

ZÁPADOČESKÁ UNIVERZITA V PLZNI
FAKULTA APLIKOVANÝCH VĚD
KATEDRA INFORMATIKY A VÝPOČETNÍ TECHNIKY

Diplomová práce

Metoda odhadu chyby výpočtu koncentrace glukózy

Plzeň, 2015

Jan Strnádek

Obsah

1	Úvod	6
1.1	Diabetes	6
1.1.1	Symbol diabetu	7
1.2	Glukóza a její distribuce	7
1.3	Inzulin a glukagon	8
1.3.1	Inzulin	8
1.3.2	Glukagon	9
1.4	Typy diabetu	9
1.4.1	Typ 1	9
1.4.2	Typ 2	10
1.5	Akutní komplikace	10
1.5.1	Hyperglykémie	10
1.5.2	Hypoglykémie	10
1.6	Komplikace	11
1.6.1	Kardiovaskulární onemocnění	11
1.6.2	Onemocnění ledvin	12
1.6.3	Onemocnění nervů	12
1.6.4	Onemocnění zrakového ústrojí	12
1.6.5	Komplikace v těhotenství	12
1.7	Příznaky	13
1.8	Podpurná léčba diabetu	13
1.8.1	Transplantace	14
1.8.2	Umělá slinivka	14
1.8.3	Prevence	15
2	Metoda odhadu chyby	16
2.1	Metrika	16
2.2	Model Steil-Rebrinové	16
2.3	Auto-regresivní model a Kalmanův filtr	17
3	Model Steil-Rebrinové	19
3.1	Rozklad na přeurčený systém rovnic	19
3.1.1	Numerická derivace	20
3.1.2	NLS - Nelineární metoda nejmenší čtverců	20
3.1.3	Rychlost výpočtu NLS	27
3.1.4	Porovnání s matematickým SW	30

4	Optimalizace pářením včelí královny	32
4.1	Genetické algoritmy	32
4.2	Algoritmus optimalizace pářením včelí královny	33
4.3	Predikční model distribuce glukózy	33
4.4	Použití HBMO při predikci hladiny glukózy	34
4.4.1	Zdatnostní funkce	34
4.4.2	Počáteční inicializace algoritmu	34
4.4.3	Svatební lety	34
4.4.4	Křížení genotypů	35
4.4.5	Dělnice	35
4.5	Problém lokálních minim	35
4.5.1	Přidání věku řešení	36
4.5.2	Vliv na řešení	36
4.5.3	Vliv iterací a nových mutací před a po přidání stáří řešení	37
4.5.4	Vliv přidání stáří řešení na výsledky algoritmu	38
5	AR model a Kalmanův filtr	39
5.1	Použití	39
5.2	RLS	40
5.2.1	Iterační rovnice RLS	40
5.2.2	Rozdíl mezi NLS a RLS	41
5.3	Kalmanův filtr	41
5.3.1	Fáze filtrování	42
5.3.2	Aplikace KF	43
5.4	Auto-Regresivní model	44
5.4.1	Aplikace AR modelu	44
5.5	Problém predikce z dat s příliš častým vzorkováním	45
6	Navržená metrika	46
6.1	Matematické vyjádření	46
6.2	Znázornění grafem	46
6.3	Numerická integrace	47
7	Programová dokumentace	48
7.1	Matice	48
7.1.1	Výhody vlastní implementace	48
7.2	Implementace jednotlivých modelů	49
7.2.1	Steil-Rebrin	49
7.2.2	HBMO	49
7.2.3	AR + Kalman	50

7.3	Vlastní metrika	50
7.4	Kompilace	50
7.5	Spuštění	51
8	Výsledky	52
8.1	Měřená data	52
8.2	Vysvětlení grafů	52
8.3	Modely koncentrace hladiny v krvi	53
8.3.1	Výsledky aplikace modelu Steil-Rebrinové	53
8.3.2	Výsledky optimalizace páření včelí královny	54
8.4	Modely predikce hladiny v intersticiální tekutině	55
8.4.1	Výsledky Kalmanův filtr a auto regresivní model	55
8.4.2	Výsledky difuse modelu	55
9	Závěr	58
9.1	Referenční stroj	58
9.2	Obsah CD	58
9.3	Závěr a porovnání metod	59
9.3.1	Porovnání metod pro koncentraci glukózy v krvi	59
9.3.2	Porovnání metod pro predikci hladiny glukózy v intersticiální tekutině	59
9.3.3	Rychlost výpočtu	59
9.3.4	Výsledky metriky	59
	Použitá literatura a zdroje	60
	Seznam obrázků	63
	Seznam tabulek	64

Poděkování

Rád bych touto cestou poděkoval Ing. Tomáši Koutnému, Ph.D. za odborné vedení a podnětné rady k diplomové práci.

Čestné prohlášení

Prohlašuji, že svou diplomovou práci na téma „Metoda odhadu chyby výpočtu koncentrace glukózy“ jsem vypracoval samostatně pod vedením vedoucího diplomové práce a s použitím odborných zdrojů a literatury, které jsou citovány a uvedeny v seznamu literatury, popřípadě u některých zdrojů přímo v textu. Jako autor práce dále prohlašuji, že si nejsem vědom, že bych v souvislosti s jejím vytvořením porušil autorská práva třetích osob.

V Plzni dne 14.5.2015

Abstrakt

English

Diabetes mellitus is an incurable autoimmune disease that affects 8.3 % of the world population. The disease manifests itself mostly as a malfunction causing pancreas to produce too little insulin or none at all, or the insulin might be produced, but the correct response is not triggered by the organism. Therefore a diabetic type-1 patient has to dose insulin as precisely as possible to avoid serious effects which may lead to death. This thesis deals with the first type of diabetes, that is such a case where the patient delivers insulin based on self-assessed measurement of the insulin blood levels, on a daily condition or according to previous food intake. With an advanced development of the disease, the patient starts using an insulin pump which gathers data from a device for continuous glucose levels measurement. It is therefore necessary to find the relationship between the measured values from interstitial fluid and the actual glucose levels in blood. Finding the correct value would require to model all metabolic paths in the entire body individually for each patient, but it's not possible now, hence we attempt to find the closest estimate instead. This thesis implements a number of methods regarding the prediction of levels in the interstitial fluid of subcutaneous tissue and the reconstruction of blood glucose levels. The efficiency and errorness of suggested methods are discussed as well.

Česky

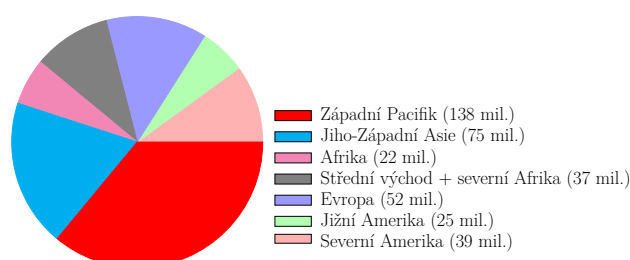
Diabetes mellitus, česky úplavice cukrová, je velmi rozšířené nevléčitelné autoimunitní onemocnění, které postihuje 8.3 % světové populace. Původem nemoci je buď nefunkčnost slinivky břišní, která produkuje málo inzulínu, nebo když je produkován a tělo na něj nereaguje. Při tomto onemocnění je nutné přesné dávkování inzulínu, aby se předešlo závažným komplikacím. Tato práce se zabývá diabetem prvního typu, kde si pacient dává inzulín podle samoměření krve a podle toho, co jedl nebo jak se cítí. Při velmi pokročilém rozvoji nemoci má inzulínovou pumpu, která se řídí údaji z přístroje pro kontinuální měření hladiny cukru z podkoží. Je proto nutné nalézt vztah mezi naměřenými hodnotami z podkoží a skutečnou hladinou glukózy v krvi. Aby tento vztah byl naprosto přesný, museli bychom modelovat celé tělo a to pro každého, což zatím není možné, a proto se snažíme nalézt pouze optimální odhad. Tato práce implementuje několik metod, pro predikci koncentrace glukózy v intersticiální tekutině podkoží a rekonstrukci koncentrace glukózy v krvi, dále se zabývá jejich efektivitou a chybovostí.

Kapitola 1

Úvod

1.1 Diabetes

Diabetes mellitus¹ (dále DM) je chronické onemocnění, při kterém slinivka břišní není schopná vyrábět dostatek inzulinu nebo ho tělo nedokáže zpracovat, následkem toho je zvýšená hladina cukru v krvi (hyperglykémie). Cukrovka může mít obecně i jiné příčiny, které nejsou ještě úplně objasněné. Ze známých příčin jmenujme například kontraindikace léků, stres, zranění... Dle IDF² [6] [1] mělo v roce 2014 diabetes 8,3 % světové populace (38,8 milionu lidí). Odhaduje se nárůst v takové míře, že v roce 2035 bude počet diabetiků okolo 243,8 milionu, jednotlivé zastoupení výskytu diabetu na jednotlivých kontinentech můžete vidět v grafu 1.1 [1].



Obrázek 1.1: Výskyt diabetu napříč jednotlivými kontinenty
(počty jsou v milíonech lidí)

Dalším zajímavým faktem je, že zhruba polovina lidí trpící diabetem o svém zdravotním stavu není obeznámena. Onemocnění se vyvíjí pozvolna, a může být zjištěno obvodním doktorem až po čase při preventivní prohlídce rozbořením krve, přičemž ani tímto není výsledek zaručen. Doktor by musel provádět měření ve chvíli, kdy je hodnota glukózy v krvi zvýšena. Aby byly výsledky opravdu relevantní, bylo by nutné aby po dobu několika dní zaznamenávat hladiny glukózy, třeba pomocí CGMS přístroje. Každý člověk má určitou hladinu cukru v krvi, protože cukr je zdrojem energie. Běžná hodnota

¹česky úplavice cukrová

²internation diabetes federation

hladiny cukru v krvi kolísá u zdravého člověka mezi 3,5 – 6,5 mmol/l krve. Více než 7 mmol/l je signálem diabetu. Nejznámější typy DM jsou typ 1 a typ 2 (viz kapitola 1.4 *Typy diabetu*).

1.1.1 Symbol diabetu

Na obrázku 1.2 můžeme vidět *Modrý kruh*, mezinárodní symbol diabetu od roku 2007, jehož modrá barva symbolizuje oblohu a současně je znakem Organizace spojených národů. Kruh má symbolizovat jednotu, protože diabetes je jedinou neinfekční chorobou, která je považována za celosvětovou hrozbu.



Obrázek 1.2: Mezinárodní symbol diabetu

1.2 Glukóza a její distribuce

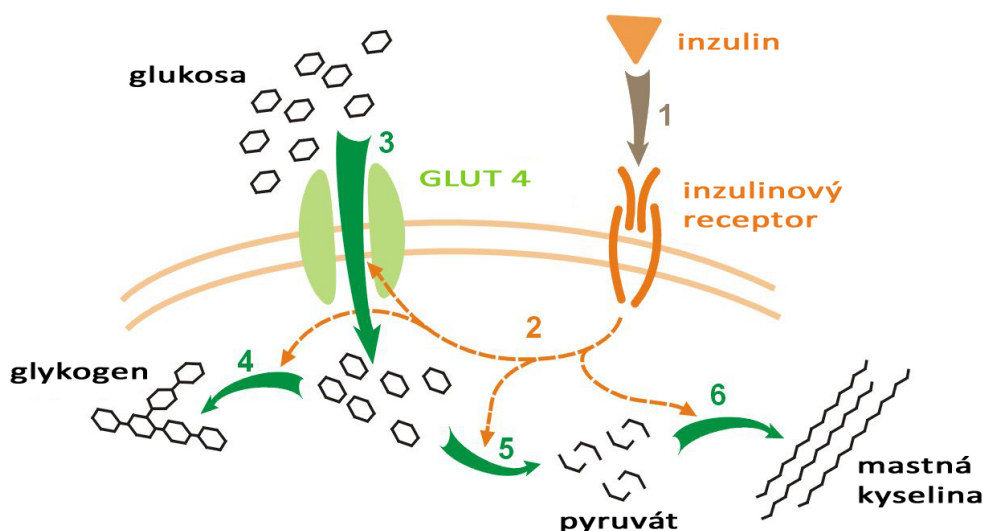
Glukóza ($C_6H_{12}O_6$) je monosacharid, který je pro všechny buňky energetickým substrátem, podílející se na intermediárním metabolismu, tj. na vzájemné přeměně sacharidů, lipidů (tuky a oleje) a proteinů. Do těla je přijímána potravou, buď volně (ovoce nebo jiné rostliny, jídla obsahující cukry - zákusky nebo i třeba oslazená káva) nebo součástí disacharidů a polysacharidů, kdy je v ústech potrava natrávena a pokračuje přes žaludek až do tenkého střeva, kde se sacharidy mění na monosacharidy a vstřebávají se přes střevní stěnu do krve. Glukóza je primárně distribuována prostřednictvím krevního oběhu k buňkám, kde je transportována skrz krevní kapiláry do intersticiální tekutiny, více o přeměně glukózy na energii bude v kapitole 1.3.1 *Inzulín*. Zajištění její normální hladiny provádí síť hormonů, nervových signálů a dalších. Hladina glukózy v intersticiální tekutině reflektuje změny hladiny v krvi s dodatečným zpožděním, které je problém určit, protože lidské tělo je velice komplikované a není zatím v lidských silách ani stavu poznání, modelovat chování celého lidského těla.

1.3 Inzulin a glukagon

Budeme-li mluvit o glukóze a jejím významu v metabolismu lidského těla, musíme se zmínit o nejvýznamnějších hormonech inzulínu a glukagonu, které s glukózou v lidském těle souvisí a udržují její normální hladinu.

1.3.1 Inzulin

Inzulin je hormon produkovaný B buňkami Langerhansových³ ostrůvků slinivky břišní. Svalové a tukové buňky potřebují inzulín k utilizaci glukózy, které tak snižují její hladiny v krvi. Inzulin je do krve vylučován zejména podle aktuální hladiny glykémie. Zhruba 50% inzulínu je denně vyloučeno nezávisle na potravě, zbytek je stimulován příjmem potravy. Množství inzulínu, které je vylučováno nezávisle na potravě je přímo úměrné obsahu cukru v krvi, tzn. čím více cukru v krvi tím více inzulínu. Na obrázku 1.3 vidíme, jak inzulín po navázání na inzulínový receptor buněk reaguje s glukózou.



Obrázek 1.3: Metabolismus glukóza - glykogen
(zdroj: *Wikimedia Commons pod licencí public domain*)

³Podle Paula Langerhansse, který je jako student Berlínské univerzity v roce 1869 objevil.

Glykogen

Glykogen je zásobní polymer, který si lze představit jako pospojované molekuly glukózy. Ukládá se v játrech a v menší míře ve svalovině. Pokud je v těle glykogenu nedostatek, je rozložen na glukózu. Glukóza z jaterního glykogenu je distribuována krví k buňkám, které jsou na něm závislé a nemají v danou chvíli jiný zdroj paliva - např. mastné kyseliny/tuky pro fyzické zátěži nad cca 20 minut.

