poddajnosti.

Lze to napsat klidně tímto způsobem, kde R má dvojí význam – nejprv jako parametr v objektu Odporu a poté jako parametr v objektu *dpBlock*:

parameter Real R;

Odpor [num] odpor(each R=R)

* 1. smyčku spojující jednotlivé prvky rovnicí *connect*

*Connect se chová jako rovnice a můžeme s ní manipulovat i pomocí polí. Zde zapojení prvního prvku na vstupní konektor:*

connect(dpBlock1[1].q\_in, q\_in);

* 1. Integer *num*, definující délku pole

1. Zapojte v modelu Céva jen jeden prvek s parametry R = 4, C = 4
2. Zapojte v modelu Céva dva prvky s parametry R=2, C = 2
3. Zapojte v modelu Céva 4 prvky s parametry R=1, C=1
4. Porovnejte a vysvětlete výsledné průběhy.

# Bonus (+2b)

Na cvičení jsme dělali model zahřívání tyče a to pouze v rovnicích. Obdobně jako v příkladu výše, navrhněte prvek (model), jehož zřetězením dosáhneme stejné funkce (+1b), tj. rozdělení teploty po ose x. Ukažte, že průběhy modelu s rozdělením do bloků je stejné, jak z rovnic ze cvičení (+1b).

# Bonus (+0.5b)

Za první report každé případné chyby nalezené v zadání. Reportujte na fórum.

# Nápověda

Klasicky na fóru.

# Řešení

Dle pokynů v zadání jsem vytvořil modely tzv. sledovače a regulace, který na základě stanovených hranic (Cmin, Cmax) řídil hodnotu koncentrace (viz graf 1).

|  |
| --- |
| C:\Users\Milhouse\Documents\Scholla\___FEL\21rocnik\MOS\5\Diskretni.png |
| 1. Graf řízení koncentrace dle stanovených hranic Cmin a Cmax |

# Diskuze

Model nultého řádu je nepřesný v tom, že nemá ošetřené parametry pro záporné hodnoty a proto koncentrace při delší simulaci skončí v záporných hodnotách, což je nesmysl.

U modelu prvního řádu to není uvažována saturace podávané látka, a proto je model nepřesný.

Náš dvou kompartmentový systém zanedbává, že by každý kompartment měl mít vstup a výstup a v důsledku toho se jedná o dva kompartmenty zapojené paralelně. Látka se tedy oproti reálné situaci distribuuje ve stejnou dobu a stejnou rychlostí. Nevzniká tedy žádné tzv. dopravní zpoždění jako by se dělo u podání léku pacientovi.

Při přidání plicní eliminace nám vznikne systém s eliminací nultého řádu.

Rozdíl mezi nultým a prvním řádem je, že koncentrace u nultého řádu klesá lineárně a u prvního řádu klesá patrně podle exponenciály (viz graf 2).

|  |
| --- |
| C:\Users\Milhouse\Documents\Schule\FEL\21rocnik\MOS\5_Uloha\eliminace.png |
| 1. Eliminace koncentrace v čase u nultého a prvního řádu |

# Bonus 1

|  |
| --- |
| C:\Users\Milhouse\Documents\Scholla\___FEL\21rocnik\MOS\5\spojite.png |
| 1. Graf řízení koncentrace dle zadání bonusu1 |

Navržená simulace odpovídá zadání a udržuje hladinu dávkování dle zadání.

# Bonus 2

Parametry jsou doseLen, dpd, dose. Proměnné jsou prePt, doseInterval, doseFlow a pulseTime, která je jako jediná také diskrétní.

Dle logického uvažování by dávka měla být stejná jen při prvním podání. Bohužel grafy modelů tomu neodpovídají, jelikož se mi nepodařilo rovnice podřídit patřičnému časovému offsetu.

|  |
| --- |
| C:\Users\Milhouse\Documents\Scholla\___FEL\21rocnik\MOS\5\koncentracedenne.png |
| 1. Koncentrace léku v krvi při dávkách po 6 hod. (modře) a po 24 hod. (červeně) |

# Závěr

I přes veškeré snahy se mi nepodařilo ani s konzultací kolegů nalézt chybu v druhém bonusovém úkolu. Doufám, že bude možno někde získat správné řešení či konzultaci k tomuto bodu úkolu.