

HUMMOD - GOLEM EDITION – ROZSÁHLÝ MODEL FYZIOLOGICKÝCH SYSTÉMŮ

Marek Mateják, Jiří Kofránek

Anotace

HumMod - Golem edition je naše implementace rozsáhlého modelu integrovaných fyziologických systémů vyvinutých týmem Mississippi University Medical Center. Na rozdíl od amerických autorů je náš model je implementován v akauzálním jazyce Modelica, což umožnilo vysoké zpřehlednění modelu i možnost hledání ustálených stavů úpravou rovnic modelu. Model jsme modifikovali a rozšířili zejména v oblasti fyziologie vnitřního prostředí, acidobazické rovnováhy a přenosu krevních plynů. Model je vyvíjen jako teoretický podklad pro lékařské výukové simulátory.

Klíčová slova:

Akauzální modelování, Fyziologie, Modely, Modelica

1. Úvod

Model **HumMod** (<http://hummod.org>) vytvářený týmem Mississippi University Medical Center patří k nejrozsáhlejším modelům fyziologických systémů (obsahuje 4000 proměnných). Rozvoj modelu je financován NASA a model slouží pro studium vlivu dlouhodobého kosmického letu na lidský organismus.

Další oblastí jeho potenciálního využití jsou lékařské simulátory pro výuku lékařů, zejména v oblasti medicíny akutních stavů.

Hlavním architektem tohoto rozsáhlého modelu je Thomas Coleman, dlouholetý spolupracovník A. C. Guytona, spoluautor dnes již legendárního článku ^[12], popisujícího fyziologické návaznosti regulace oběhu, který byl jedním z milníků vzniku tzv. integrativní fyziologie, zkoumající funkce propojených fyziologických systémů organismu jako celku s využitím matematických modelů ^[7]. Thomas Coleman si již v osmdesátých letech uvědomoval velké možnosti modelů propojených fyziologických systémů ve výuce a implementoval ve Fortranu výukový simulátor **Human** ^[6]. Moderním pokračováním simulátoru Human byl model **Quantitative Circulatory Physiology (QCP)** ^[1], naprogramovaný v jazyce C++. Coleman se později rozhodl plně zveřejnit strukturu modelu (všechny matematické vztahy a všechny hodnoty konstant a počátečních hodnoty stavových proměnných). Vytvořil speciální programovací jazyk (ve formě XML souborů), a příslušný překladač. Model, nazvaný **Quantitative Human Physiology (QHP)**, nedávno přejmenovaný na **HumMod**, je pak šířen (i s překladačem) jako Open Source ^[15,16,17]. Uživatel pak může model upravovat i modifikovat. Jak je zmíněno v přehledové přednášce v tomto sborníku ^[21], potíží tkví ale v tom, že zdrojové XML texty celého modelu jsou napsány v celkem 3235 souborech umístěných v 1367 složkách. Celková struktura modelu a jednotlivé návaznosti jsou pak velmi nepřehledné a pro řadu tvůrčích modelovacích týmů je model HumMod,

jako podklad pro další rozvoj a modifikace modelu, nepoužitelný^[39].

Pro přehledné zobrazení matematických vztahů z XML jsme nejprve vytvořili speciální softwarový nástroj **QHPView**, který nám rozkryl matematické vztahy zakódované v tisícovkách XML souborů. To nám umožnilo se v rozsáhlém modelu orientovat. Navázali jsme pak úzkou pracovní spoluprací a americkými autory.

Model QHP/HumMod jsme proto mohli zvolit jako základní východisko pro vývoj integrovaného modelu fyziologických regulací, který bude teoretickým podkladem námi vyvíjených výukových lékařských simulátorů.

Pro implementaci tohoto rozsáhlého, hierarchicky uspořádaného modelu bylo nutno nejprve zvolit vhodné vývojové prostředí a vhodné vývojové nástroje.

V poslední době se na trhu objevila skupina velice efektivních modelovacích nástrojů, které umožňují popisovat jednotlivé části modelu **přímo jako soustavu rovnic a nikoli jako algoritmus řešení těchto rovnic**. Jedním z těchto tzv. akauzálních modelovacích nástrojů je jazyk **Modelica**.

2. Modelica jako základní modelovací nástroj

Jazyk Modelica^[11] patří k tzv. **akauzálním** programovacím jazykům, které, na rozdíl od blokově orientovaných jazyků (kam patří např. Simulink), pracují přímo s rovnicemi (a způsob jejich řešení přenechávají kompilátoru)^[23].

Modelica nachází stále větší uplatnění zejména v průmyslových aplikacích. V biomedicínských aplikacích se však Modelica prozatím příliš neuplatnila.

Drtivá většina biomedicínských simulačních aplikací je dosud realizována v kauzálních blokově orientovaných prostředích. Patří k nim například vývojové prostředí referenčních databází biomedicínských modelů (v jazycích JSIM <http://physiome.org/model/doku.php> nebo CEIIML <http://www.cellml.org>).

Zhusta využívaným prostředím v biologii a medicíně je Matlab/Simulink - monografie věnované biomedicínským modelům bývají často doprovázeny přídatným softwarem pro toto vývojové prostředí, zatím ale bez využití nových akauzálních simulinkových knihoven^[např. 40, 29, 34].

Nicméně již v roce 2006 Cellier a Nebot^[4] ukázali výhody, které Modelica přináší pro přehlednou implementaci popisu fyziologických systémů. V Modelice implementovali klasický McLeodův model cirkulačního systému PHYSBE (PHYSiological Simulation Benchmark Experiment)^[31, 32, 33]. Tyto rozdíly zvláště vyniknou, porovnáme-li si Cellierovu implementaci modelu^[4] s volně stažitelnou verzí implementace modelu PHYSBE v Simulinku <http://www.mathworks.com/products/demos/simulink/physbe/>.