1.3.2 Glukagon

Glukagon je produkován alfa buňkami slinivky, jakmile detekují nízkou hladinu glykémie. Při detekci glukagonu v krvi játra začnou rozkládat glykogen na glukózu a vypouštět jí do krevního oběhu. Tento proces rozkladu se označuje jako *glykogenolýza*. Glukagon stejně jako Inzulin se podává jako podpůrná léčba diabetu (viz kapitola 1.8 *Podpůrná léčba diabetu*), ale zatímco inzulin je podáván kontinuálně, glukagon se podává pouze při silné hypoglykémii, kdy pacient není schopen nic jíst a tudíž jeho tělo nemůže získat glukózu z jídla.

1.4 Typy diabetu

Zaměříme se na dva nejznámější typy diabetu:

1.4.1 Typ 1

Typ 1, na který se primárně orientují všechny modely uvedené v této práci, je způsoben nedostatečnou produkcí inzulinu (hormonem slinivky břišní). Laicky řečeno imunitní systém, který chrání a zajišťuje integritu organismu likvidací cizích či vlastních potenciálně škodlivých struktur, se dostane do fáze *Autoimmunity*. Autoimunita je stav, kdy imunitní systém začne reagovat na vlastní „bezpečné“ struktury organismu a tím je poškozují. Konkrétně u diabetu ničí imunitní systém buňky slinivky břišní, které produkují inzulin. Inzulin je pro lidské tělo životně důležitou součástí, umožňuje utilizaci glukózy buňkami. Glukóza je energetické palivo pro všechny buňky, bez energie buňky zahynou a následně i jejich hostitel. Diabetes prvního typu se objevuje v dětství či dospívání. V individuálních případech je možný i pozdní vznik onemocnění po třicátém roce věku. Ten je označován jako LADA (*latent autoimmune diabetes in adults*).

1.4.2 Typ 2

Typ 2 je zapříčiněn sníženou citlivostí tkání těla k inzulínu. Slinivka u pacientů postižených typem 2, tedy produkuje dost až nadbytek inzulínu, ale tělo na něj nereaguje. Na rozdíl od typu 1 se vyskytuje především u dospělých a jedinců trpících obezitou nebo formou nadváhy. Většinou propuká pozvolna po 30. roce života, ovšem poslední dobou se objevuje i u dětí s akutním nedostatkem pohybu (*vlohy pro typ 2 se přenášejí i dědičně*).

1.5 Akutní komplikace

Diabetes jakožto autoimunitní onemocnění způsobuje akutní i chronické komplikace. Mezi ně patří hyperglykémie, hypoglykémie a jiné komplikace spojené např. se srdeční činností, zrakem, trávením apod.

1.5.1 Hyperglykémie

Hyperglykémie je způsobena vysokou hladinou glukózy v krvi – přesáhne-li 11 mmol/l (jak již bylo zmíněno a znovu pro představu: normální hodnota se pohybuje mezi 3,5 až 5,6 mmol/l). Vzniká nedostatkem inzulínu, ke kterému může dojít z těchto příčin:

- vynechaná dávka inzulínu,
- nedostatečná dávka bolusu⁴ k jídlu nebo
- vytažení / porušení kanily k inzulínové pumpě.

1.5.2 Hypoglykémie

Hypoglykémie je způsobena nízkou hladinou glukózy v krvi pod 3,3 mmol/l a vzniká většinou z těchto příčin:

- nadměrná fyzická zátěž,
- stres,
- alkohol,
- opomenutí dávky jídla nebo při příliš velkém příjmu potravy, kdy se vyprodukuje příliš velké množství inzulínu, nebo

⁴z lat. *jednorázová dávka k jídlu (doplněk stravy pro diabetiky)*

- požití některých léků.

Pokud je tento stav dlouho ignorován nebo nerozpoznán, může vést až k trvalému poškození mozku. Hlavní příznaky hypoglykémie bývají:

- bolesti hlavy,
- pocení,
- třes,
- únava a bledost,
- pálení či píchání rtů, jazyka, prstů,
- ztráta vědomí
- ...

Vždy je nutné vyhledat pomoc lékaře!

1.6 Komplikace

Lidé s diabetem mají zvýšené riziko vzniku značného množství vážných zdravotních komplikací. Konzistentní zvýšená hladina glukózy v krvi vede k vážným problémům se srdcem, cévami, zrakovým ústrojím, ledvinami, nervy a zuby. Nejvíce jsou však zastoupeny kardiovaskulární choroby, selhání zrakového ústrojí, selhání ledvin a často je nutná amputace části nohy.

1.6.1 Kardiovaskulární onemocnění

Tato onemocnění postihují srdce a krevní kapiláry a mohou způsobit fatální komplikace jako například onemocnění koronární artérie⁵, což může vést až k infarktu nebo mrtvici. Kardiovaskulární onemocnění jsou nejčastějším důvodem úmrtí pacientů s diabetem. Ke zvýšení rizik kardiovaskulárních komplikací samozřejmě přispívají i další důvody, mezi něž patří například zvýšený krevní tlak, zvýšená hladina cholesterolu a jiné.

⁵Tzv. *věnčité tepny*, které přivádí krev do srdeční svaloviny a vyživují ji.

1.6.2 Onemocnění ledvin

Chronické onemocnění ledvin nazývané *diabetická nefropatie* se objevuje v důsledku dlouhodobě zvýšené glukózy v krvi. Způsobuje nižší účinnost ledvin a v některých případech vede až k jejich úplnému selhání. Onemocnění ledvin je mnohem častější u lidí s diabetem než u zdravé populace, proto tomuto onemocnění může zdravý člověk částečně předejít (respektive snížit riziko jeho vzniku) udržováním normální úrovně hladiny glukózy v krvi.

1.6.3 Onemocnění nervů

Diabetická neuropatie, jak se toto onemocnění nazývá, je nezánetlivé poškození funkce a struktury periferního nervstva, které je rovněž způsobeno dlouhodobě zvýšenou hladinou glukózy v krvi a zvýšeným krevním tlakem. Tato chronická komplikace je u pacientů s diabetem nejčastější, ale jak již bylo zmíněno, není nejčastější příčinou úmrtí. Diabetická neuropatie může zasáhnout jak nervy motorické, senzitivní, tak i vegetativní (*pozn.: vegetativní nervy ovládají například pocení nebo pohyb střev; naše vědomí na ně nemá vliv*).

Autonomní neuropatie

Tzv. syndrom *dead-in-bed*, kdy diabetik zdánlivě bez příčiny umře během spánku. Tato komplikace bývá často bez jakýchkoliv příznaků a tudíž není diagnostikována ani léčena.

1.6.4 Onemocnění zrakového ústrojí

Diabetická retinopatie je označení pro nejčastější patologické onemocnění sítnice u diabetiků. Objevuje se v průběhu deseti let od propuknutí diabetes, avšak doba vzniku onemocnění je zcela individuální. U léčených diabetiků se může projevit později, ale rovněž i dříve. Diabetická retinopatie je velice závažný zdravotní problém vedoucí k *snížení zrakové ostrosti*, v nejhorším případě i *slepotě*. Slepota nastává zhruba u 2 % [2] diabetiků, a je tak nejčastější příčinou slepoty v průmyslově vyspělých zemích.

1.6.5 Komplikace v těhotenství

Schopnost otěhotnět je u diabetiček nevýznamně menší, než u zdravých žen. Se správně léčeným a stabilizovaným diabetem je šance na porod zdravého dítěte téměř stejná jako u žen bez diabetu, přesto jsou zde ale větší rizika komplikací v průběhu těhotenství i porodu. Poporodní nemocnost u dětí je

zhruba 2-3x vyšší [9]. Dědičnost diabetu je závislá na rodičích. Pokud má diabetes matka, je pravděpodobnost výskytu diabetu u dítěte okolo 2 %, jestliže otec, je riziko 5-6 %, pokud oba, pak riziko dědičného přenosu dosahuje 10-15 %. Tzv *těhotenskou cukrovku* může mít i nediabetička, u které se zjistí zvýšená hladina glukózy až při vyšetření v těhotenství. Většinou je tento stav dočasný a po porodu se hladiny vrací k normálu. I přesto ženy s tímto problémem chodí na pravidelné prohlídky. Pokud není těhotenská cukrovka objevena a léčena, je velká možnost akutních komplikací po porodu nebo zvýšené riziko cukrovky druhého typu u dítěte. Tomu však lze předcházet (viz kapitola 1.8.3 *Prevence*).

1.7 Příznaky

Příznaky cukrovky jsou ze začátku nerozpoznatelné, zvláště u typu 1, než imunitní systém zničí alespoň 80% beta buněk slinivky břišní a tělo začne pociťovat nedostatek inzulínu, v tu chvíli již lze cukrovku rozpoznat. Diabetes typu 2 vzniká ještě mnohem pomaleji. Častými příznaky bývají:

- **Časté močení a vysoký příjem tekutin** - zvýšenou hladinu glukózy v krvi nezvládnou ledviny korigovat a proto jej propouštějí do moči, glukóza na sebe váže molekuly vody.
- **Ztráta váhy** - nedostatek inzulínu dostatečně nerozkládá glukózu v krvi a proto tělo využívá zásoby energie v tucích.
- **Únava a letargie** - jsou značně spojeny se ztrátou váhy, protože tělo z tuku nedokáže získat dostatečné množství energie.
- **Ztráta vědomí** - čím více nemoc propuká u pacienta a není léčena, tím víc se nám pacient může zdát apatický, až „opilý“. Pokud tento stav trvá déle vede ke ztrátám vědomí, až k *diabetickému kómatu* a smrti.

1.8 Podpůrná léčba diabetu

Veškerá léčba je závislá na včasnosti objevení choroby, pokud je objevena v raném stádiu, dostává se pacient do péče imunologa, který se pokusí léky zastavit imunitní systém, aby neničil dále poškozené buňky slinivky. Pokud není postup možné zastavit, nebo nebyl zastaven včas imunitní systém zničí veškeré beta-buňky a pacientovi nezbývá nic jiného, než užívat do konce života inzulín. Proto je následně nutná *kompensace* diabetu a to správným

stravováním, sportem, vyhýbáním se stresových situací a co nejpřesnějším dávkováním inzulínu. Pokud pacient toto všechno zvládne, má velkou šanci, že nebude žít výrazně kratší život než nediabetická část populace.

1.8.1 Transplantace

Jedinou možností, jak v dnešní době úplně vyléčit cukrovku typu 1. je transplantace slinivky břišní nebo transplantací Langerhansových ostrůvků, které inzulín vytvářejí. Ovšem při obou případech pacient bude muset trvale užívat *imunosupresiva*, které potlačují imunitní systém, aby neodmítl nově transplantovanou tkáň. Ovšem tyto léky mají značné vedlejší účinky a proto se k transplantaci přistupuje pouze v případech, kdy cukrovka je přímo život ohrožující a není další možnost.

1.8.2 Umělá slinivka

Umělá slinivka je velkou nadějí pro všechny diabetiky v současné době se zabývá vývojem umělé slinivky ve světě přes 25 týmů [13]. Tento přístroj je složen z:

- **inzulínové pumpy**, která dodává potřebné množství inzulínu do lidského těla,
- **CGM⁶** - senzor pro měření hladiny glukózy v podkoží,
- **softwarového vybavení**, které umožňuje predikovat hladinu glukózy v krvi s dostatečnou rezervou.

Výzkumy ukazují, že umělá slinivka je schopná dávkovat inzulín lépe než průměrný diabetolog či pacient[3]. Jedním z významných předních vědců pracujících na umělé slinivce je i český rodák Roman Hovorka z University of Cambridge. Predikční software funguje na základě učení, což znamená, že na začátku testuje, jak bude organismus diabetika reagovat na dávky inzulínu a měří hodnoty. Z nich následně dává dávku inzulínu, již správně. Nástroj pro predikci hladiny glukózy provádí značné výpočty a hodně často, proto u velké části projektů využívá „chytrého“ mobilního telefonu, který má každý u sebe, na kterém je tento software nainstalován a komunikuje přes rozhraní bluetooth s CGM a inzulínovou pumpou.

⁶z angl. *continuous glucose monitoring* volně přeloženo jako: kontinuální měření glukózy

1.8.3 Prevence

Diabetu prvního typu není možné předejít, jeho vývoji předchází řada faktorů, které nejsme schopni ovlivnit, ovšem diabetu druhého typu může předejít každý. Mezi nejdůležitější faktory patří:

- pohyb,
- vyvážená zdravá strava,
- správná váha,
- nekouřit - kouření zvyšuje šanci chronických onemocnění včetně diabetu,
- nadměrná konzumace alkoholu,
- vyhnout se stresovým situacím,
- správná délka spánku - jak lidé mající nadbytek tak nedostatek spánku mají zvýšenou šanci cukrovky 2. typu.

Mezinárodní diabetická organizace IDF[1] uvádí, že 30 minut sportu každý den snižuje riziko diabetu druhého typu o 40%.

Kapitola 2

Metoda odhadu chyby

Protože aproximujeme reálná data modelem (funkcí) potřebujeme vědět, jak moc přesná aproximace je. K tomu slouží několik postupů, ty co vycházejí z modelů a jsou použité v této práci, si v této kapitole shrneme. Jejich konkrétní použití bude pro přehlednost vysvětleno u každého modelu zvlášť.

2.1 Metrika

Metriku chápeme jako rozdíl mezi dvěma množinami prvků - v našem případě rozdílů mezi vypočítanými a naměřenými koncentracemi glukózy. Budeme-li na obě množiny aplikovat metriku, dostáváme tzv. *metrický prostor*. Matematicky je možné metrický prostor definovat jako dvojici (M, p) , kde M je neprázdná množina a p je metrika, což je zobrazení 2.1.

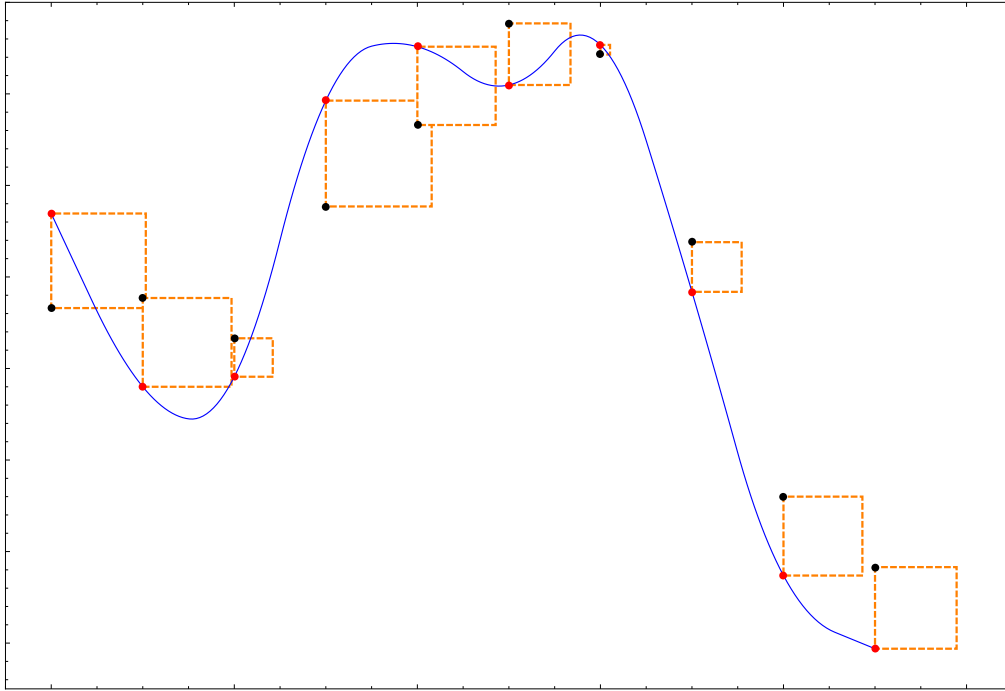
$$M \times M \rightarrow \mathbb{R} \quad (2.1)$$

Jako metriku můžeme použít průměr, medián, směrodatnou odchylku a další. Jednou z možností je také vytvořit si metriku vlastní (viz kapitola 6. *Navržená metrika*). Aplikace metriky může být vhodná například pro hledání koeficientů koncentrace glukózy v krvi při použití genetického algoritmu, kde metrika značí tzv. *fitness* funkci (dále v kapitole 4.4. *Použití HBMO při predikci hladiny glukózy*).

2.2 Model Steil-Rebrinové

Model Steil-Rebrinové (kapitola 3. *Model Steil-Rebrinové*), využívá aproximační metodu nelineárních nejmenších čtverců, která má odhad chyby aproximace svůj vlastní, pro lepší představu se podíváme na obrázek 2.1, kde jsou čtverce / rozdíly mezi naměřenou hodnotou.

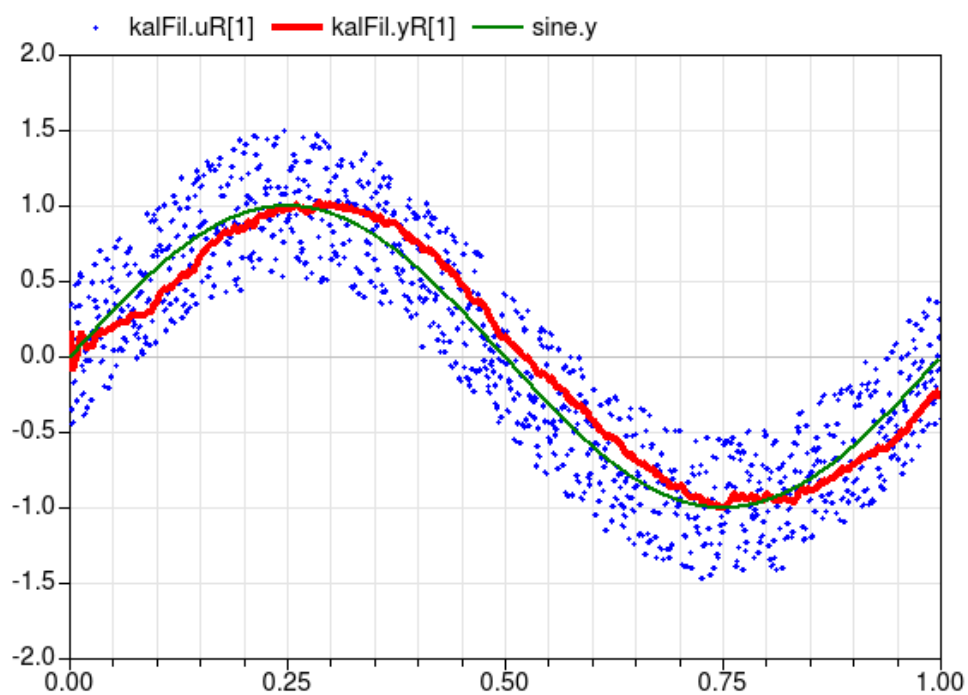
Její iterace se řídí chybou aproximace, proto tato chyba má vztah k zastávající podmínce iteračního procesu. Není proto důvod počítat znovu metriku, i když bychom mohli, protože chybu již známe.



Obrázek 2.1: Reziduum u metody nejmenších čtverců
(Černé body jsou naměřené hodnoty, modrá křivka je výsledek NLS.)

2.3 Auto-regresivní model a Kalmanův filtr

Tento model se zabývá predikcí hladiny glukózy v intersticiální tekutině, jako jediný nepotřebuje data naměřená z krve. Chybu odhadu získáváme z Kalmanova filtru, který ji při predikci, odstraňování šumu a aktualizaci dat vypočítává (Kalmanův filtr a jeho aplikace bude detailně vysvětlena v kapitole 5. *AR model a Kalmanův filtr*). Pro jednoduchou představu, jaké má Kalmanův filtr výsledky se můžeme podívat na obrázek 2.2, kde máme kanál a na vstupu zkreslený sinusový signál, ze kterého Kalmanův filtr rekonstruuje zpátky originální vstup.



Obrázek 2.2: Ukázka Kalmanova filtru
(modrá - vstup, zelená - originál, červená - Kalmanův filtr)
(zdroj: Modelica Buildings Library pod licencií Modelica Licence 2)

Kapitola 3

Model Steil-Rebrinové

Po nástupu zařízení umožňující neinvazivní metodou měřit hladinu glukózy v intersticiální tekutině (dnes již standardně používané CGM), dříve se měřil obsah glukózy z bříška prstů, bylo tudíž nezbytně nutné nalézt vztah mezi intersticiální tekutinou a krví. V roce 2000 Steil s Rebrinovou představili[14] jejich model pro tento vztah:

$$\frac{dC_2(t)}{dt} = -\frac{1}{\tau}C_2(t) + \frac{g}{\tau}C_1(t) \quad (3.1)$$

Studii fungování modelu prováděli na několika subjektech a zjistili chybu menší než 6%, čímž dospěli k závěru, že není překážkou používat hodnoty z intersticiální tekutiny jako náhradu hladiny v krvi.

Proměnná	Popis
C_1	Krev
C_2	Intersticiální tekutina
τ	Přírůstek
g	Koncentrace

Tabulka 3.1: Vysvětlení jednotlivých proměnných rovnice 3.1

3.1 Rozklad na přeурčený systém rovnic

Postupným dosazováním dat získáváme přeурčený systém rovnic 3.2, jehož řešení můžeme získat například pomocí metody nejmenších čtverců, která

Levenberg-Marquardtova metoda

Levenberg-Marquardtova (dále LM) a Gauss-Newtonova metoda jsou si principiálně velice podobné, až na to, že LM místo *line search* techniky využívá *trust region strategy*¹. Použití této strategie nám říká, že vypočítaná změna vektoru p_k musí být v intervalu $0 < \|p_k\| \leq \Delta$, jinak musí nastat aktualizace p_k . Přírodným důsledkem je, že se v každé iteraci se snažíme zajistit minimalizaci $\|p_k\|$. Pro lepší pochopení bude použití NLS vysvětleno na konkrétním příkladu.

Aplikace NLS na SR model

Nejlepší bude si uvést postup na konkrétním příkladu.

1. Mějme množinu dat:

t	$C_1(t)$	$C_2(t)$	$dC_2(t)$
1	9.660	10.903	-26.917
2	9.770	8.815	9.300
3	9.330	9.061	-17.704
4	10.770	11.978	6.314
5	11.660	12.757	-30.716
6	12.770	12.350	-33.556
7	12.430	12.774	-30.143
8	10.380	9.962	-11.164
9	7.600	6.785	0.016
10	6.830	6.109	-25.688

Tabulka 3.2: Data pro aplikaci NLS

2. Abychom mohli použít NLS na náš problém, potřebujeme získat rovnici pro aproximaci hodnot, proto musíme upravit originální rovnici 3.1):

$$\frac{dC_2(t)}{dt} = -\frac{1}{\tau}C_2(t) + \frac{g}{\tau}C_1(t)$$

, vyjádříme si krev (C_1):

$$C_1 = \frac{\frac{dC_2(t)}{dt}}{\frac{g}{\tau}} + \frac{\frac{1}{\tau}C_2(t)}{\frac{g}{\tau}} = \frac{\tau}{g} \frac{dC_2(t)}{dt} + \frac{1}{g}C_2(t)$$

¹překlad z angl: *Strategie důvěryhodného intervalu*, pozn.: český jazyk neumožňuje správný překlad tohoto výrazu, bude proto raději v textu použita originální anglická verze.

, uděláme jednoduchou substituci pro zpřehlednění rovnic:

$$x_1 = \frac{\tau}{g}$$

$$x_2 = \frac{1}{g}$$

a po dosazení získáváme výslednou rovnici, která je vhodná pro využití s NLS:

$$C_1(t) = x_1 \frac{dC_2(t)}{dt} + x_2 C_2(t) \quad (3.3)$$

3. Zvolíme si počáteční parametry, díky systému přeřčených rovnic můžeme například první dvě dosadit a vypočítat, získáme tak počáteční parametry (*pozn: ve výsledné implementaci, kdy se aplikuje NLS na model Steil-Rebrinové je použit jako odhad počátečních parametrů výsledek normální metody nejmenších čtverců, která je NLS upřesňována, dosazení zde slouží pouze pro ukázkou*):

$$9.660 = -26.917x_1 + 10.903x_2$$

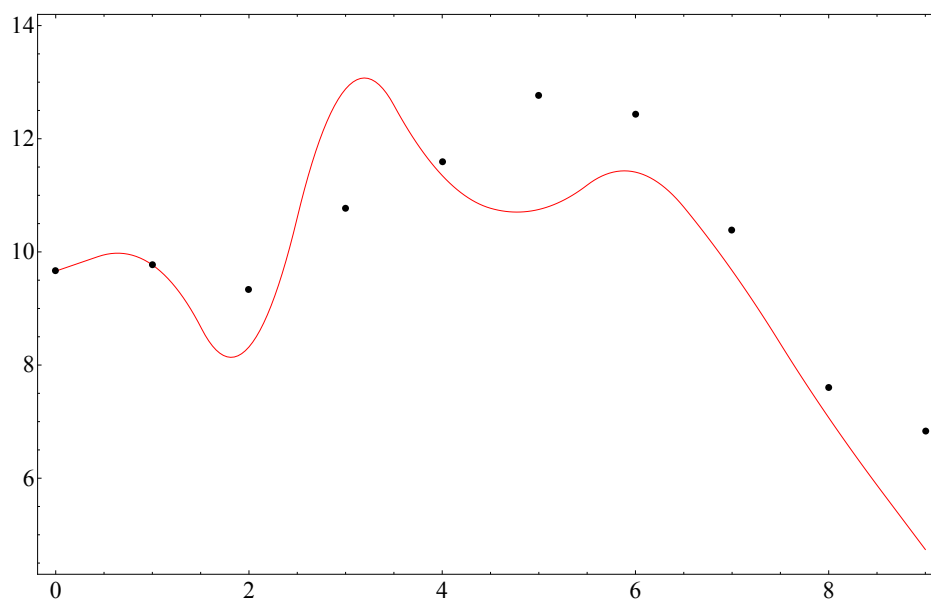
$$9.770 = 9.300x_1 + 8.815x_2$$

Po vyřešení získáváme počáteční parametry:

$$x_1 = 0.06309$$

$$x_2 = 1.04177$$

Pro ukázkou chyby odhadu počátečních parametrů, si funkci vykreslíme (viz obrázek 3.1 na straně 23). Podle očekávání, je chyba u prvních dvou členů zanedbatelná, je to dáno počátečními hodnotami, které vycházely z prvních dvou rovnic.



Obrázek 3.1: Počáteční odhad

4. Nyní můžeme spočítat reziduální vektor $r(x)$ komponenty:

$$r_j = x_1 \frac{dC_2(t)}{dt} + x_2 C_2(t) - C_1(t) \quad (3.4)$$

po dosazení získáváme:

$$r(x) = \begin{pmatrix} 0.0003011 \\ 0.0000013 \\ -1.007404 \\ 2.10676 \\ -0.30792 \\ -2.02110 \\ -1.02406 \\ -0.70615 \\ -0.53053 \\ -2.08644 \end{pmatrix}$$

5. Jestliže máme matici $r(x)$ můžeme získat **chybu aproximace**. *Residuum* - $\|r(x)\|^2$ (označení chyby aproximace NLS) získáme tak, že přidáme mocninu ke každému prvku z $r(x)$ a spočítáme velikost vektoru:

$$\|r(x)\|^2 = 15.815$$

6. V tomto kroku je již rozdíl mezi přístupem Gauss-Newtonovy a LM metody. Musíme si určit Δ (dříve zmíněný *trust region radius*), který budeme předpokládat pro začátek opravdu malý:

$$\Delta = 0.05$$

7. Odhad parametru λ může být komplikovaný (více naleznete v [15] kapitola 4. *Trust region methods*), pro zjednodušení budeme předpokládat, že hodnota λ bude upravována podle chyby z předchozí iterace. Začneme tedy s hodnotou:

$$\lambda = 1$$

8. Aktualizace p_k [16] získáme pomocí:

$$(J_k^T J_k + \lambda I) p_k = -J(x)^T r_k, \lambda > 0 \quad (3.5)$$

9. Matici p_k hledáme, tudíž nám zbývá najít Jaccobiho matici J_k , jejíž sloupce obsahují derivace podle jednotlivých proměnných dx_1 a dx_2 . Protože máme jednoduchou rovnici, můžeme si derivaci vypočítat předem a ušetřit tak čas:

$$\begin{aligned} \frac{dC_1(t)}{dx_1} &= \frac{d}{dx_1} \left(x_1 \frac{dC_2(t)}{dt} + x_2 C_2(t) \right) \\ &= \frac{dC_2(t)}{dt} \end{aligned} \quad (3.6)$$

$$\begin{aligned} \frac{dC_1(t)}{dx_2} &= \frac{d}{dx_2} \left(x_1 \frac{dC_2(t)}{dt} + x_2 C_2(t) \right) \\ &= C_2(t) \end{aligned}$$

Rovnice máme, nyní můžeme dosadit vzorové hodnoty. Výsledná Jaccobiho matice bude vypadat takto:

$$J(x) = \begin{pmatrix} -26.92 & 10.90 \\ 9.30 & 8.82 \\ -17.70 & 9.06 \\ 6.31 & 11.98 \\ -30.72 & 12.76 \\ -33.56 & 12.35 \\ -30.14 & 12.77 \\ -11.16 & 9.96 \\ 0.02 & 6.79 \\ -25.69 & 6.11 \end{pmatrix}$$

10. Dále si všimněme tvaru rovnic pro Jacobiho matici, jak vidíme z 3.6 ani jeden z členů neobsahuje koeficienty x_1 ani x_2 , což znamená, že se tato matice v průběhu iteračního procesu NLS nebude měnit. A proto se nám výpočet 3.5 může značně zjednodušit (*velikosti matic jsou uvedeny v závorkách*).