Haas a Burnhan ve své nedávno vydané monografii upozorňují na velké možnosti jazyka Modelica pro modelování medicínských adaptivních regulačních systémů^[13]. Nejnověji Brugard a spol.^[2] referují o práci na implementaci knihovny značkovacího jazyka SBLM (System Biology Markup Language), používaného jako jeden ze standardů pro popisování modelů biologických systémů (<http://sbml.org/>), do jazyka Modelica. To by v budoucnu umožnilo jednoduchým způsobem přímo spouštět modely, jejichž struktura

je popsana v jazyce SBLM, na vyvojovych platformach, zalozenych na jazyce Modelica.

Nicmene akauzalni modely je dnes mozne vytvaret i v Simulinku s využitim novych akauzalnich knihoven (Simscape a dalsich).

V Matlabu a Simulinku jsme po leta vyvijeli modely fyziologickych systemu a rozvijeli prslunou aplikani simulinkovou knihovnu Physioblibrary (<http://physiome.cz/simchips>). Vyvinuli jsme take prslusne softwarove nstroje usnadnujici prevod modelu implementovanych v Simulinku do vyvojovych prostedı (Control Web a Microsoft .NET), v nichz vytvarime vlastnı vyukove simulatory. Ns vyvojovı tym ma dlouholetou praxı pomerne slusne zkusenosti v pracı s vyvojovym prostedım Matlab/Simulink od renomovane firmy MathWorks. Na druhe strane ns lakaly nove moznosti vyvojovych prostedı vyuzıvajici jazyk Modelica.

Pred implementacı rozsahleho modelu QHP/HumMod jsme proto stali pred rozhodnutım, zda nadale pokracovat ve vyvoji modelu fyziologickych systemu v prostedı Simulink (s využitım novych akauzalnich knihoven), nebo zda uinit radikalnejsı rozhodnutı a prejıt na novou platformu jazyka Modelica.

Model jsme se nejprve pokusili implementovat v prostedı Simulink.

V modelu se vyskytuje cela řada vztahu, ktere vedou na řeenı implicitnıch rovnic. Z tohoto duvodu je blokove orientovana implementace modelu (kde vystupy jednoho bloku jsou vyuzity jako vstupy do dalsıch bloku) velmi slozita a v prubehu implementace se stoupajici slozitostı modelu rapidne klesala jeho prehlednost. Využitı novych akauzalnich simulinkovych knihoven v takto slozitem modelu se ukazalo problematickym a prehlednost modelu se prilısı nezvysila.

Proto jsme simulinkovou implementaci prerusili a model zaali implementovat pomocı jazyka Modelica (s využitım vyvojoveho prostedı Dymola (<http://www.3ds.com/products/catia/portfolio/>). Velmi rychle se ukazalo, ze **implementace rozsahleho modelu v Modelice je mnohem efektivnejsı nez pouhe vyuzıvanı akauzalnich knihoven v Simulinku**. Prı porovnanı simulinkove a modelicove implementace se projevil podstatny rozdıl, spoıvajici zrejme v tom, ze nove akauzalni knihovny jsou pouhou akauzalni nadstavbou Simulinku a nikoli objektove orientovanym na rovnicıch postavenym modelovacım jazykem, jakym je Modelica.

Jak jsme jız referovali na lonske konferenci MEDSOFT ^[30], implementace modelu QHP/HumMod v Modelice podstatne zprehlednila strukturu modelu a mimo jıne pomohla odhalit i nektere chyby v pvodnı americke implementaci modelu HumMod.

Porovnane-li vyvojova prostedı, zalozena na simulanım jazyce Modelica s vyvojovym prostedım Matlab/Simulink od firmy Mathworks mužeme nyní, po vıce ne z dvoulete zkusenosti s vyvojem modelu v Modelice, konstatovat ze:

- na rozdıl od Simulinku model implementovany v Modelice mnohem lepe vystihuje podstatu modelovane reality a **simulanı modely jsou mnohem ˇıtelnejsı i mene nachylne k chybam**;

- **objektová architektura** Modeliky umožňuje **postupně stavět a ladit modely s hierarchickým uspořádáním s využitím knihoven znovupoužitelných prvků**;
- na rozdíl od Simulinku (který je průmyslovým standardem firmy Mathworks) je Modelica **normalizovaný programovací jazyk**, proto také mohou **existovat různá komerční (i nekomerční) vzájemně si konkurující vývojová prostředí** a pro řešení specifických problémů z různých aplikačních oblastí se v tomto jazyce vytvářejí (**komerční i nekomerční specializované knihovny**);
- v Modelice je možné **nenásilně kombinovat kauzální (většinou signálové) a akauzální vazby**; na rozdíl od Simulinku v propojení kauzálních bloků je možné bez větších problémů vytvářet algebraické smyčky – součástí kompilátoru Modeliky jsou symbolické manipulace na pozadí a proto **rozpojení algebraických smyček je starostí vývojového prostředí a nikoli programátora**.

Výše uvedené důvody nás vedly k tomu, že jako **hlavní implementační prostředek** pro tvorbu modelů jsme zvolili **Modelicu** a postupně upouštíme od vývoje modelů v prostředí Matlab/Simulink [19, 23, 24, 25, 26].

Do tvorby aplikačních knihoven a nástrojů vývojových prostředí pro jazyk Modelica jsme se zapojili i v rámci mezinárodní spolupráce.