(a) Předpočítáme si hodnotu (tato matice se již nebude měnit)

$$A_{(2,2)} = J_k^T J_k$$

(b) každém kroku si spočítáme

$$B_{(2,2)} = J(x)^T r_k$$

(c) získáváme celkem triviální výpočet

$$(A_{(2,2)} + \lambda I)p_k = -B_{(2,2)} \quad (3.7)$$

(d) spočítáme si levou stranu

$$C = (A_{(2,2)} + \lambda I)$$

(e) a přenásobíme rovnici inverzní maticí C^{-1} zleva, abychom získali pouze vektor p_k (protože $C^{-1}C = I$):

$$p_k = -C^{-1}B \quad (3.8)$$

11. Nyní jsme schopni inkrementálním procesem optimalizovat úlohu nejmenších čtverců. V nulté iteraci dostáváme:

$$A_0 = \begin{pmatrix} 4927.22 & -1755.49 \\ -1755.49 & 1083.26 \end{pmatrix}$$

$$B_0 = \begin{pmatrix} 200.74 \\ -49.2354 \end{pmatrix}$$

$$p_0 = \begin{pmatrix} -0.0581506 \\ -0.048831 \end{pmatrix}$$

12. V tomto případě je $\|p_0\| = 0.07593$, které nesplňuje podmínku $\|p_0\| \leq \Delta$. Musíme proto pokračovat:

$$r(x) = \begin{pmatrix} 1.03314 \\ -0.97124 \\ -0.42036 \\ 1.15469 \\ 0.85529 \\ -0.67286 \\ 0.10500 \\ -0.54342 \\ -0.86278 \\ -0.89098 \end{pmatrix}$$

Pokud si spočítáme reziduum, vidíme, že hodnota klesla.

$$\|r(x)\|^2 = 6.55$$

Protože se ale hodnota λ nezměnila, bude matice C identická a změní se pouze matice B :

$$B_0 = \begin{pmatrix} -0.0396343 \\ -0.050484 \end{pmatrix}$$

$$p_0 = \begin{pmatrix} 0.0000584614 \\ 0.000141475 \end{pmatrix}$$

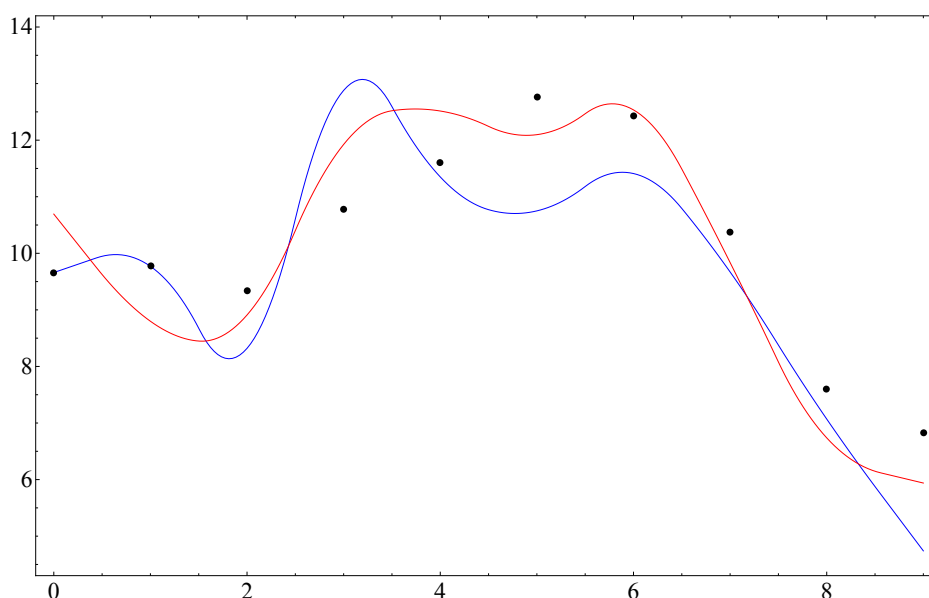
13. V první iteraci nám vyšlo, že $\|p_1\| = 0.000153078$, které splňuje podmínku $\|p_0\| \leq \Delta$. Což znamená, že hodnoty je možné použít. Pokud bychom pokračovali dále v procesu optimalizace aproximace, již bychom nemohli používat aktualizací rovnice, protože naše nové p_k již je v *trust regionu*. Místo toho bychom minimalizovali každé následující p_k pomocí:

$$\frac{1}{2} \|J_k p_k + r_k\|^2, \|p_k\| \leq \Delta_k \quad (3.9)$$

(více v [16]). Zajímavá situace nastává v případě, kdy by se $\|p_k\|$ změnilo, ale stále by nesplňovalo podmínku $\|p_0\| \leq \Delta$. Byly by proto nutné úpravy λ a to:

- $\|p_1\| > \|p_2\|$ - λ se zvýší,
- $\|p_1\| \leq \|p_2\|$ - λ se sníží.

14. Graf 3.2 ukazuje hodnoty po druhé iteraci NLS, kde *červená* je nová hodnota a *modrá* je původní inicializační odhad i pouhým okem, je patrné, že rozdíly jsou mnohem menší.



Obrázek 3.2: Výsledek druhé iterace NLS
(červená - nové hodnoty, modrá - inicializace)

3.1.3 Rychlost výpočtu NLS

Jedním z dílčích cílů této diplomové práce je také optimalizace rychlosti výpočtu. Výpočet inverzní matice může být výpočetně náročná úloha, proto bychom se měli snažit tento výpočet eliminovat, v této části si ukážeme jak toho docílit a porovnáme rychlost výpočtu a počet operací.

Převod na systém lineárních rovnic

Rovnici 3.7 totiž můžeme převést na triviální systém lineárních rovnic. Zaměříme se proto pouze na tento konkrétní krok (protože ostatní výpočty zů-

stávají). Nejprve si ji rozepišme do maticové formy ($Ax = B$):

$$\left(\begin{pmatrix} a_{1,1} & a_{1,2} \\ a_{2,1} & a_{2,2} \end{pmatrix} + \lambda \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \right) \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = - \begin{pmatrix} b_1 \\ b_2 \end{pmatrix}$$

matice sečteme a roznásobíme, abychom dostali systém lineárních rovnic:

$$\begin{aligned} (a_{1,1} + \lambda)x_1 + a_{1,2}x_2 &= -b_1 \\ a_{2,1}x_1 + (a_{2,2} + \lambda)x_2 &= -b_2 \end{aligned}$$

ke druhé rovnici přičteme $-\frac{a_{2,2}+\lambda}{a_{1,2}}$ násobek první, abychom eliminovali člen x_2 , kvůli přehlednosti si upravíme nejdříve pravou a levou stranu zvlášť ($L = R$):

$$L = \left(a_{2,1}x_1 + (a_{2,2} + \lambda)x_2 \right) - \frac{a_{2,2} + \lambda}{a_{1,2}} \left((a_{1,1} + \lambda)x_1 + a_{1,2}x_2 \right)$$

$$L = a_{2,1}x_1 - \frac{(a_{2,2} + \lambda)(a_{1,1} + \lambda)}{a_{1,2}}x_1$$

$$R = -b_2 - \frac{(a_{2,2} + \lambda)}{a_{1,2}}(-b_1)$$

$$R = \frac{(a_{2,2} + \lambda)}{a_{1,2}}b_1 - b_2$$

nyní vytkneme x_1 a necháme ho samostatně na levé straně:

$$x_1 = \left(\frac{(a_{2,2} + \lambda)}{a_{1,2}}b_1 - b_2 \right) \frac{1}{a_{2,1} - \frac{(a_{2,2} + \lambda)(a_{1,1} + \lambda)}{a_{1,2}}}$$

$$x_1 = \left(\frac{(a_{2,2} + \lambda)b_1 - b_2a_{1,2}}{a_{1,2}} \right) \left(\frac{a_{1,2}}{a_{2,1}a_{1,2} - (a_{2,2} + \lambda)(a_{1,1} + \lambda)} \right)$$

a získáváme tak x_1 , toto vyjádření již dále upravit nelze, ale můžeme jeho výpočet optimalizovat

$$\mathbf{x}_1 = \frac{(a_{2,2} + \lambda)b_1 - b_2a_{1,2}}{a_{2,1}a_{1,2} - (a_{2,2} + \lambda)(a_{1,1} + \lambda)}$$

protože již x_1 máme vypočítané, můžeme jeho hodnotu dosadit do jedné z původních rovnic:

$$\begin{aligned}(a_{1,1} + \lambda)x_1 + a_{1,2}x_2 &= -b_1 \\ a_{1,2}x_2 &= -b_1 - (a_{1,1} + \lambda)x_1 \\ x_2 &= \frac{-b_1 - (a_{1,1} + \lambda)x_1}{a_{1,2}}\end{aligned}$$

Při počítání potřebných operací k výpočtu, uvažujme ukládání mezi výsledků, protože obě rovnice obsahují 2 podobné výrazy $(a_{2,2} + \lambda)$ a $(a_{1,1} + \lambda)$.

Inverzní matice

Abychom měli stejné podmínky jako u metody rozkladu, počítejme i s přičtením matice λI .

$$C = (A_{(2,2)} + \lambda I)$$

Inverzní matici budeme výpočetně hledat pomocí determinantu (Gaussova eliminační metoda nemusí být vždy jednoznačná):

$$a_{ij}^{-1} = \frac{(-1)^{i+j} \cdot |A_{j,i}|}{|A|}$$

Pokud bychom si tento postup rozepsali na matici (2×2) zjistíme, že počítáme determinanty podmatic o velikosti 1, což je skalár jehož determinant je ta samá hodnota, a proto:

$$C^{-1} = \begin{pmatrix} a & b \\ c & d \end{pmatrix}^{-1} = \frac{1}{\det(C)} \begin{pmatrix} d & -b \\ -c & a \end{pmatrix}$$

Tuto matici ještě dosadíme do rovnice 3.8 a získáváme finální tvar rovnice pro implementaci.

$$p_k = -C^{-1}B$$

Porovnání počtu operací

V tabulce 3.3 můžeme vidět, že při použití systému lineárních rovnic provedeme o 3 operace méně. Důležité je si také povšimnout poměru násobení a dělení, inverzní matice 12x násobí a jednou dělí, zatímco lineární systém násobí pouze 6x a 2x dělí - tj. jedno dělení navíc má menší režii než 6x operace násobení, což při velkých datech může být již podstatná hodnota.

Operace	Systém rovnic	Inverzní matice
Sčítání	2	2
Odečítání	3	1
Násobení	6	12
Dělení	2	1
Celkem	13	16

Tabulka 3.3: Porovnání počtu operací

3.1.4 Porovnání s matematickým SW

V této kapitole se budeme zabývat kvalitou implementace NLS a porovnání jejich výsledků s běžně používanými matematicko-výpočetními programy, které mají NLS implementováno.

Matlab

Matlab (Matrix Laboratory)² je celosvětově uznávaný program pro matematické výpočty. Byl v sedmdesátých letech vymyšlen uznávaným matematikem Cleve Molerem z důvodů ulehčení práce jeho studentům s matematickými knihovnamy, které byly napsané v jazyce Fortran.

Wolfram Mathematica

Wolfram Mathematica³ je nástroj pro matematické výpočty (stejně jako předem zmíněný konkurenční MATLAB), za kterým stojí stejnojmenná firma *Wolfram Research* založena v roce 1987 Stephenem Wolframem. Mathematica obsahuje množství funkcí včetně sady aproximační výpočtů, jejichž je NLS součástí a také nástrojů pro kreslení 2D i 3D grafů.

Výsledky

Protože NLS již z principu v tomto konkrétním případě nemůže dát různá data, stačí, když výpočet pustíme pouze jednou. Následně se zaměříme na klíčový faktor výpočtu a tím je reziduum $\|r(x)\|^2$, které určuje kvalitu aproximace (tzn. *čím menší tím lepší*). Všechny použité kódy pro porovnání můžete nalézt ve složce *nlstest*.

Z tabulky 3.4 vidíme, že vlastní implementace dosahuje lepších hodnot, pro vizuální představu si můžeme nechat vykreslit graf (viz: obrázek 3.3).

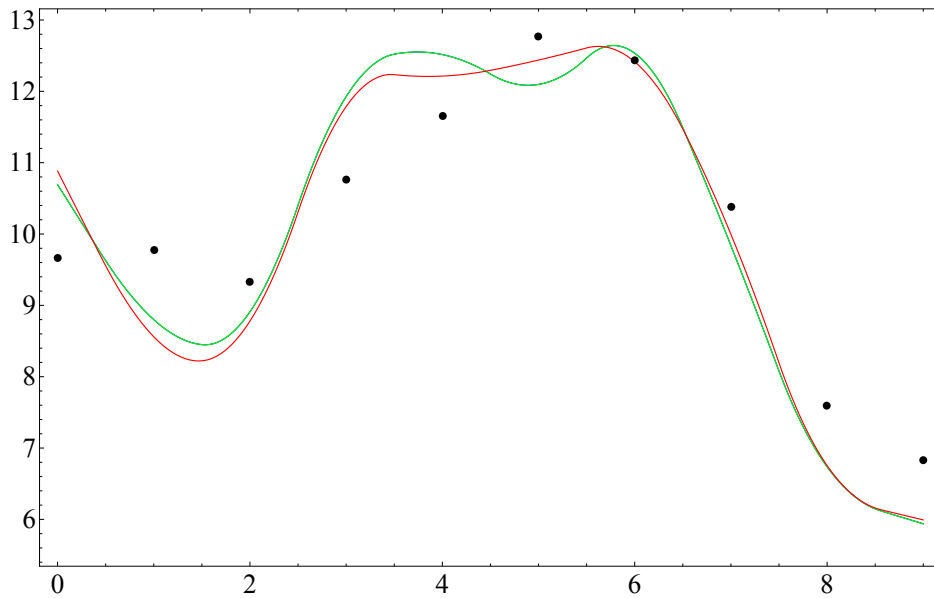
²MathWorks Inc., Natick, Massachusetts, U.S.A, - <http://mathworks.com>

³Wolfram Research - <https://www.wolfram.com/company/>

Operace	Implementace	Wolfram Mathematica	Matlab
$\ \mathbf{r}(\mathbf{x})\ ^2$	6.29736	6.54954	6.5478
\mathbf{t}	-0.00492	0.00499	0.0050
\mathbf{g}	1.02158	0.99306	0.9931

Tabulka 3.4: Srovnání vlastní implementace a Wolfram Mathematica

Bohužel Wolfram Mathematica a Matlab mají přibližně stejné výsledky a graf není dostatečně velký, aby byly tyto rozdíly vidět, ale zelená a modrá čára se překrývají.



Obrázek 3.3: Graf porovnání výsledků
(modrá - Mathematica, zelená - Matlab, červená - vlastní implementace,
černá - naměřená data)

Kapitola 4

Optimalizace pářením včelí královny

Protože bude dále řešit složitější a nelineární model bez známého uzavřeného řešení (než jakým je například model Steil-Rebrinové z předchozí kapitoly) potřebujeme využít heuristický přístup. Obecně používaným heuristickým postupem s relativně dobrými výsledky jsou genetické algoritmy. Jednou z implementací genetických algoritmů je právě optimalizace páření včelí královny.

4.1 Genetické algoritmy

Genetický algoritmus (dále GA) je heuristický postup, který se snaží pomocí principu evoluční biologie získat řešení problémů, pro které neznáme žádný použitelný algoritmus fungující v omezeném čase. GA napodobují evoluční procesy, mezi něž patří dědičnost, mutace, přirozený výběr a křížení. Pro každý úkol se ovšem algoritmus chová jinak. V praxi se GA často používají pro úlohy optimalizace, hledání nejlepších topologií apod. Na rozdíl od gradientních metod, které reprezentují hledání lokálních minim nebo maxim jednoho zpřesňujícího se řešení, GA volí přístup populace (každý jedinec představuje jedno prozatímní řešení), které se v prostoru potkávají a navzájem se ovlivňují.

Pro hledání varianty GA na daný problém je nutná podmínka existence zdatnostní (tzv. *fitness*) funkce, která vyjadřuje kvalitu získaného řešení, a zastavující podmínka (postačující řešení). Statistikou fitness funkce můžeme ukázat rychlost konvergence algoritmu k lepšímu řešení (viz kapitola 4.3 *Předikční model distribuce glukózy*).

Obecné schéma GA je jednoduché. Nejprve se vygeneruje nultá populace, tj. náhodně složení jedinci nebo výsledek z předchozích algoritmů. Dále se pomocí určité výběrové metody zvolí z populace několik jedinců s vysokou zdatností, ze kterých se již zmíněnými evolučními procesy vygeneruje další generace. U všech jedinců nové generace se vypočítá zdatnost. Jestliže je splněna zastavovací podmínka, algoritmus končí. V opačném případě pokračuje

opět mutacemi jedinců za účelem vytvoření další nové generace.

4.2 Algoritmus optimalizace pářením včelí královny

Optimalizace pářením včelí královny (dále HBMO) je varianta genetického algoritmu navrženého v roce 2001 H. A. Abassem [8]. Původní implementace byla použita pro hledání řešení SAT rovnic, kde každé dělnici byl přiřazen různý algoritmus lokálního prohledávání. HBMO, stejně jako většina GA, se na problém dívá jako na černou skříňku, což znamená, že nerozumí vnitřní strukturu problému, a přistupuje tedy pouze k jeho optimalizaci.

4.3 Predikční model distribuce glukózy

V této práci využijeme již existující, vedoucím práce navržený model [4] predikce glukózy, který predikuje hladiny zmíněného zpoždění rozdílu hladin intersticiální tekutiny a krve.

$$\varphi(t) = t + \Delta t + k \times (i(t) - i(t - h))/h \quad (4.1)$$

$$p \times b(t) + c(g) \times b(t) \times (b(t) - i(t)) + c = i(\varphi(t)) \quad (4.2)$$

Proměnná	Popis
$b(t) - i(t)$	rozdíl koncentrace v krvi a intersticiální tekutině přes membránu
p	zisk glukózy z krve v důsledku mezibuněčných rozštěpů
cg	plocha povrchu vynásobená propustností membrány a vydělená tloušťkou membrány
c	residuální koncentrace glukózy
Δt	výchozí interval predikce
k	efekt rychlosti změny koncentračního gradientu
h	interval, na kterém je aplikován parametr k

Tabulka 4.1: Vysvětlený proměnných rovnic 4.1 a 4.2

V získaném modelu nás zajímá právě hodnota $i(\varphi(t))$, která reprezentuje budoucí hladinu glukózy v intersticiální tekutině.

4.4 Použití HBMO při predikci hladiny glukózy

Abychom mohli využít optimalizaci páření včelí královny pro predikci hladiny glukózy v krvi, postačí abychom si uvědomili, že hledáme kořeny rovnice 4.2, budeme-li ji uvažovat jako funkci krve. Vybereme všechny koeficienty a získáváme vektor řešení 4.3:

$$\vec{v} = [cg, p, c, dt, k, h, s] \quad (4.3)$$

Předpokládejme, že všechny prvky mají v počáteční konfiguraci meze – minimum a maximum. Tato podmínka je nezbytná (bude vysvětleno později v kapitole 4.4.5).

4.4.1 Zdatnostní funkce

Zdatnostní (fitness) funkce nám říká, jak dobré nebo dostačující řešení máme. Vzhledem k tomu, že hledáme koeficienty spojité funkce, pak zdatností funkce bude výstupem hodnot vypočtených a teoretických, takže budeme porovnávat dvě spojité funkce. Zde existuje opět vícero možností. V našem případě použijeme jako zdatnostní funkci metriku - viz sekce.

4.4.2 Počáteční inicializace algoritmu

Inicializace algoritmu spočívá v získání počáteční populace včel. Vygenerujeme tedy $N * (V + 1)$ náhodných řešení, kde N je počet královen a koeficient V je velikost vektoru (v našem případě 7). Také zde můžeme využít řešení získaná z některých předchozích metod nebo měření. Následuje krok, který bude v každé iteraci opakován: vypočítání zdatnosti jednotlivých řešení a jejich následné seřazení. Dále je pak prvních N řešení prohlášeno za královny, se kterými se bude uvažovat při svatebních letech.

4.4.3 Svatební lety

Každá královna (q) má tři atributy: energii (E), rychlost (v) a velikost spermaváčku (S). Svatební lety simulují pohyb královny v diskrétním prostoru, kde se setkává s náhodně vygenerovanými trubci (t). Po setkání s trubcem se rozhoduje, zda je genofond trubce natolik silný, že bude nebo nebude křížen s královnou, a to podle následujícího vzorce 4.4:

$$p_{Accept} = \begin{cases} 1 & \text{zdatnost}(q) < \text{zdatnost}(t) \\ e^{\frac{-|\text{zdatnost}(q) - \text{zdatnost}(t)|}{v}} & \text{zdatnost}(q) \geq \text{zdatnost}(t) \end{cases} \quad (4.4)$$

Po výpočtu $pAccept$ musíme vygenerovat náhodné číslo z intervalu $\{0,1\}$, jestliže je toto číslo menší než $pAccept$, bude trubcův genofond přidán do spermaváčku královny. Po každém setkání s trubcem klesá královně energie a rychlost, z čehož vyplývá, že svatební let končí ve chvíli, kdy je energie královny menší nebo rovna 0, nebo obsah jejího spermaváčku dosáhl maximální možné hodnoty.

4.4.4 Křížení genotypů

Pokud je nový vektor přidán ke královně, nastává v další fázi křížení genotypů, toto křížení tudíž probíhá mezi vektorem královny a vektorem trubce. Výsledný vektor je kombinací s 50% šancí prvku z každého členu.

4.4.5 Dělnice

Dojde-li ke křížení genotypu královny a trubce, vzniká nový jedinec. Tohoto jedince se snaží dále optimalizovat dělnice lokálním prohledáváním. V tomto případě právě pomocí horních a dolních mezí pro jednotlivé složky vektoru. Každý interval pro danou složku na-vzorkují 10000 krát a následně vyzkouší, zda-li zdatnost řešení se změněnou hodnotou nedosahuje lepších výsledků než ta původní. Pokud ano, tato složka vektoru je nahrazena. Po vyzkoušení možností optimalizace všech složek vektoru práce dělnic končí.

4.5 Problém lokálních minim

V této variantě HBMO „trpí“ stejným problémem jako gradientní metody, za jejichž zlepšující se řešení je zvoleno lokální minimum. Předpokládejme, že máme počáteční řešení (získané z jiné dříve použité metody, nebo se povedlo náhodně nějaké vygenerovat). Pokud je toto řešení dostatečně silné, prakticky zjišťujeme, že standardní varianta HBMO algoritmu nedostačuje, protože nejsme schopni vygenerovat jiné dostatečně silné náhodné řešení, které by splnilo pářící podmínku 4.4, čímž pak nedochází k dalšímu křížení a nutné obměně genofondu (viz tabulka 4.2 na stránce 37) Zde vzniká problém lokálního minima funkce: jestliže předáme dostatečně silné lokální minimum, pak již s touto variantou nebudeme pravděpodobně schopni nalézt hledané globální minimum.

4.5.1 Přidání věku řešení

Tento problém můžeme vyřešit přidáním věku či *stáří řešení*, jehož hodnota bude po každé iteraci algoritmu narůstat. Přidání věku musí nezbytně ovlivnit i podmínku pro páření 4.4:

$$p_{Accept} = \begin{cases} 1 & \text{zdatnost}(q) < \text{zdatnost}(t) \\ 1 & \text{věk}(q) > \text{věk}(t) \\ e^{\frac{-|\text{zdatnost}(q) - \text{zdatnost}(t)|}{v}} & \text{zdatnost}(q) \geq \text{zdatnost}(t) \end{cases} \quad (4.5)$$

Jak si můžeme všimnou přibyla nová podmínka, tzn. pokud je řešení královny starší než řešení trubce tak je podmínka splněna a hodnota p_{Accept} je nastavena na 1, což v důsledku znamená, že řešení bude přidáno. Ve výsledku tudíž dochází k mnohem častějším mutacím s dalšími trubci a tím se snižuje pravděpodobnost stagnace řešení na lokálním minimu. Praktické výsledky můžeme vidět na první pohled, porovnáme-li tabulku 4.2 s tabulkou 4.3, ve které je již stáří řešení uvažováno.

4.5.2 Vliv na řešení

Několik následujících tabulek a graf ukazují chování a rychlost optimalizace (konvergence) výsledku, který je po seřazení na prvním místě.

Tabulka 4.4 odpovídá tabulce 4.2, aby nedošlo k špatné interpretaci, tabulka 4.4 ukazuje pouze nejlepší objevené řešení, zatímco tabulky 4.2 a 4.3 ukazují mutace všech královen. Všechny tabulky byly umístěny samostatně na další stránky pro přehlednost a snadnější porovnání výsledků.

4.5.3 Vliv iterací a nových mutací před a po přidání stáří řešení

Iterace \ Nových mutací	Bez počátečního řešení	S počátečním řešením
1.	2/1	1/1
2.	1/0	0/0
3.	0/0	0/0
4.	1/0	0/0
5.	0/0	0/0
6.	0/0	0/0
7.	0/0	0/0
8.	0/0	0/0
9.	0/0	0/0
10.	0/0	0/0

Tabulka 4.2: Iterace a počet nových mutací
(První číslo říká, kolik bylo nalezených mutací a druhé, zda byly použity.)

Iterace \ Nových mutací	Bez počátečního řešení	S počátečním řešením
1.	33/33	32/32
2.	7/7	4/1
3.	7/6	8/4
4.	5/4	8/2
5.	7/5	7/2
6.	7/3	10/6
7.	5/4	6/3
8.	11/5	15/7
9.	15/9	17/6
10.	2/1	8/3

Tabulka 4.3: Iterace a počet nových mutací po přidání věku
(První číslo říká, kolik bylo nalezených mutací a druhé, zda byly použity.)

4.5.4 Vliv přidání stáří řešení na výsledky algoritmu

Svatební let \ Metoda	Bez počátečního řešení	S počátečním řešením
1.	0.8612970	1.1386660.
2.	0.8512970	1.0382390.
3.	0.8512970	0.9980690.
4.	0.8512970	0.9980690.
5.	0.8512970	0.9980690.
6.	0.8512970	0.9980690.
7.	0.8512970	0.9980690.
8.	0.8512970	0.9980690.
9.	0.8512970	0.9980690.
10.	0.8512970	0.9980690.

Tabulka 4.4: Sumář výsledků po svatebních letech
(Metoda bez přidáním věku řešení.)

Svatební let \ Metoda	Bez počátečního řešení	S počátečním řešením
1.	0.8612970	2.4918540
2.	0.7942880	0.9129430
3.	0.7923580	0.8714760
4.	0.7923580	0.8039720
5.	0.7923580	0.8005340
6.	0.7921400	0.8001470
7.	0.7920090	0.7994840
8.	0.7917760	0.7982720
9.	0.7917760	0.7949210
10.	0.7917760	0.7949210

Tabulka 4.5: Sumář výsledků po svatebních letech
(Metoda s přidáním věkem řešení.)

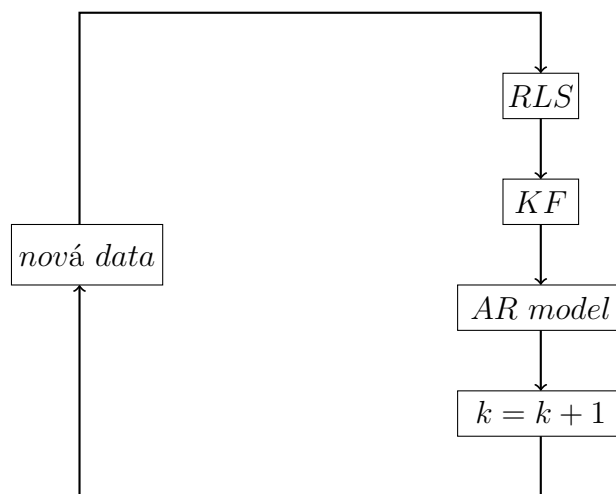
Kapitola 5

AR model a Kalmanův filtr

Predikce hladiny glukózy v intersticiální tekutině je neméně důležitý úkol pro ovládací software umělé slinivky. Bylo představeno několik metod kde byl použit auto regresivní model jakožto slibný model pro predikci hladin, ačkoliv identifikace parametrů je stále problém. Youqing Wang a Xiangwei Wu představili využití Kalmanova filtru pro identifikaci parametrů AR modelu[7].

5.1 Použití

Predikce hladiny glukózy se bude řídit podle následujícího stavového diagramu 5.1 pro aktualizaci hodnot [7], jednotlivé principy budou vysvětleny dále.



Obrázek 5.1: Stavový diagram průběhu a aktualizace hodnot

Každá z metod pracuje pouze s omezeným množstvím dat $\{-dt, t+1\}$, velikost tohoto okénka je závislá na počtu naměřených hodnot a odvíjí se od ní výpočetní rychlost, protože určuje velikost matic, které budou násobeny. Zásadním poznatkem této práce je, že musíme použít originálně naměřená data nebo aproximaci s dostatečně velkým krokováním, protože touto metodou

jdou velice špatně predikovat hodnoty z dat s malým rozestupem hodnot (více v sekci 5.5. *Problém predikce z dat s příliš častým vzorkováním*).

5.2 RLS

Rekurzivní metoda nejmenších čtverců¹ je (podobně jako NLS použitá v kapitole 3. *Model Steil-Rebrinové*) metoda řešení soustavy nelineárních rovnic, která minimalizuje součet čtverců residuí mezi vypočítanými a naměřenými hodnotami.