Společným úsilím 14 firem a 11 univerzit sdružených v Open Modelica Source Consortium je společně vyvíjeno vývojové prostředí Open Modelica a překladač jazyka Modelica – Open Modelica Compiler (<http://www.ida.liu.se/labs/pelab/modelica/OpenSourceModelicaConsortium.html>).

Tohoto vývoje se účastní i náš vývojový tým v rámci firmy Creative Connections s.r.o., která je členem tohoto konsorcia (viz <http://www.creativeconnections.cz/>). Pro toto konsorcium jsme vyvinuli nástroj, umožňující z modelu vyvinutého a odladěného v Modelice vygenerovat zdrojový text modelu v jazyce C#, což nám umožní vytvářet modely pustitelné v internetovém prohlížeči s nainstalovaným pluginem Silverlight [26].

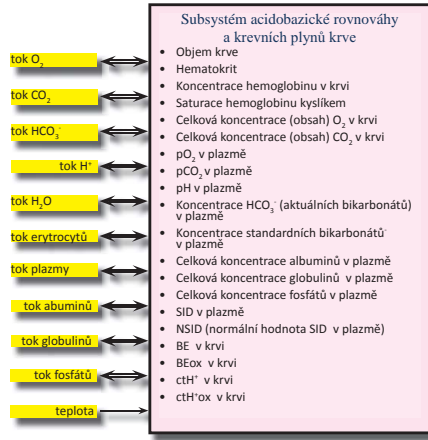
3. HumMod-Golem Edition

Přehlednost modelu nám okamžitě umožnila vyjevit některé chyby v implementaci původního modelu o nichž jsme americké kolegy informovali.

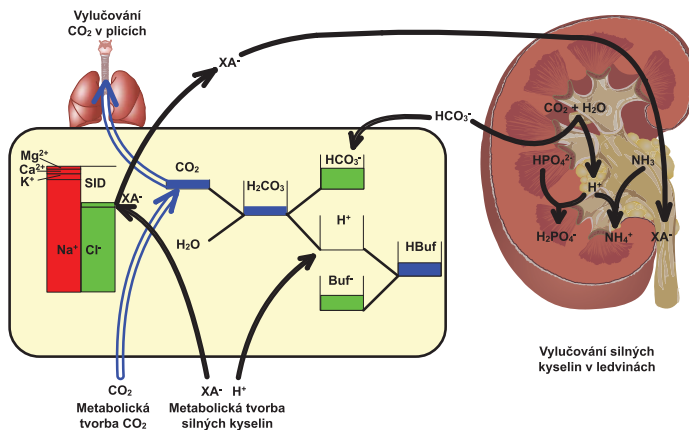
Model **HumMod** jsme modifikovali a rozšířili především v oblasti modelování přenosu krevních plynů a homeostázy vnitřního prostředí, protože poruchy právě těchto subsystémů jsou časté právě v akutních medicíně, pro kterou je koncipován námi vyvíjený simulátor a výukové simulační hry.

Při modifikaci jsme mimo jiné vycházeli z našeho původního komplexního modelu fyziologických regulací, který byl jádrem výukového simulátoru **Golem** [21, 22] a z [16, 18, 35, 36].

Tak např. jsme zásadně modifikovali subsystém acidobazické rovnováhy. Američtí autoři vycházeli z tzv. Stewartova přístupu k acidobazické rovnováze [8, 9, 10, 37, 38]. Jeho podstata spočívá na rovnici, která je důsledkem chemických



Obrázek 1 – Model subsystému acidobazické rovnováhy a krevních plynů krve. Na rozdíl od Stewartova přístupu je zde hodnota SID výstupním parametrem (závislou proměnnou), stejně jako i parciální tlaky krevních plynů. Jedním z výstupů je i normální hodnota SID v plazmě, jako taková hodnota SID, kdyby při $pCO_2=40$ torr byla hodnota $pH=7.4$. Vstupy modelu jsou látkové toky a teplota. To umožní využít model jako subsystém v modelu vyšší hierarchické úrovně, kde jsou látkové toky (vstup subsystému) často řízeny gradientem koncentrací látek (výstup subsystému). Jedním ze vstupů je látkový tok vody, což mimo jiné umožňuje modelovat vliv diluce/koncentrace na acidobazickou rovnováhu..



Obrázek 2 – Acidobazická rovnováha je výslednicí dvou bilancí propojených přes pufrční systém – bilancí mezi tvorbou a vylučováním oxidu uhličitého a bilancí mezi tvorbou a vylučováním silných kyselin. Pufrční reakce samy o sobě nemění elektroneutralitu. Přesun iontů mezi kompartmenty tělních tekutin je vždy elektroneutralní, proto jsou přesuny bikarbonátů a vodíkových iontů vždy provázeny přesunem komplementárních iontů.

rovnovah v systému plazmy a dovoluje vypočítat pH plazmy z nezávisle proměnných: parciálního tlaku oxidu uhličitého pCO_2 , celkové plazmatické koncentrace fosfátů $[Pi]$, celkové koncentrace albuminů v plazmě $[Alb]$ a z hodnoty „strong ion difference“ (SID) – definované jako rozdíl koncentrací plně disociovaných kationtů (Na^+ , K^+ , Ca^{2+} , Mg^{2+}) a plně disociovaných aniontů (Cl^-), vyjádřený v mEq/l:

$$pH = \text{funkce} (pCO_2, SID, [Alb_{TOT}], [Pi])$$

K nevýhodám Stewartovy teorie patří to, že pracuje pouze s krevní plazmou. V plazmě platí, že hodnoty pCO_2 a SID jsou vzájemně nezávislé, v krvi to však neplatí, protože se změnou pCO_2 dochází k výměně bikarbonátů za (plně disociované) chloridové anionty a SID se proto mírně mění (Obrázek 1)

Krom toho, někteří následovníci Stewarta, fascinováni tím, že acidobazické parametry – pH (a příslušné koncentrace bikarbonátů, karbonátů, nebikarbonátových kyselin) lze vypočítat z nezávislých proměnných (pCO_2 , SID, $[Alb_{TOT}]$, $[Pi]$) nezřídka docházejí v jejich interpretaci k věcně nesprávným názorům.