Mějme pře-určenou soustav rovnic (tj. soustava, kde je více rovnic než neznámých).

$$a_1(t)x_1 + a_2(t)x_2 + \dots + a_n(t)x_n = b(t)$$

Tuto soustavu si rozepíšeme do maticové podoby (bude využita později).

$$A_0x_0 = B_0$$

Matice A_0 resp. $a_j(t)$ zde představuje naměřená data, matice B_0 resp. $b(t)$ predikované hodnoty a x_j jsou parametry, které musíme určit. Narozdíl od NLS zde ale nedosazujeme všechny hodnoty, ale pouze množství dat dané velikostí zpracovávaného úseku, jak již bylo zmíněno dříve.

5.2.1 Iterační rovnice RLS

Uvedme si iterační rovnice[16].

$$\begin{aligned} x &= x_0 + k(b - a^T x_0) \\ k &= \frac{1}{1 + a^T P_0 a} P_0 a \\ P &= [I - k a^T] P_0 \end{aligned} \tag{5.1}$$

Pro určení zastavující podmínky, musíme určit požadovanou přesnost výsledného řešení např. $limit = 0.0001$. Po každé aktualizaci x si spočítáme Ax a získáme hodnotu B (matice B je v našem případě skalár). Tuto hodnotu odečteme s očekávanou hodnotou b a získáváme chybu aproximace. Jako počáteční konfiguraci zvolíme P_0 diagonální jednotkovou matici a x_0 bude jednotková matice.

¹z angl. *recursive least squares*

Optimalizace výpočtu

Z iteračních rovnic RLS 5.1 můžeme vyzorovat, že hodnoty k a P budeme počítat pouze jednou (jejich parametry a , P_0 se neaktualizují). Oproti tomu první rovnice se bude opakovat, dokud nebude splněná zatahující podmínka.

5.2.2 Rozdíl mezi NLS a RLS

Hlavním rozdílem je, že u NLS si můžeme výslednou funkci volit sami podle zadaných dat (např. exponenciální) a nejsme tak, nijak omezeni. Zatímco RLS vždy hledá koeficienty $a_1(t)x_1 + a_2(t)x_2 + \dots + a_n(t)x_n = b(t)$, nemůžeme si tudíž zvolit funkci jejíž parametry hledáme.

5.3 Kalmanův filtr

Kalmanův filtr byl představen v roce 1960 Rudolfem E. Kalmanem, je to jednoduchý algoritmus, který slouží k predikci stavu systému z aktuálních dat pomocí diferenciální rovnice 5.2. V této konkrétní aplikaci neupravuje výsledné koeficienty rovnic z RLS, ale řeší chybu a šum v systému.

$$x_k = Ax_{k-1} + Bu_k + w_{k-1} \quad (5.2)$$

a naměřených hodnot z 5.3, jednotlivé proměnné jsou vysvětleny v tabulce 5.1.

$$z_k = Hx_k + v_k \quad (5.3)$$

Proměnná	Popis
$x(k)$	stav v k
z	měření
u	kontrolní signál / vstup
w	šum v procesu
v	šum v měření

Tabulka 5.1: Vysvětlení jednotlivých proměnných

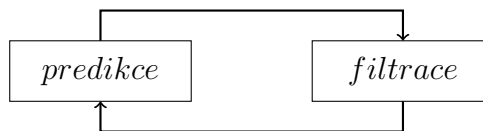
A, B a H jsou matice, v reálném příkladě to budou s největší pravděpodobností skaláry a ke všemu rovné $\mathbf{1}$. Využití kalmanův filtr nalézá také v teorii řízení (letadla, auta). Mezi hlavní výhody použití kalmanova filtru:

- predikce stavu systému v závislosti na předchozím chování,
- využití redundantních zdrojů stavu systému, ignorování těch chybných a tím přesnější chování,
- filtrování šumu,
- **stanovení přesnosti.**

Kalmanův filtr nám tedy umožňuje stanovit **chybu modelu!** Tato chyba bude v následujících rovnicích vystupovat jako matice P , která degraduje na skalár.

5.3.1 Fáze filtrování

Pomocí rovnic 5.2 a 5.3 můžeme rozvést jednotlivé iterace Kalmanova filtru, tyto iterace jdou rozdělit na dvě fáze *predikci* a *korekci* [12] (viz obrázek 5.2).



Obrázek 5.2: Fáze filtrování

Predikce

Jak již název napovídá, predikce určuje stav systému v X_{k+1} .

1. Predikce stavu systému

$$\hat{x}_k^- = A\hat{x}_{k-1} + Bu_k$$

2. Výpočet kovariance chyby v predikci

$$P_k^- = AP_{k-1}A^T + Q$$

Korekce

1. Výpočet kalmanova přírůstku²

$$K_k = P_k H^T (H P_k H^T + R)^{-1}$$

2. Aktualizace odhadu x_k pomocí naměřených hodnot z_k

$$\hat{x}_k = \hat{x}_k^- + K_k (z_k - H \hat{x}_k^-)$$

3. Aktualizace kovariance chyby

$$P_k = (I - K_k H) P_k^-$$

5.3.2 Aplikace KF

Protože zpracováváme pouze 1D signál, tak se rovnice Kalmanova filtru značně zjednoduší. Všechny hodnoty se stávají skaláry místo matic, navíc nemáme kontrolní signál $u(k)$. Matice H , B a A jsou rovny $\mathbf{1}$, protože víme, že máme naměřené hodnoty stavu, je v nich šum a nemáme kontrolní signál. Získáváme tedy:

$$\begin{aligned} x_k &= A x_{k-1} + u_k + w_{k-1} = x_{k-1} + w_{k-1} \\ z_k &= H x_k + v_k = x_k + v_k \end{aligned}$$

Tyto poznatky můžeme aplikovat na predikční rovnice:

$$\begin{aligned} \hat{x}_k &= A \hat{x}_{k-1} \\ P_k &= P_{k-1} \end{aligned}$$

a následně i na korekční rovnice:

$$\begin{aligned} K_k &= \frac{P_k^-}{P_k^- + R} \\ \hat{x}_k &= \hat{x}_k^- + K_k (z_k - \hat{x}_k^-) \\ P_k &= (1 - K_k) P_k^- \end{aligned}$$

Jako počáteční P_0 musíme zvolit jinou hodnotu než 0, protože pokud bychom jí zvolili, znamenalo by to, že v systému není žádný šum a to by vedlo k tomu, že všechny další \hat{x}_k by byly nulové.

²z angl. *kalman gain*

5.4 Auto-Regresivní model

Auto-Regresivní (dále jen AR) model je oblíbená struktura pro popis dynamických systémů / procesů závislých na čase, který je popsán rovnicí 5.4.

$$y(k) = \sum_{i=1}^n a_i(k)y(k-i) + e(k) \quad (5.4)$$

Proměnná	Popis
$y(k)$	predikovaná hodnota
n	počet předchozích hodnot
a_i	koeficienty AR modelu
$y(k-i)$	předchozí hodnoty
$e(k)$	šum

Tabulka 5.2: Vysvětlení jednotlivých proměnných

AR model obsahuje *paměťový buffer* na základě, kterého predikuje data, jinak řečeno predikuje hodnotu proměnné lineární kombinací předchozích hodnot.

5.4.1 Aplikace AR modelu

Definici AR modelu si můžeme zjednodušit (následující odstavec a výpočty jsou citace z [7]). Abychom mohli rovnici 5.4 upravit, předpokládejme:

$$E[e(k)] = 0, \text{Var}[e(k)] = \sigma^2$$

a získáváme:

$$y(k) = h_{k-1}^T \theta(k) + e(k) \quad (5.5)$$

h a θ jsou matice, které můžeme rozepsat jako:

$$h_{k-1} = [y(k-1) \quad y(k-2) \quad \dots \quad y(k-n)]^T$$

$$\theta(k) = [a_1(k) \quad \dots \quad a_n(k)]^T$$

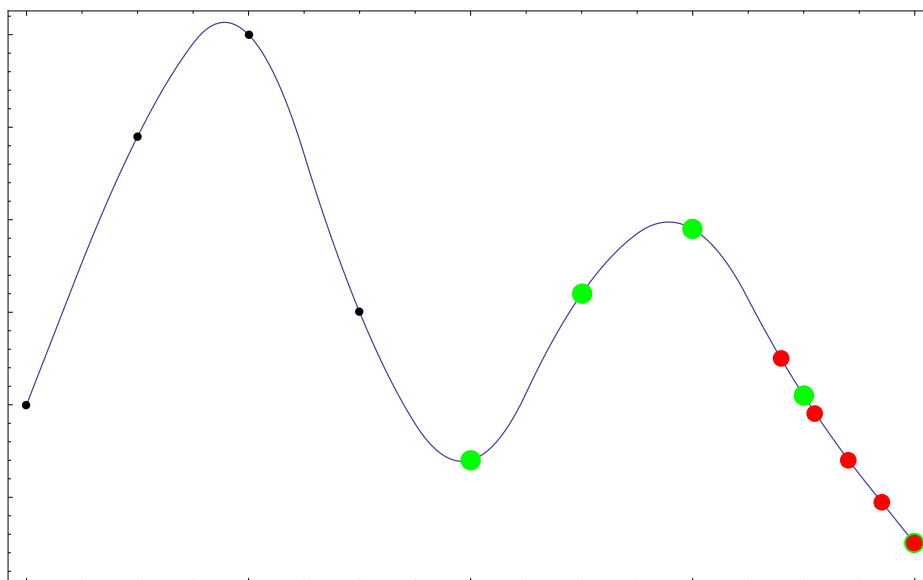
Do těchto rovnic již stačí dosadit výsledky a máme požadovanou predikovanou hodnotu.

5.5 Problém predikce z dat s příliš častým vzorkováním

Tento problém vychází již z matematické podstaty AR modelu a úzce souvisí i s výpočetní náročností. Tento problém se vyskytuje u aproximace s vysokým vzorkováním. AR model má „omezený“ paměťový buffer, ze kterého dělá lineární kombinaci pro získání nové hodnoty. Není náhodou, že velikost tohoto paměťového bufferu je v tomto případě stejně velká jako velikost *výpočetního okénka modelu*, které bylo uvedeno na začátku. Protože model obsahuje výpočty maticových operací, tak pokud by byl buffer jinak velký, nebude splněna nutná podmínka pro násobení dvou matic.

Pro lepší pochopení tohoto problému si ukažme jednoduchý příklad. Mějme data měřená každých 15 minut, které aproximujeme a použijeme data vzorkovaná po 30 sekundách. Pokud na konci měření (začátek predikovaných dat) byl pokles nebo vzestup hladiny glukózy, tak při použití dat s příliš častým vzorkováním bude náš AR model obsahovat s největší pravděpodobností data pouze o tomto poklesu (viz obrázek 5.3), tudíž další predikovaná hodnota bude nezbytně znovu pokles, který díky chybě, se kterým AR model počítá bude neustále větší.

Proto je dobré používat data naměřená nebo s dostatečně vypovídajícím vzorkováním.



Obrázek 5.3: Ukázka dat s příliš častým vzorkováním
Zelená - dlouhé vzorkování, Červená - krátké vzorkování

Kapitola 6

Navržená metrika

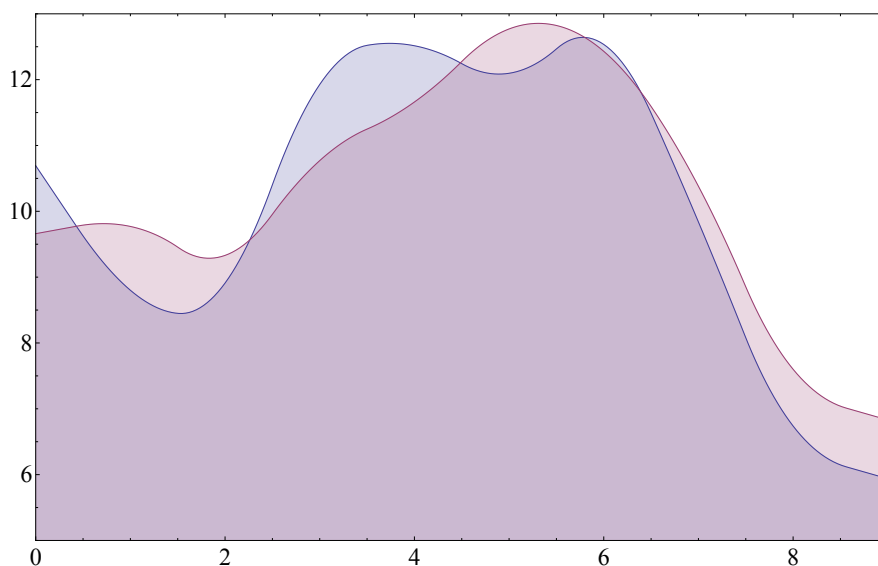
Budeme-li vycházet z předpokladu, že funkce, které budeme porovnávat jsou zhruba stejné, tak můžeme vyjádřit jejich rozdíl pomocí obsahu pod grafem.

6.1 Matematické vyjádření

Mějme funkce $f(x)$ a $g(x)$ na intervalu $\{0, t\}$ v konkrétním případě, by $f(x)$ byla naše výsledná aproximace a $g(x)$ byla interpolace hladiny glukózy v krvi. Výslednou metriku m , kterou počítáme můžeme zapsat 6.1.

$$m = \left| \int_0^t f(x)dx - \int_0^t g(x)dx \right| \quad (6.1)$$

6.2 Znázornění grafem



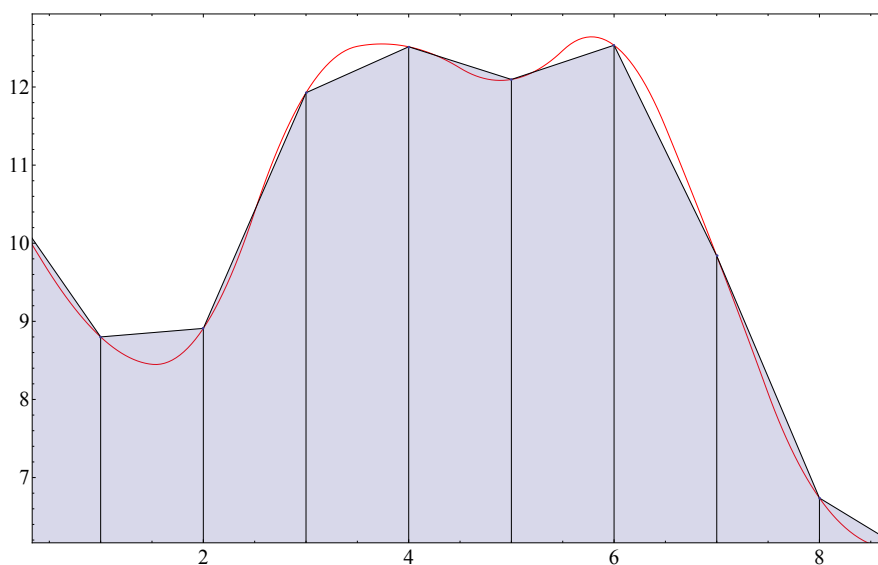
Obrázek 6.1: Metrika

6.3 Numerická integrace

Budeme muset numericky vypočítat tyto dva integrály, vhodným postupem bude *lichoběžníková metoda*:

$$\int_a^b f(x)dx \approx = \frac{h}{2}(f(x_0) + f(x_n)) + h(f(x_1) + \dots + f(x_{n-1})) \quad (6.2)$$

Zde bychom mohli bychom použít i jinou (např. Simpsonova metodu), ovšem h budeme volit dostatečně malé tak, že by jiné metriky způsobovaly zbytečné výpočty navíc avšak se stejným výsledkem. Zvolíme si tedy dostatečně malý interval h , funkci na-vzorkujeme a u každé hranice si vypočítáme obsah lichoběžníka, který vzorkováním vznikne. Pro lepší představu, využijeme opět grafické znázornění:



Obrázek 6.2: Lichoběžníková metoda integrace

Výsledné obsahy odečteme a získáváme požadovanou metriku.