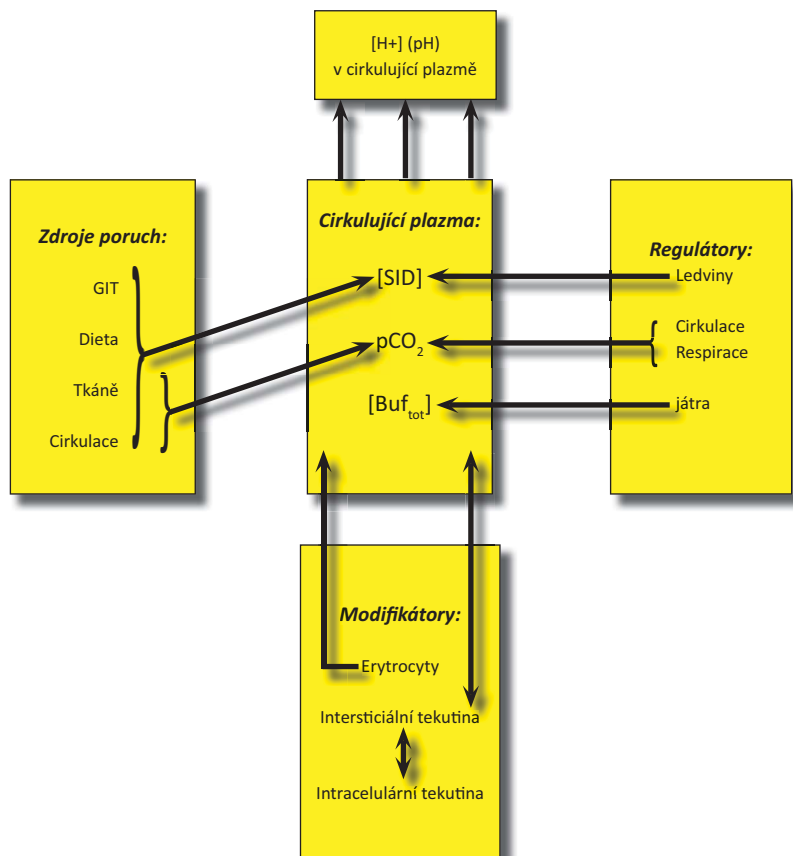
Nezávislost výchozích proměnných, především **SID** je při výpočtu míněna **nikoli v kauzálním, ale v striktně matematickém slova smyslu**. Ovšem v klinicko-fyziologické praxi se na to zapomíná, což často vede k nesprávnému výkladu kauzálního řetězce příčin acidobazických poruch.

Při modelování acidobazické rovnováhy jsme vycházeli z námi prosazovaného bilančního přístupu k acidobazické rovnováze^[18, 27, 28]. **Bilanční přístup**, který počítá s bilancí toku látek (Obrázek 2) je lepším východiskem pro modelování acidobazické rovnováhy vnitřního prostředí a přenosu krevních plynů než přístup dle Stewarta, zejména jedná li se o součást složitějšího modelu se zahrnutím regulace respirace a ledvin.

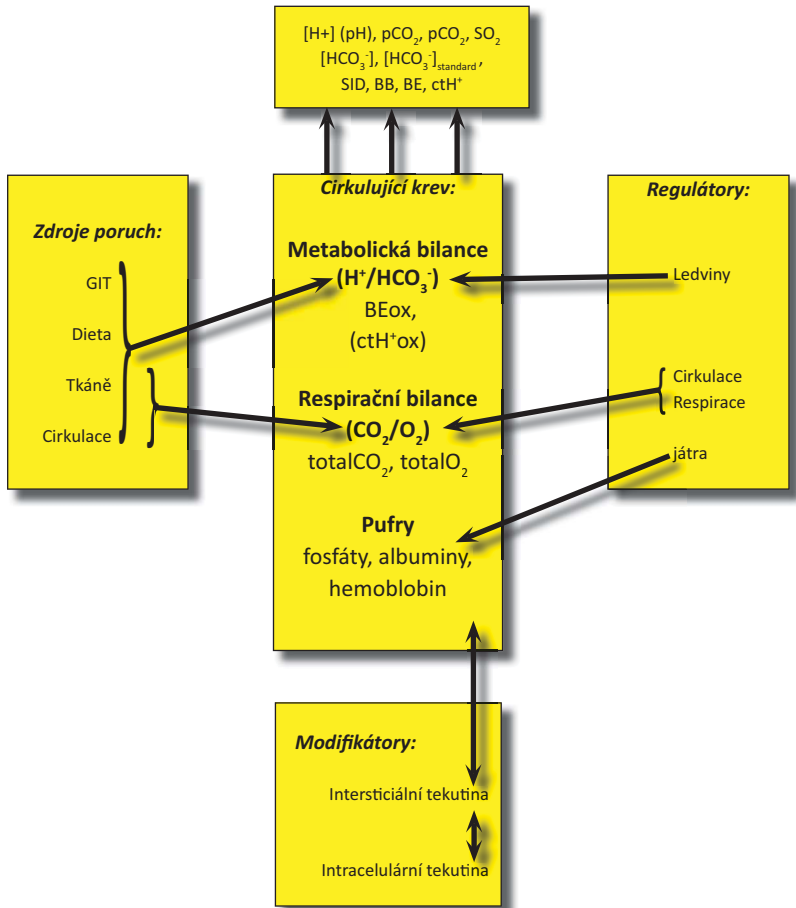
Problém modelování acidobazické rovnováhy vycházející ze Stewartova pojetí spočívá v tom, že je nutno přesně modelovat hodnotu SID, ovlivňovanou bilancí silně disociovaných kationtů a aniontů – v praxi to znamená, že přesnost modelování pH pak velmi silně závisí na přesnosti modelování bilance silně disociovaných kationtů a aniontů (tj. především iontů sodíkových, draselných, chloridových a sulfátových) – jejich příjmu a především vylučování v ledvinách. Drobné nepřesnosti, které vznikají při modelování příjmu a vylučování sodíku, draslíku a chloridů se pak odrážejí v nepřesnostech modelování acidobazického stavu (Obrázek 3).

To se právě projevuje v původní implementaci modelu HumMod. Přestože jeho autoři^[5, 6, 14, 15] v něm podstatně zlepšili přesnost modelování příjmu a vylučování sodíku, draslíku a chloridů v ledvinách, modelujeme-li dlouhodobý stav (kdy se s virtuálním pacientem nic neděje) má virtuální pacient v současné verzi modelu po měsíci simulovaného času tendenci upadat do lehké ustálené metabolické alkalózy.

Náš bilanční přístup (Obrázek 4) na rozdíl od Stewartova přístupu modeluje nikoli ovlivnění SID, ale skutečné metabolické bilance tvorby a vylučování silných kyselin (včetně acidifikačních procesů v ledvinách) .



Obrázek 3 – Sumarizovaný diagram regulace acidobazické rovnováhy podle Stewart. Organismus reguluje tři nezávislé proměnné, na nichž závisí závisle proměnná - pH plazmy. Obrázek převzat ze Stewartovy monografie (Stewart 1981). Zdánlivě jednoduché schéma však přináší problémy při modelování přenosu krevních plynů a acidobazické rovnováhy v organismu. Největší problém spočívá v požadavku na přesnost regulace hodnoty SID. Renální acidifikační procesy jsou modelovány nepřímou, přes renální regulaci hodnoty SID. Proto je zapotřebí s dostatečnou přesností modelovat vylučování silných kationtů (především sodíku a draslíku) a aniontů (především chloridů) a malá nepřesnost modelu se může projevit velkou chybou v acidobazické regulaci.



Obrázek 4 – Sumarizovaný diagram regulace acidobazické rovnováhy podle bilančního přístupu – organismus reguluje metabolickou a respirační bilanci, která je propojena přes pufrční systém. Metabolickou bilanci (tvorba a vylučování H^+/HCO_3^-) charakterizuje v krvi standardizovaná odchylka nárazníkových bází BEox (případně Siggaard-Andersenem prosazovaná hodnota parametru ctH⁺ ve virtuálně plně oxygenované krvi). Respirační bilanci charakterizují celkové koncentrace kyslíku a oxidu uhličitého v krvi (totalO₂ a totalCO₂), pufrční sílu krevních nárazníků ovlivňují koncentrace hemoglobinu, fosfátů a plazmatických albuminů. Z výše uvedených na sobě nezávislých proměnných je možno odvodit pO₂, pCO₂, pH, koncentrace aktuálních a standardizovaných bikarbonátů, saturaci hemoglobinu kyslíkem a hodnoty parametrů BE a SID..

Na rozdíl od amerických autorů v naší implementaci modelu QHP/HumMod – Golem Edition ^[28, 30] modelujeme skutečné metabolické bilance tvorby a vylučování silných kyselin

Je modelována metabolická produkce silných kyselin (v normě i patologii – např. zvýšená produkce kyseliny mléčné či ketokyselin) v tkáních – produkce a metabolická utilizace laktátu je uvažována v 12 různých tkáních. Zároveň je modelována i acidifikace v ledvinách, kde se uvažuje tvorba titrovatelné acidity a vylučování amonných iontů ^[28].

5. Stavové proměnné a ustálené stavy

Deklarativní objektový přístup při výstavbě modelu vedl k vysoce přehledné a modifikovatelné struktuře modelu – viz Obrázky 14–20 v přehledové přednášce ^[20] v tomto sborníku.

Objektový přístup nám umožnil dekomponovat model na jednotlivé univerzálně použitelné třídy a počítat model různými způsoby a směry. Když je např. nutné předefinovat směr výpočtu, je velmi jednoduché předefinovat výstupy za vstupy a změnu algoritmu výpočtu nechat na kompilátoru (pokud je to z hlediska matematických rovnic v pozadí modelu ovšem možné) ^[24, 25]. Je například možné pomocí jediného parametru přepínat mezi počítáním ustálených hodnot (steady state) a mezi klasickou simulací dynamického systému v čase.

Ustálený stav představuje řešení soustavy rovnic, které je v čase konstantní. To však nutně nemusí znamenat, že se omezujeme jen na děje, které nejsou oscilačního charakteru. Ustálené stavy oscilačního charakteru je možné vyjádřit časově konstantními hodnotami vyjadřujícími např. amplitudy, frekvence, délky periody, minimální hodnoty, maximální hodnoty nebo střední (mean) hodnoty.