Kapitola 7

Programová dokumentace

Celý programový kód je koncipován jako knihovna, kterou využívá program vedoucího diplomové práce. Tento postup byl vybrán z důvodu, že modely budou již nadále integrovány do tohoto programu, pro potřeby porovnání výsledků naměřených hodnot a budou zdrojem dalšího výzkumu. To je důvod proč výsledný zdrojový kód neřeší žádná uživatelská rozhraní ani načítání dat. Celý zdrojový kód je na přiloženém CD ve složce *src*.

7.1 Matice

- **lib/matrix.h** - Definice šablony představující matici.

Ve výpočtech z kapitol 3. *Model Steil-Rebrinové* a 5. *AR model a Kalmanův filtr* byly nutné komplexnější a často se opakující maticové operace. Tento problém bylo možné řešit dvěma způsoby a to vlastní implementací jednoduché maticové knihovny, nebo použití již hotového řešení, jakým je třeba *Eigen*¹. Nakonec byla vybrána implementace vlastní knihovny jako nejlepší řešení, díky možné kontrole a optimalizacím vlastního zdrojového kódu s ohledem na specifikaci řešeného problému.

7.1.1 Výhody vlastní implementace

Nespornou výhodou je, že nemusíme řešit N -rozměrné matice obecně, ale stačí nám pouze dvourozměrné a vektory. To má za důsledek, že je tato třída jednodušší a výkonnější. Veškerá data jsou ukládána do jednorozměrného pole, což vede i k méně známým postupům jako například transponování matice s minimální další alokovanou pamětí. Třída je psaná jako šablona, takže nejsme omezeni jedním konkrétním datovým typem pro matici, ale můžeme si vytvářet vlastní a tím šetřit paměťovou náročnost.

¹C++ knihovna pro matematické operace (<http://eigen.tuxfamily.org/>)

7.2 Implementace jednotlivých modelů

Podle typu modelu a výpočtu může model obsahovat třídu pro řešení (\times *Solver*) a pro vypočítání rozdílů oproti původním datům (*Calculation*), to-muto designovému patternu se bylo nutné podřídit, aby knihovna fungovala bez problému.

Třída pro řešení dědí od abstraktní třídy *ISolver* a musí implementovat metodu *Solve*, která vypočítá hledané parametry.

Třída pro výpočet rozdílů oproti naměřeným hodnotám dědí od abstraktní třídy *CIstCalculation* a musí implementovat metody *DoTheCalculation*, která slouží pro vizualizaci a *CalculateDifferences*, která se využívá pro zjištění rozdílů a tudíž chyby aproximace.

7.2.1 Steil-Rebrin

Implementace Steil-Rebrinova modelu jako jediná v této práci má jak třídu pro hledání řešení tak pro kalkulaci odchylek.

- **SRSolver.h** - Definice třídy *SRSolver*.
- **SRSolver.cpp** - Implementace třídy *SRSolver*, která se snaží vyřešit parametry Steil-Rebrinův modelu.
- **SRCalculation.h** - Definice *SRCalculation*.
- **SRCalculation.cpp** - Implementace třídy *SRCalculation*, této třídě jsou předávány parametry vyřešené *SRSolverem*.

7.2.2 HBMO

Odhad parametrů pomocí optimalizace včelím rojem má pouze třídu pro hledání parametrů, protože hledá parametry modelu, pro který již kalkulace odchylek v programu existuje.

- **HBMO Solver.h** - Definice třídy a pomocných konstant pro HBMO.
- **HBMO Solver.cpp** - Samotná třída pro hledání koeficientů modelu.

7.2.3 AR + Kalman

Opakem HBMO je implementace AR + Kalmana, která potřebuje pouze třídu pro kalkulaci. Důvod je ten, že Kalmanův filtr nehledá žádné parametry modelu, ale pouze aplikuje iterační chování na naměřená data s dostatečnou predikcí.

- **ARCalmanCalculation.h** - Definice třídy *ARKalmanCalculation* a *ARKalmanPrediction*.
- **ARCalmanCalculation.cpp** - Implementace třídy *ARKalmanCalculation*.

7.3 Vlastní metrika

Metrika dědí od abstraktní třídy *CCommonDiffMetric*, vzhledem k jednoduchosti metriky je nutné přepsat pouze metodu *DoCalculateMetric*, která vrací hodnotu metriky ze zadaných dat.

- **AreaUnderCurve.h** - Definice třídy *CAreaUnderCurve*.
- **AreaUnderCurve.cpp** - Implementace třídy *CAreaUnderCurve*.

7.4 Kompilace

Pro kompilaci zdrojových kódů je potřeba následující:

- Visual Studio 2013 - soubor s projektem je součástí CD ve složce *src/msvc*.
- Intel® Thread Building Blocks ve verzi 4.3 Update 5 - knihovna pro paralelní výpočty, je zdarma dostupná na oficiální stránce projektu <https://www.threadingbuildingblocks.org/>.
- Visual Leak Detector - jednoduchý nástroj, který detekuje neuvolněnou paměť v kódu (<https://vld.codeplex.com/>).

7.5 Spuštění

Součástí CD je i spustitelná verze programu poskytnutá vedoucím práce. Ve složce **bin** můžeme spustit soubor **gpredict2.exe**, pokud jsou správně nastavené cesty k databázi dat (soubor *config.ini*) program se spustí a je možné vybrat metodu výpočtu a výpočty spustit. Příložená data jsou ukázková, náhodně vygenerována.

Kapitola 8

Výsledky

Všechny modely implementovány v této práci byly testovány tak jak byly navrženy autory. Tato kapitola obsahuje pouze hodnoty měření modelů a informace o datech, výsledné porovnání bude v sekci *9.3. Závěr a porovnání metod* na stránce 59.

8.1 Měřená data

Data pro měření byla poskytnuta panem vedoucím, pouze pro účely této diplomové práce, data obsahovala data 9 pacientů s diabetem typu 1. Veškeré zkoumané chyby modelů a výsledky v této diplomové práci jsou získány pouze z dat lidských subjektů. Všechny implementované modely byly identifikovány na diabetických pacientech.

8.2 Vysvětlení grafů

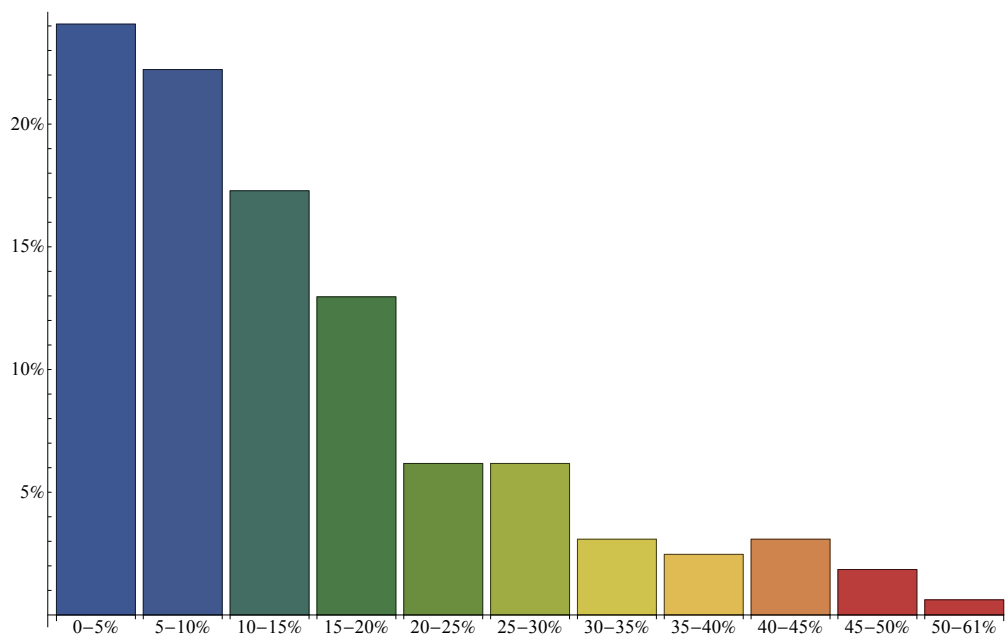
Všechny grafy v této kapitole jsou koncipovány stejně, ukazují chybu modelu, kde osa X značí meze relativní chyby a osa Y , celkové procento počtu prvků, které v dané relativní chybě byly. Jednoduše řečeno čím větší procento prvků je v menších relativních chybách (levá strana grafu), tím je model přesnější.

8.3 Modely koncentrace hladiny v krvi

8.3.1 Výsledky aplikace modelu Steil-Rebrinové

Relativní chyba	Počet hodnot	%
0.0 - 5.0%	39	24.07%
5.0 - 10.0%	36	22.22%
10.0 - 15.0%	28	17.28%
15.0 - 20.0%	21	12.96%
20.0 - 25.0%	10	6.17%
25.0 - 30.0%	10	6.17%
30.0 - 35.0%	5	3.09%
35.0 - 40.0%	4	2.46%
40.0 - 45.0%	5	3.09%
45.0 - 50.0%	3	1.85%
50.0 - 61.6%	1	0.62%

Tabulka 8.1: Relativní chyby modelu Steil-Rebrinové

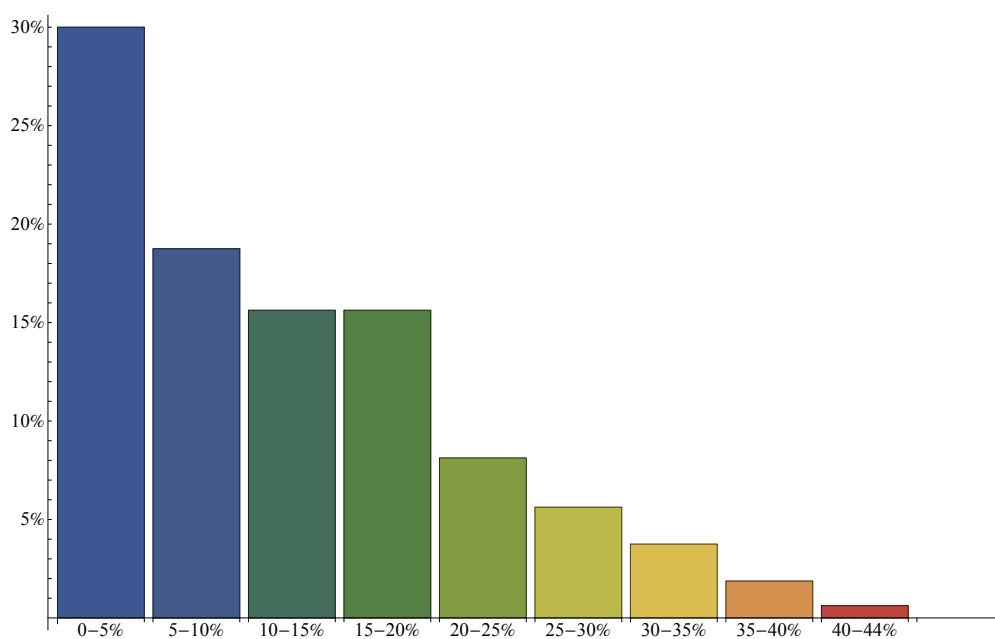


Obrázek 8.1: Relativní chyby modelu Steil-Rebrinové

8.3.2 Výsledky optimalizace páření včelí královny

Relativní chyba	Počet hodnot	%
0.0 - 5.0%	48	30.00%
5.0 - 10.0%	30	18.75%
10.0 - 15.0%	25	15.63%
15.0 - 20.0%	25	15.63%
20.0 - 25.0%	13	8.13%
25.0 - 30.0%	9	5.63%
30.0 - 35.0%	6	3.75%
35.0 - 40.0%	3	1.88%
40.0 - 44.8%	1	0.63%

Tabulka 8.2: Relativní chyby modelu řešeném HBMO



Obrázek 8.2: Relativní chyby modelu řešeném HBMO

8.4 Modely predikce hladiny v intersticiální tekutině

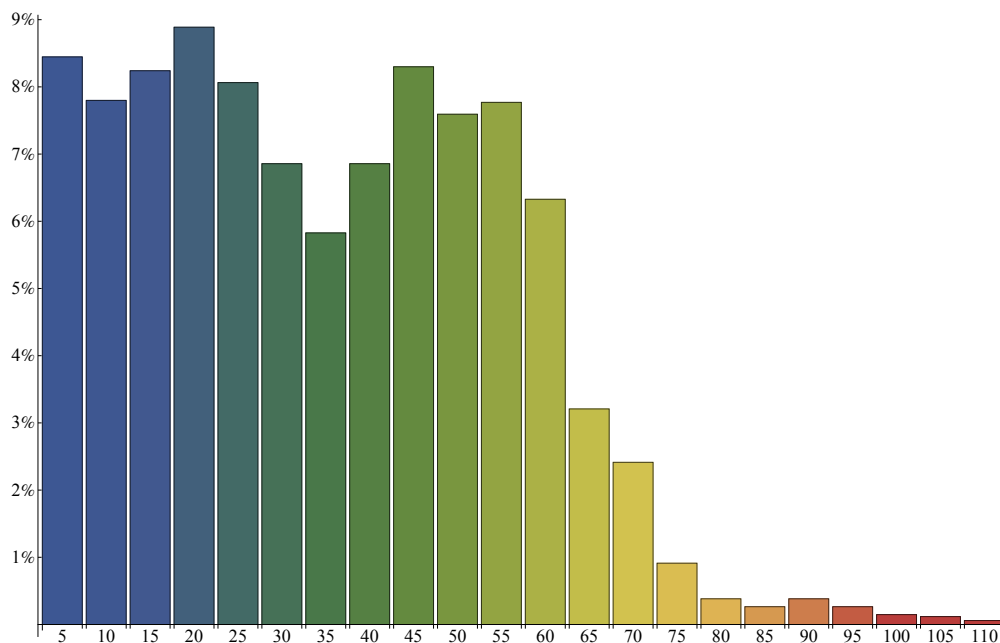
8.4.1 Výsledky Kalmanův filtr a auto regresivní model

Relativní chyba	Počet hodnot	%
0.0 - 5.0%	287	8.45%
5.0 - 10.0%	265	7.80%
10.0 - 15.0%	280	8.24%
15.0 - 20.0%	302	8.89%
20.0 - 25.0%	274	8.06%
25.0 - 30.0%	233	6.86%
30.0 - 35.0%	198	5.83%
35.0 - 40.0%	233	6.86%
40.0 - 45.0%	282	8.30%
45.0 - 50.0%	258	7.59%
50.0 - 55.0%	215	7.77%
55.0 - 60.0%	109	6.33%
60.0 - 65.0%	82	3.21%
65.0 - 70.0%	31	2.41%
70.0 - 75.0%	13	0.91%
75.0 - 80.0%	9	0.38%
80.0 - 85.0%	13	0.26%
85.0 - 90.0%	9	0.38%
90.0 - 95.0%	5	0.26%
95.0 - 100.0%	4	0.15%
ostatní > 100%	36	1.06%

Tabulka 8.3: Relativní chyby Kalmanova filtru a auto regresivního modelu

8.4.2 Výsledky difuse modelu

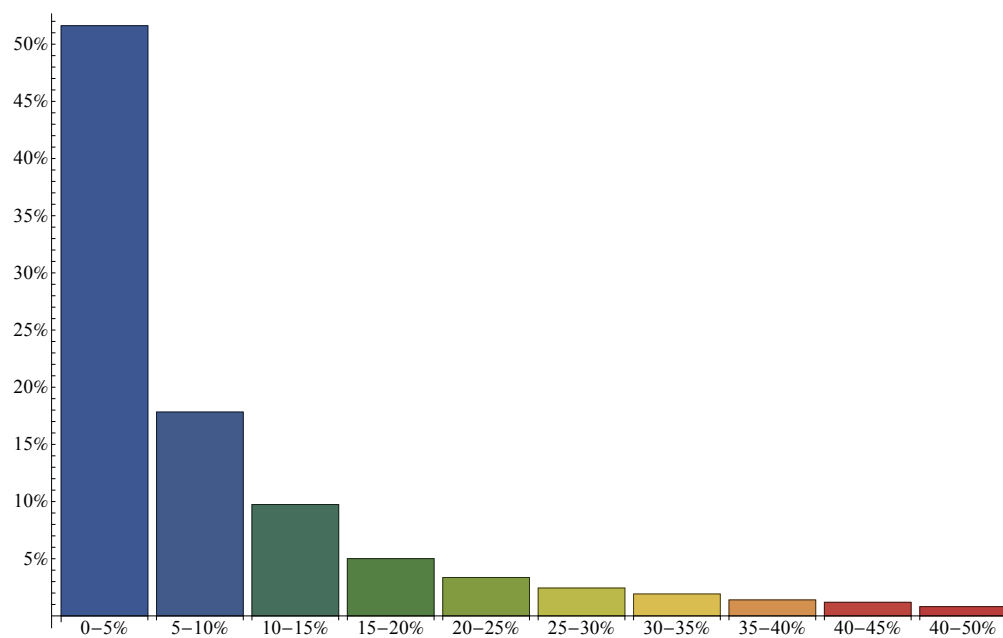
Protože v této práci byl implementován pouze jeden model pro predikci hladiny glukózy v intersticiální tekutině, využijeme, že ve stávajícím programu je již implementován difuse model[17]. Difuse model také predikuje hladinu glukózy v intersticiální tekutině.



Obrázek 8.3: Relativní chyby Kalmanova filtru a auto regresivního modelu

Relativní chyba	Počet hodnot	%
0.0 - 5.0%	4615	51.62%
5.0 - 10.0%	1594	17.83%
10.0 - 15.0%	871	9.74%
15.0 - 20.0%	448	5.01%
20.0 - 25.0%	300	3.36%
25.0 - 30.0%	218	2.44%
30.0 - 35.0%	172	1.92%
35.0 - 40.0%	125	1.40%
40.0 - 45.0%	106	1.19%
45.0 - 50.0%	106	0.81%
ostatní > 50%	347	3.88%

Tabulka 8.4: Relativní chyby difuze modelu



Obrázek 8.4: Relativní chyby difuze modelu

Kapitola 9

Závěr

9.1 Referenční stroj

Veškeré pokusy byly prováděny na následující konfiguraci, u HBMO byl kladen velký důraz na paralelizaci.

- CPU: Intel Core i5 750 @ 2.67 GHz
- RAM: 16GB DDR3
- OS: Windows 7 64bit Professional

Byla-li porovnávána rychlost kódu, byl vždy kompilován jako *release* bez ladících informací.

9.2 Obsah CD

- **src** - zdrojové soubory implementované knihovny
- **sr-compare** - soubory pro Wolfram Mathematica a Matlab pro porovnání výsledků NLS
- **doc** - tento text
- **tex** - zdrojový kód L^AT_EXu pro tuto práci
- **charts** - soubory Wolfram Mathematica pro generování grafů do této práce
- **bin** - přeložená knihovna a aktuální verze *Glucose 2 predictor*
- **data** - testovací datové soubory

9.3 Závěr a porovnání metod

Tato práce se zabývá implementací algoritmů dostupných modelů pro predikci a koncentraci glukózy, na kterých zkoumá odhad chyby pomocí metrik nebo jiných matematických aparátů.

9.3.1 Porovnání metod pro koncentraci glukózy v krvi

Ze dvou porovnávaných metod (model Steil-Rebrinové a hledáním koeficientů pomocí optimalizace včelím rojem) vychází mnohem přesněji model, na který byla aplikována optimalizace včelím rojem. Tento rozdíl můžeme vidět v tabulkách 8.1 a 8.2, kde relativní chyba u HBMO obsahuje mnohem více dat s menší chybou. I přes to, že model není úplně přesný, lze ho prohlásit za dostačující.

9.3.2 Porovnání metod pro predikci hladiny glukózy v intersticiální tekutině

Obě použité metody nevrací dobré výsledky, ale podle výsledků z tabulek 8.4 a 8.1 vychází lépe difuze model.

9.3.3 Rychlost výpočtu

Pokud by nás zajímala rychlost výpočtu, tak nejnáročnější a tudíž nejpomalejší je *optimalizace včelím rojem*, protože provádí nejvíce operací nad relativně velkými daty. Naopak nejrychlejší metoda je *model Steil-Rebrinové*, který primárně počítá s daty z krve, kterých je vždy mnohem méně.

9.3.4 Výsledky metriky

Program již obsahoval sofistikované jiné metriky, šance vymyslet novou a lepší, byla mizivá. Metrika splňuje požadavky a lze jí používat, ovšem při porovnání s již implementovanými metrikami nevrací ani z daleka tak dobré výsledky.

Reference a zdroje

- [1] IDF Diabetes Atlas. *International Diabetes Federation* [online]. 2014. vyd. [cit. 2015-05-09]. Dostupné z: <http://www.idf.org/diabetesatlas>
- [2] IDF Diabetes Complications. *International Diabetes Federation* [online]. [cit. 2015-04-20]. Dostupné z: <http://www.idf.org/complications-diabetes>
- [3] World first use of artificial pancreas for Type 1 diabetes 'a success'. *Cambridge Networks* [online]. [cit. 2015-04-25]. Dostupné z: <https://www.cambridgenetwork.co.uk/news/first-use-of-artificial-pancreas-type1-diabetes-success/>
- [4] KOUTNY, Tomas. Glucose predictability, blood capillary permeability, and glucose utilization rate in subcutaneous, skeletal muscle, and visceral fat tissues. *Computers in Biology and Medicine*. 2013, vol. 43, issue 11, s. 1680-1686. DOI: 10.1016/j.combiomed.2013.08.008. Dostupné z: <http://linkinghub.elsevier.com/retrieve/pii/S0010482513002230>
- [5] CELEDA, Tomas. *Optimalizace pářením včelí královny*. Praha, 2011. Dostupné z: <https://cyber.felk.cvut.cz/research/theses/papers/132.pdf>. Diplomová práce. ČVUT. Vedoucí práce Doc. Ing. Jiří Lažanský, CSc.
- [6] WHITING, David R., Leonor GUARIGUATA, Clara WEIL a Jonathan SHAW. IDF Diabetes Atlas: Global estimates of the prevalence of diabetes for 2011 and 2030. *Diabetes Research and Clinical Practice*. 2011, vol. 94, issue 3, s. 311-321. DOI: 10.1016/j.diabres.2011.10.029. Dostupné z: <http://linkinghub.elsevier.com/retrieve/pii/S0168822711005912>
- [7] YOUQING WANG a XIANGWEI WU. An adaptive glucose prediction method using auto-regressive (AR) model and Kalman filter. *Proceedings of 2012 IEEE-EMBS International Conference on Biomedical and Health Informatics*. IEEE, 2012, s. 293-296. DOI: 10.1109/BHI.2012.6211570. Dostupné z: <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6211570>

- [8] ABBASS, H.A. MBO: marriage in honey bees optimization-a Haplometrosis polygynous swarming approach. *Proceedings of the 2001 Congress on Evolutionary Computation (IEEE Cat. No.01TH8546)*. IEEE, 2001, s. 207-214. DOI: 10.1109/CEC.2001.934391. Dostupné z: <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=934391>
- [9] Diabetes melitus 1. typu a těhotenství. FN MOTOL. *FN Motol: Informace o diabetes melitus* [online]. 2012. vyd. [cit. 2015-04-20]. Dostupné z: <http://www.fnmotol.cz/kliniky-a-oddeleni/cast-pro-dospele/interni-klinika-uk-2lf-a-fn-motol/pro-pacienty/informace-o-diabetes-melitus/diabetes-mellitus-1typu-a-tehotenstvi/>
- [10] BRANDNER, Marek a Petr PŘIKRYL. *Numerické metody II*. Skripta ZČU. Dostupné z: http://home.zcu.cz/~danek/project/DATA/WWW_STRANKY/soubory/Prikryl,Brandner-NumerickeMetody2.pdf
- [11] FACCHINETTI, Andrea, Giovanni SPARACINO a Claudio COBELLI. Reconstruction of Glucose in Plasma from Interstitial Fluid Continuous Glucose Monitoring Data: Role of Sensor Calibration. University of Padova, Italy, 2007. Dostupné z: <http://www.ncbi.nlm.nih.gov/pubmed/19885129>
- [12] WELSCH, Greg a Bishop GARRY. UNIVERSITY OF NORTH CAROLINA AT CHAPEL HILL. *An introduction to Kalman Filter*. University of North Carolina at Chapel Hill, 2001. Dostupné z: http://www.cs.unc.edu/~tracker/media/pdf/SIGGRAPH2001_CoursePack_08.pdf
- [13] Umělá slinivka. DIABETICKÁ ASOCIACE ČR. *Diabetická asociace ČR* [online]. [cit. 2015-04-22]. Dostupné z: <http://www.diabetickaasociace.cz/radi/umela-slinivka-nadeje-blizke-budoucnosti-pro-diabetiky/>
- [14] REBRIN, Kerstin a Garry M. STEIL. Can Interstitial Glucose Assessment Replace Blood Glucose Measurements?. *Diabetes Technology*. 2000, vol. 2, issue 3, s. 461-472. DOI: 10.1089/15209150050194332. Dostupné z: <http://online.liebertpub.com/doi/abs/10.1089/15209150050194332>
- [15] NOCEDAL, Jorge a Stephen J WRIGHT. *Numerical optimization*. New York: Springer, 1999, 636 s. Springer series in operations research. ISBN

03-879-8793-2. Dostupné z: http://www.bioinfo.org.cn/~wangchao/maa/Numerical_Optimization.pdf

- [16] CROEZE, Alfonso, Lindsey PITTMAN a Winney REYLONDS. LOUISIANA STATE UNIVERSITY. *Solving nonlinear least-squares problems with the Gauss-Newton and Levenberg-Marquardt methods*. Louisiana State University, 2013. Dostupné z: <https://www.math.lsu.edu/system/files/MunozGroup1%20-%20Paper.pdf>
- [17] KOUTNY, T. Prediction of Interstitial Glucose Level. *IEEE Transactions on Information Technology in Biomedicine*. 2012, vol. 16, issue 1, s. 136-142. DOI: 10.1109/TITB.2011.2177469. Dostupné z: <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6087283>

Seznam obrázků

1.1	Výskyt diabetu napříč jednotlivými kontinenty (<i>počty jsou v milionech lidí</i>)	6
1.2	Mezinárodní symbol diabetu	7
1.3	Metabolismus glukóza - glykogen (<i>zdroj: Wikimedia Commons pod licencí public domain</i>)	8
2.1	Reziduum u metody nejmenších čtverců (<i>Černé body jsou naměřené hodnoty, modrá křivka je výsledek NLS.</i>)	17
2.2	Ukázka Kalmanova filtru (<i>modrá - vstup, zelená - originál, červená - Kalmanův filtr</i>) (<i>zdroj: Modelica Buildings Library pod licencí Modelica Licence 2</i>)	18
3.1	Počáteční odhad	23
3.2	Výsledek druhé iterace NLS (<i>červená - nové hodnoty, modrá - inicializace</i>)	27
3.3	Graf porovnání výsledků (<i>modrá - Mathematica, zelená - Matlab, červená - vlastní implementace, černá - naměřená data</i>)	31
5.1	Statový diagram průběhu a aktualizace hodnot	39
5.2	Fáze filtrování	42
5.3	Ukázka dat s příliš častým vzorkováním (<i>Zelená - dlouhé vzorkování, Červená - krátké vzorkování</i>)	45
6.1	Metrika	46
6.2	Lichoběžníková metoda integrace	47
8.1	Relativní chyby modelu Steil-Rebrinové	53
8.2	Relativní chyby modelu řešeném HBMO	54
8.3	Relativní chyby Kalmanova filtru a auto regresivního modelu	56
8.4	Relativní chyby difuse modelu	57

Seznam tabulek

3.1	Vysvětlení jednotlivých proměnných rovnice 3.1	19
3.2	Data pro aplikaci NLS	21
3.3	Porovnání počtu operací	30
3.4	Srovnání vlastní implementace a Wolfram Mathematica	31
4.1	Vysvětlení proměnných rovnic 4.1 a 4.2	33
4.2	Iterace a počet nových mutací (<i>První číslo říká, kolik bylo nalezených mutací a druhé, zda byly použity.</i>)	37
4.3	Iterace a počet nových mutací po přidání věku (<i>První číslo říká, kolik bylo nalezených mutací a druhé, zda byly použity.</i>)	37
4.4	Sumář výsledků po svatebních letech (<i>Metoda bez přidáním věku řešení.</i>)	38
4.5	Sumář výsledků po svatebních letech (<i>Metoda s přidáním věkem řešení.</i>)	38
5.1	Vysvětlení jednotlivých proměnných	41
5.2	Vysvětlení jednotlivých proměnných	44
8.1	Relativní chyby modelu Steil-Rebrinové	53
8.2	Relativní chyby modelu řešeném HBMO	54
8.3	Relativní chyby Kalmanova filtru a auto regresivního modelu	55
8.4	Relativní chyby difuze modelu	56