V literatuře je popsáno i mnoho modelů vedoucích k nestálých chaotickým dějům (typu deterministického chaosu) v organismu. Většinou se ale jednalo o relativně jednoduché modely, popisující nějaký fyziologický subsystém. Za normálních podmínek lze ale zřejmě lidský organismus charakterizovat jako ultrastabilní systém s velkým množstvím velmi efektivních regulací, kde chaotické chování (např. fibrilace síní nebo komor srdečních) je poměrně vzácné. Je zajímavé, že chování rozsáhlého modelu tuto naši domněnku potvrzuje.

5.1 Implementace stavů

Velmi jednoduše lze implementovat přepínání výpočtu mezi ustáleným stavem stavové proměnné a dynamickým chováním. Použití operátoru derivace/integrace je implementováno pomocí abstraktní třídy **DynamicState**. To umožňuje přepnout všechny příslušné stavové proměnné počítané v ustáleném stavu (kdy je derivace nula) na stav dynamický. Stačí přepnout předvolenou hodnotu parametru STEADY na hodnotu „false“. Další výhodou

je možnost ponechání některých proměnných dynamickými i při výpočtu ustáleného stavu pomocí předdefinování hodnoty parametru STEADY v konkrétní instanci.

```

partial model DynamicState
  parameter Boolean STEADY = true;
  Real stateValue „state must be connected in inherited class definition“;
  Real changePerMin „dynamic change of state value per minute“;
equation
  if STEADY then
    changePerMin = 0;
  else
    der(stateValue) = changePerMin / Library.SecPerMin;
    //derivation time in Modelica is in seconds
  end if;
end DynamicState;

```

Ukažme si využití abstraktní třídy **DynamicState** v ovlivnění stavových proměnných v třídě charakterizující elastický kompartment v kardiovaskulárním systému.

Stavové proměnné v kardiovaskulárním systému často vyjadřují hromadění krve na určitých místech oběhového systému. Základní částí modelu jsou často instance třídy **VascularElasticBloodCompartment**, která vyjadřuje vztah mezi tlakem a naplněním určité části cévního systému.

Jako každý blok stavové proměnné i tato třída dědí abstraktní třídu DynamicState, což umožní s celým modelem (nebo s jeho celým odděleně modelovaným subsystémem) pracovat na velmi abstraktní úrovni. Drobnou nevýhodou je pouze ukrytí operátoru derivování/integrace ($der(stateValue)=change$) do abstraktní třídy **DynamicState**.

```

model VascularElasticBloodCompartment
  extends HumMod.Library.Utilities.DynamicState;
  extends HumMod.Blood.VascularCompartments.Interfaces.
    IVascularElasticBloodCompartment;
equation
  referencePoint.pressure = (max(Volume-V0,0)/
    Compliance)+ ExternalPressure;
  stateValue = Volume;
  changePerMin = referencePoint.q;
end VascularElasticBloodCompartment;

```

5.2 Problém inicializace stavových proměnných

Každá stavová proměnná vyžaduje počáteční hodnotu od níž začíná dyna-

mická simulace. Pokud jsou tyto hodnoty vzaty z ustáleného stavu simulace, tak pokračováním bude i následující ustálená simulace. Problémem však je, jak zjistit, že stav modelu je už po nějakém simulačním čase ustálený, a zda vůbec je ustálený. Jednou z možností, kterou nám dává deklarativní přístup v Modelice, je pokusit se spočítat stabilizovaný stav přímo – nastavíme systém tak že derivace stavových proměnných budou rovny nule. Tím se určí stavové proměnné jako konstanty a ze soustavy algebroidiferenciálních rovnic se stane nediferenciální algebraická soustava rovnic, která vyjadřuje určité možné řešení při dlouhodobé stabilizaci simulovaného pacienta. Je ale nutné si uvědomit, že vstupy do modelu/submodelu z okolí mají přímý vliv na optimální hodnoty stavových proměnných a naopak. Například stav hospodaření s živinami, elektrolyty nebo krevními plyny ovlivní hodnoty stavových proměnných – a neadekvátní přísun může vést k tomu, že ustálený stav ani nemusí existovat. Výjimkou není ani nejednoznačnost řešení.

Implementace ukládání a načítání hodnot proměnných z/do souboru umožňuje nad stejnými daty spouštět model po částech. To umožňuje jednoduché hledání problémů metodou oddělené simulace jednotlivých částí modelu.

Automatický výpočet ustálených hodnot je vhodné provádět při každé změně některého parametru modelu. Při nepřímém fitování modelu na experimentální data optimalizačními algoritmy je tedy vhodné před dynamickou simulací spustit statický výpočet ustáleného stavu. To pak zaručí pro daný odhad parametrů plynulý start dynamické simulace.

6. Závěr

Provedli jsme několik změn v modelu, zejména v problematice homeostázy vnitřního prostředí. Každá změna modelu ovšem vyžadovala příslušný přepočítání všech parametrů rozsáhlého modelu, které se mohou změnit. Pro hledání ustálených stavů modelu a určení počátečních podmínek jsme využívali deklarativní možnosti jazyka Modelica které nam umožnily lépe tyto ustálené stavy nacházet a nastavovat zohledňovat a přeshetřovat různá možná nastavení modelu.

Rozsáhlý model bude teoretickým základem pro vyvíjený lékařský výukový тренаžér. Při jeho tvorbě mimo jiné využijeme i námi vyvinutou technologii tvorby webových simulátorů [26].

Literatura

- [1.] Abram, S. R., Hodnett, B. L., Summers, R. L., Coleman, T. G., & Hester, R. L. (2007). *Quantitative circulatory physiology. An integrative mathematical model of human mathematical model of human physiology for medical education. Advanced Physiology Education, 31*, str. 202-210.
- [2.] Brugard, J., Hedberg, D., Cascante, M., Gedersund, G., Gómez-Garrido, A., Maier, D., Nyman, E, Selivanov, V & Stralfors, P. (2009). *Creating a Bridge between Modelica and the Systems Biology Community. Proceedings 7th Modelica Conference, Como, Italy, Sep. str. 20-22, 2009. Como: The Modelica Association.*

- [3.] Cellier, F. E., & Nebot, A. (2006). *Object-oriented modeling in the service of medicine. Proceedings of the 6th Asia Conference, Beijing, China 2006*. 1, str. 33-40. Beijing: International Academic Publishers.
- [4.] Cellier, F. E., & Nebot, A. (2006). *Object-oriented modeling in the service of medicine. Proceedings of the 6th Asia Conference, Beijing, China 2006*. 1, str. 33-40. Beijing: International Academic Publishers.15,16,17
- [5.] Coleman, T. G., Hester, R. L., & Summers, R. L. (2010). *Hummod model version 1.5*. [Online] <http://www.hummod.org>.
- [6.] Coleman, T. G., & Randall, J. E. (1983). *HUMAN. A comprehensive physiological model. The Physiologist*, 26, str. 15-21.
- [7.] Coleman, T. G., & Summers, R. L. (1997). *Using mathematical models to better understand integrative physiology. Journal of Physiology and Biochemistry*, 53, str. 45–46.
- [8.] Fencľ, J., Jabor, A., Kazda, A. & Figge, J. (2000). *Diagnosis of metabolic acid-base disturbances in critically ill patients. Am. J. Respir. Crit. Care*, 162, str. 2246-2251
- [9.] Fencľ V.& Leith D.E. (1993). *Stewart' s quantitative acid-base chemistry: applications in biology and medicine. Respir. Physiol.* 91, str. 1-16.
- [10.] Fencľ V.& Rossing T.H. (1989). *Acid-base disorders in critical care medicine. Ann Rev. Med.* 40, str. 17–20
- [11.] Fritzon, P. (2003). *Principles of object-oriented modeling and simulation with Modelica 2.1*. Wiley-IEE Press.
- [12.] Guyton, A. C., Coleman, T. G., & Grander, H. J. (1972). *Circulation: Overall regulation. Ann. Rev. Physiol.*, 41, str. 13–41.
- [13.] Haas, O. C., & Burnham, K. J. (2008). *Systems Modeling and Control Applied to Medicine*. V O. C. Haas, & K. J. Burnham, *Intelligent and Adaptive Systems in Medicine* (str. 17-52). Boca Raton FL, USA: CRC Press.
- [14.] Hester, R. K., Summers, R. L., Ilescu, R., Esters, J., & Coleman, T. (2009). *Digital Human (DH): An integrative mathematical model of human physiology. Proceedings of MOD-SIM World Conference NASA/CP-2010-216205* (str. 129-134). NASA.
- [15.] Hester, R. L., Coleman, T., & Summers, R. L. (2008). *A multilevel open source model of human physiology. The FASEB Journal*, 22, str. 756.
- [16.] Ikeda, N., Marumo, F., & Shirsataka, M. (1979). *A Model of overall regulation of body fluids. Ann. Biomed. Eng.*, 7, str. 135–166r
- [17.] Hester, R. L., Ilescu, R., Summers, R. L., & Coleman, T. (2010). *Systems biology and integrative physiological modeling. Journal of Physiology*, published ahead of print December 6, 2010, doi:10.1113/jphysiol.2010.201558, str. 1–17.
- [18.] Kofránek, J. (2009). *Komplexní model acidobazické rovnováhy. (Anglická verze: Complex model of acid-base balance je dostupná na adrese <http://www.physiome.cz/references/medsoft2009acidbase.pdf>, model je na adrese <http://www.physiome.cz/acidbase>).* V M. Zeithamlová (Editor), *MEDSOFT 2009* (str. 23–60). Praha: Agentura Action M.
- [19.] Kofránek, J. (2009). *What is behind the curtain of a multimedia aducational games? EA-TIS 09 Contribution Proceedings. Euro American Conference on Telematics & Information Systems, 2009.* (str. 229-236). Praha: Wirelesscom sro., ISBN 978-80-87205-07-5. *Práce je dostupná na adrese <http://www.physiome.cz/references/eatis2009.pdf>.*

- [20.] Kofránek, J. *Komplexní modely fyziologických systémů jako teoretický podklad pro výukové simulátory*. Ibidem, 2011.
- [21.] Kofránek, J., Anh Vu, L. D., Snášelová, H., Kerekeš, R., & Velan, T. (2001). *GOLEM – Multimedia simulator for medical education*. V L. Patel, R. Rogers, & R. Haux (Editor), *MEDINFO 2001, Proceedings of the 10th World Congress on Medical Informatics*. 1042-1046. London: IOS Press.
- [22.] Kofránek, J., Andrlík, M., Kripner, T., & Mašek, J. (2002). *Simulation chips for GOLEM – multimedia simulator of physiological functions*. V J. G. Anderson, & M. Kapzer (Editor), *Simulation in the Health and Medical Sciences 2002*. (str. 159-163). San Diego: Society for Computer Simulation International, Simulation Councils.
- [23.] Kofránek, J., Mateják, M., Privitzer, P., & Tribula, M. (2008). *Causal or acausal modeling: labour for humans or labour for machines*. V C. Moler, A. Procházka, R. Bartko, M. Folin, J. Houška, & P. Byron (Editor), *Technical Computing Prague 2008, 16th Annual Conference Proceedings*, CD ROM (str. 058_kofranek.pdf: 1-16). Praha: Humusoft s.r.o. Práce je dostupná na adrese <http://www.physiome.cz/references/TCP2008.pdf>.
- [24.] Kofránek, J., Mateják, M., & Privitzer, P. (2009). *Leaving toil to machines - building simulation kernel of educational software in modern software environments*. CD ROM. V L. Dušek, D. Schwarz, & S. Štípek (Editor), *Mefanet 2009, Conference Proceedings* (str. kofranek.pdf: 1-39). Brno: Masarykova Univerzita. Práce je dostupná na adrese <http://www.physiome.cz/references/MEFANET2009.pdf>.
- [25.] Kofránek, J., Privitzer, P., Mateják, M., Tribula, M. (2009) *Akuzální modelování – nový přístup pro tvorbu simulačních her*. In *MEDSOFT 2009*. In Ziethamlová (Editor) Praha: Agentura Ac2tion M, Praha 2008, str. 31–75.
- [26.] Kofránek, J., Mateják, M., & Privitzer, P. (2010). *Web simulator creation technology*. *Mefanet Report*, 3, str. 32–97. Práce je dostupná na adrese <http://physiome.cz/references/mefanetreport03.pdf>.
- [27.] Kofránek, J., Matoušek, S., & Andrlík, M. (2007). *Border flux ballance approach towards modelling acid-base chemistry and blood gases transport*. V B. Zupanec, S. Karba, & S. Blažič (Editor), *Proceedings of the 6th EUROSIM Congress on Modeling and Simulation, Full Papers* (CD) (str. TU-1-P7-4: 1-9). Ljubljana: University of Ljubljana. Práce je dostupná na adrese <http://www.physiome.cz/references/ljubljana2007.pdf>.
- [28.] Kofránek, J., Matoušek, S., Mateják, M. (2010). *Modelování acidobazické rovnováhy*. *Medsoft*, 2010, sborník příspěvků, Zeithamlová, M. (Ed.), ISSN 1803-8115, *Creative Connections*, Praha 2010, str. 66-80..
- [29.] Logan, J. D., & Wolesensky, J. D. (2009). *Mathematical methods in biology*. Hoboken, NJ: John Wiley & Sons, Inc.
- [30.] Mateják, M., Kofránek, J. (2010). *Rozsáhlý model fyziologických regulací v Modelice*. *Medsoft*, 2010, sborník příspěvků, Zeithamlová, M. (Ed.), ISSN 1803-8115, *Creative Connections*, Praha 2010, str. 126–146.
- [31.] McLeod, J. (1967). *PHYSBE... a year later*. *Simulation*, 10, str. 37–45.
- [32.] McLeod, J. (1966). *PHYSBE: A ophysiological simulation benchmark experiment*. *Simulation*, 15, str. 324–329.

- [33.] McLeod, J. (1970). *Toward uniform documentation-PHYSBE and CSMP. Simulation*, 14, str. 215–220
- [34.] Oomnes, C., Brekleman, M., & Baaijens, F. (2009). *Biomechanics: concepts and computation*. Cambridge: Cambridge University Press.
- [35.] Siggaard-Andersen, O., Wimberley, P.D., Fogh-Andersen, N., Gøthgen I.H. (1988): *Measured and derived quantities with modern pH and blood gas equipment: calculation algorithms with 54 equations. Scand J Clin Lab Invest.* 48, Suppl 189, str. 7–15.
- [36.] Siggaard-Andersen, M., & Siggaard-Andersen, O. (1995). *Oxygen status algorithm, version 3, with some applications. Acta Anaesth Scand*, 39, Suppl 107, str. 13–20.
- [37.] Stewart, P., A. (1981): *How to understand acid-base. A Quantitative Primer for Biology and Medicine*. New York: Elsevier, New York, 1981. ISBN 0444004068, druhé vydání (2009) in Kellum J. A., Elbers Paul. W. G.,(eds) *Stewart's textbook of Acid-Base, AcidBase.org*, Amsterdam, 2009, ISBN 978-1-4092-5470-6, str. 29–197.
- [38.] Stewart, P. A. (1983). *Modern quantitative acid-base chemistry. Can. J. Physiol. Pharmacol.*, 61, str. 1444-1461.
- [39.] Thomas, R. S., Baconnier, P., Fontecave, J., Francoise, J., Guillaud, F., Hannaert, P., a další. (2008). *SAPHIR: a physiome core model of body fluid homeostasis and blood pressure regulation. Philosophical Transactions of the Royal Society*, 366, str. 3175–3197.
- [40.] Wallish, P., Lusignan, M., Benayoun, M., Baker, T. I., Dickey, A. S., & Hatsopoulos, N. G. (2008). *MATLAB for Neuroscientists: An Introduction to Scientific Computing in MATLAB*. Burlington, MA: Academic Press.

Poděkování

Model bude základem lékařského výukového trenažéru vyvíjeného v rámci projektu MPO FR-TI3/859. Využití optimalizačních technik pro nepřímý odhad parametrů modelu bylo financované z Fondu rozvoje CENET 2009/2 jako projekt „Efektivní identifikace fyziologických systémů ve výpočetním gridu“.

Kontakt:

MGr. Marek Mateják

MUDr. Jiří Kofránek, CSc.

Oddělení biokybernetiky a počítačové podpory výuky

Ústav patologické fyziologie 1.LF UK a VFN

Praha, U nemocnice 5, 128 53 Praha 2

tel: +420 224965912

e-mail: matejak.marek@gmail.com

e-mail: kofranek@gmail.com